Generation and Grounding of Natural Language Descriptions for Visual Data

A dissertation submitted towards the degree
Doctor of Engineering
(Dr.-Ing.)
of the Faculty of Mathematics and Computer Science
of Saarland University

by

Anna Rohrbach, M.Sc.

Saarbrücken
March 2017

Day of Colloquium	15^{th} of May, 2017
Dean of the Faculty	Univ.-Prof. Dr. Frank-Olaf Schreyer

Examination Committee

Chair	Prof. Dr. Antonio Krüger
Reporters	Prof. Dr. Bernt Schiele
	Prof. Dr. Vera Demberg
	Prof. Trevor Darrell, Ph.D.
Academic Assistant	Dr. Björn Andres

ABSTRACT

Generating natural language descriptions for visual data links computer vision and computational linguistics. Being able to generate a concise and human-readable description of a video is a step towards visual understanding. At the same time, grounding natural language in visual data provides disambiguation for the linguistic concepts, necessary for many applications. This thesis focuses on both directions and tackles three specific problems.

First, we develop recognition approaches to understand video of complex cooking activities. We propose an approach to generate coherent multi-sentence descriptions for our videos. Furthermore, we tackle the new task of describing videos at variable level of detail.

Second, we present a large-scale dataset of movies and aligned professional descriptions. We propose an approach, which learns from videos and sentences to describe movie clips relying on robust recognition of visual semantic concepts.

Third, we propose an approach to ground textual phrases in images with little or no localization supervision, which we further improve by introducing Multimodal Compact Bilinear Pooling for combining language and vision representations. Finally, we jointly address the task of describing videos and grounding the described people.

To summarize, this thesis advances the state-of-the-art in automatic video description and visual grounding and also contributes large datasets for studying the intersection of computer vision and computational linguistics.

ZUSAMMENFASSUNG

Die Erstellung natürlicher Sprachbeschreibungen für visuelle Daten verbindet Computer Vision und Computerlinguistik. Die Fähigkeit eine prägnante und menschlich lesbare Beschreibung eines Videos zu produzieren, ist ein Schritt zum visuellen Verständnis. Gleichzeitig ermöglicht Lokalisierung der natürlichen Sprache in visuellen Daten die Disambiguierung der sprachlichen Konzepte. Diese Dissertation konzentriert sich auf beide Richtungen wie folgt.

Zuerst entwickeln wir Methoden, um komplexe Kochaktivitäten in Videos zu verstehen und für diese dann kohärente Multi-Satz-Beschreibungen mit variabler Detaillierung zu generieren.

Zweitens präsentieren wir einen umfangreichen parallelen Datensatz von Filmen mit professionellen Beschreibungen. Wir schlagen einen Ansatz vor, der aus Videos und Sätzen lernt Videoclips zu beschreiben, und der sich auf einer robusten Erkennung visueller Konzepte stützt.

Drittens schlagen wir einen Ansatz vor, um sprachliche Konzepte in Bildern mit wenig oder keiner Überwachung zu lokalisieren, den wir durch eine neue multimodale Kombination der Sprach- und Bild-Repräsentationen verbessern. Abschließend beschreiben wir Videos während wir gleichzeitig die beschriebenen Personen lokalisieren.

Zusammenfassend stellt diese Dissertation neue Methoden in der automatischen Videobeschreibung und Lokalisierung natürlicher Sprache in visuellen Daten vor. Zur weiteren Forschung am Schnittpunkt von Computer Vision und Computerlinguistik trägt diese Dissertation große Datensätze bei.

CONTENTS

1 INTRODUCTION

Humans live in a multi-modal world, where vision and language are the primary channels of perception and communication. Naturally, we would like to develop machines, or intelligent agents, that are able to similarly communicate with us. Such agents should be able, among other things, to describe what they see, understand what we refer to and answer questions about the visual world. The interplay between natural language understanding and visual recognition is thus an important research direction, studied in the computer vision as well as computational linguistics communities.

Many multi-modal tasks that involve vision and language have emerged recently. One such task that has drawn a lot of attention is *generating* natural language descriptions for images and videos (Venugopalan *et al.*, 2015a; Vinyals *et al.*, 2015). Such descriptions make visual data accessible in text form, which enables many applications, e.g. visual search or Audio Description (AD) for the visually impaired people. The main difference of the image/video description task from the classical visual recognition problems is that it goes beyond predicting a set of class labels by generating a coherent description of the entire visual scene. An important challenge in working with language is that words are ambiguous and they exist independently of specific image instances. It is thus important to enable machines to understand, or *ground*, natural language descriptions in visual data. Such grounding can be done at different levels, e.g. coarsely, like video-to-text alignment (Bojanowski *et al.*, 2015; Dogan *et al.*, 2016) or very finely, like localizing textual phrases in images (Kazemzadeh *et al.*, 2014). This provides further research opportunities and interesting applications. Such ability would allow interacting with a robot by asking to localize certain visual objects or answering questions about visual scenes.

Vision and language have been shown to benefit each other, e.g. we found that vision can provide useful cues for common sense knowledge acquisition (Tandon *et al.*, 2016). In this thesis we focus on three problems which are introduced in the following. There are, of course, other tasks that involve language and vision interaction, e.g. cross-modal retrieval (Wang *et al.*, 2016a), which are not the focus of this thesis.

An automatic computer assistant, that is able to understand our daily activities by watching us, can assist us more effectively and provide help to people with special needs. Human daily activity recognition has been extensively studied in the past (Tenorth *et al.*, 2009; la Torre *et al.*, 2009). In particular, the long, composite activities, which consist of many fine-grained steps are challenging to understand (Rohrbach *et al.*, 2012b). In this thesis we target the task of *video description of fine-grained cooking activities*. Our goal is to recognize the fine-grained cooking activities and then to describe them automatically with natural language. In the first part we employ hand-

centric visual representations targeted to both activities and manipulated objects. In the second part we propose approaches to describe long cooking videos with multiple sentences and at multiple levels of detail. Chapters 3 and 4 of the thesis focus on fine-grained cooking video understanding and description.

In addition to the challenging fine-grained activities in the cooking scenario, we also address an open domain scenario. Motivated by a lack of a large parallel corpus of videos and sentences, we present a new large-scale dataset of movies with associated textual descriptions. We source these descriptions from movie scripts as well as Audio Descriptions (AD) (Salway, 2007), also known as Descriptive Video Service, for the visually impaired people. AD is provided as additional audio stream in the movies to help the visually impaired to better follow the story. The collected movie data is diverse and visually challenging. We propose an approach to automatic movie description, which recognizes diverse activities, objects and locations in movies and translates the predictions to a sentence with a recurrent neural network. We have also organized two workshops and challenges for movie description to help foster research in this area. Chapter 5 extensively discusses our endeavor on *large-scale movie description*.

As discussed above, the goal of visual grounding is to disambiguate language concepts by linking them to visual concepts. In particular, we are interested in the task of localizing natural language phrases in visual data. The classical object detection task, i.e. given an image, to predict a bounding box for a given object class, is a well researched problem with a long history (Viola and Jones, 2001; Felzenszwalb *et al.*, 2010). The object classes are predefined and their number is limited. At the same time, language provides a natural way of describing as well as referring to particular objects in the visual scene. In this thesis we address the task of grounding textual phrases in images with bounding boxes. Importantly, we focus on a scenario when limited localization supervision is available. Although generating video descriptions and visual grounding are typically addressed separately, it is natural to reason about them jointly. I.e. while describing a video one could also try to localize the described concepts. To this end we propose a novel task of grounded video description, where we describe video while jointly grounding the described people. Chapters 6, 7 and 8 focus on *language grounding and grounded video description*.

The rest of this chapter is organized as follows. First we discuss the main challenges towards solving the aforementioned tasks and our contributions to address them (Section 1.1). Next, we provide an outline of the thesis in Section 1.2.

1.1 CONTRIBUTIONS OF THE THESIS

Tasks which involve both linguistic and visual modalities typically require addressing the following aspects: a) language representation, b) visual recognition, c) joint language and visual modeling. The language representation should capture language semantics and syntax. The visual recognition should provide information about a visual scene including objects and activities. The joint language-vision modeling

depends on the task that we are addressing. In video description we aim to *translate* the visual signal into language, i.e. generate a novel description. To do so we need to understand what is in the scene, which parts of it to describe and at which level of abstraction. In visual grounding we aim to estimate the *compatibility* between a language query and multiple visual regions in order to find the most relevant region. Both the language query and the visual scene can be complex. However, grounding requires a detailed understanding of both language and vision to estimate their compatibility. Additionally, learning joint representations requires parallel corpora with aligned language and visual data, which are not always readily available. In the following we detail these and related challenges towards generation and grounding of natural language descriptions for visual data, as well as the contributions this thesis makes to address them.

1.1.1 Video description of fine-grained cooking activities

The first target of the thesis is understanding and describing fine-grained cooking activities. Specifically, we work with the cooking activity dataset "MPII Cooking 2", presented in Chapter 3 of this thesis. Each of our videos represents a dish preparation (e.g. pizza, scrambled eggs) and consists of many steps, which involve fine-grained activities (e.g. take out, cut dices) and objects (e.g. broccoli, spice shaker). "MPII Cooking 2" provides low level semantic annotations (e.g. activity, object, tool, location) for all videos. We start with defining the main challenges in this research area.

1.1.1.1 *Challenges*

Lack of a large parallel dataset of videos and sentences for long-term composite activities. Typical existing datasets consist of short clips described with just a single sentence. To learn describing long-term composite activities it is necessary to have a parallel dataset of videos and sentences with the following properties. First, such a dataset should have continuous multi-sentence video descriptions. Second, is should provide high-level as well as detailed descriptions of a video. Depending on the context we might be either interested in a detailed description or rather want to read a short summary of a video or simply find out that e.g. "The person made a cup of coffee."

Fine-grained visual recognition. Performing recognition in domains with low inter-class and high intra-class variation is naturally challenging. E.g. in cooking videos it is important to successfully distinguish "peeling an orange" from "cutting an orange". Most activity recognition datasets and approaches do not focus on fine-grained activities. Additionally, in a cooking scenario many activities involve little body movement and rather focus on hands.

Multi-sentence description. Most research on automatic video description focuses on describing short video clips with single sentences. Such short clips are either specifically collected or manually pre-segmented from longer videos. To address multi-sentence description, however, one needs to automatically segment the videos and produce coherent descriptions for them. The latter is especially important, as descriptions that "jump" from one subject to another would be unnatural.

Multi-level description. For the long and complex videos it is important to provide descriptions at different levels of detail for different purposes, i.e. detailed descriptions or short summaries. This ability requires understanding of the differences between the language used in different scenarios. To the best of our knowledge no prior work has addressed this problem.

1.1.1.2 *Contributions*

The following summarizes the contributions for *video description of fine-grained cooking activities*.

The first contribution is the Tacos Multi-Level dataset, which augments 185 cooking videos with detailed multi-sentence descriptions (up to 15 sentences). Moreover, Tacos Multi-Level provides two additional descriptions for each cooking video: short (3-5 sentences) and single sentence, thus making it possible to study multi-level video description, see Chapter 4. The dataset contains over 24K video clips with almost 16 hours of video.

The second contribution of this thesis is an approach to fine-grained activity and object recognition. Our approach relies on the fact that hands are frequently highly informative for the hand-centric type-of-activities, such as cooking. We combine holistic motion features with hand-centric features informative of motion, color and appearance (Chapters 3, 4).

The third contribution is an automatic temporal segmentation algorithm as well as a coherent multi-sentence description approach. Our approach is based on the method of Rohrbach *et al.* (2013b), which first predicts an intermediate semantic representation (SR) from a video and then translates it with Statistical Machine Translation (SMT) (Koehn, 2010). The approach relies on the low level semantic annotations provided with the dataset to predict the SR. To ensure coherence across different generated sentences we integrate high level topic information in this approach, and ensure the consistent prediction across multiple video segments (Chapter 4).

The fourth contribution is a novel task of multi-level video description, or video description at different levels of detail. On the Tacos Multi-Level dataset we find that the language used in detailed and short descriptions is quite similar, while single-sentence descriptions differ. Thus we propose extractive/abstractive summarization-based approaches to obtain short summaries/single-sentence descriptions, respectively, from detailed descriptions, see Chapter 4.

1.1.2 Large-scale movie description

Next we address a broader domain, more specifically, movies. Given a movie clip we aim to generate a natural language description. Unlike the previous scenario, in this case we are given no low level semantic annotations for the training videos, only the sentence descriptions. When moving to open domain video, new challenges arise.

1.1.2.1 *Challenges*

Lack of a large-scale, open domain video description dataset. Good datasets benefit most areas of research, however many existing datasets on video description suffer from some limitations, e.g. size or diversity. We have witnessed how large scale datasets have benefited Deep Learning approaches for the tasks of object (Deng *et al.*, 2009) or scene (Zhou *et al.*, 2014) recognition. In order to learn to describe videos we also require a large dataset of videos and sentences. Challenges (competitions) are also important for measuring progress in the field, comparing different approaches and understanding the main difficulties. We witnessed a lot of advancement in object classification and image captioning largely due to popular challenges, like the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) (Russakovsky *et al.*, 2015) and the MS COCO Captioning Challenge (Chen *et al.*, 2015). There is thus a demand for a similar video description challenge.

Noisy sentence annotations. When working with the cooking videos we had access to manually annotated low level semantic representations (e.g. activity, object). For large-scale and open domain vdeos we need to learn to translate videos to language without relying on an intermediate semantic representation. We are given videos paired with sentences, while the sentences can be noisy or mention non-visual concepts. It is thus rather challenging to capture various interactions which exist between the two modalities.

Large-scale visual recognition. When working with large-scale, diverse video data visual recognition becomes more challenging. While object recognition in images has made impressive progress in the last years, these findings do not immediately transfer to video. Not only is the movie data different from typical object recognition data, e.g. ImageNet Deng *et al.* (2009), due to a domain shift, but also the videos bring additional challenges, like people and camera motion. Combined with the need to recognize various aspects of the video, like locations and activities, all of this makes visual recognition a real challenge.

1.1.2.2 *Contributions*

In the following we present the contributions of the thesis for *large-scale movie description*.

The first contribution is a large-scale dataset, MPII Movie Description (MPII-MD), which consists of movies and associated textual descriptions (Chapter 5). Movies

provide an excellent source of long, realistic and diverse videos. The descriptions are obtained from professionally written movie scripts and Audio Descriptions (AD) for the visually impaired. We analyze AD for the first time in the computer vision community and show that they provide a good resource for learning video-language models. All sentences are cleaned-up and manually aligned to the video. The dataset contains 94 movies, 68,337 clips and almost 78 hours of video in total. To date MPII-MD was requested by almost 200 research groups from all over the world.

The second contribution is an approach to movie description, *Visual-Labels*, which extracts the most reliable and *visual* information from sentence annotations, i.e. the visual semantic labels (actions, objects, locations) and learns the visual classifiers to recognize them. To tackle the large-scale visual recognition we employ state-of-the-art visual representations to recognize actions, objects and locations, and subsequently focus on the most reliable ones. We then rely on a Long Short-Term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997) to translate the visual representations into sentence descriptions, see Chapter 5.

The third contribution is the Large Scale Movie Description Challenge (LSMDC) which is based on the MPII-MD and M-VAD datasets (Torabi *et al.*, 2015). We set up the evaluation server where the participants can submit to a public test set and to a blind test set (only video available). We perform automatic as well as human evaluation of the challenge submissions and determine the winner according to human evaluation on the blind test set. The challenge results were presented at two workshops, one at the International Conference on Computer Vision (ICCV) in 2015 and the second at the European Conference on Computer Vision (ECCV) in 2016. In addition to the movie description task, in the last challenge edition we also proposed the fill-in-the-blank task (Maharaj *et al.*, 2017). In a related effort, Torabi *et al.* (2016) contributed a movie retrieval track for LSMDC 2016. We discuss and extensively analyze the submissions of the challenge participants in Chapter 5.

1.1.3 Language grounding and grounded video description

The goal of visual grounding in this thesis is to localize natural language phrases in images and videos. Our first target is predicting a bounding box for a phrase given an image. Considering the high cost of collecting localization supervision (bounding boxes) for phrases, we aim to reduce the need for such supervision. Our second target is to combine video description and visual grounding. Namely we propose a task of video description with grounded and co-referenced people.

1.1.3.1 *Challenges*

Limited localization supervision for visual grounding. To provide supervision for the visual grounding task we need to annotate natural language phrases along with the corresponding bounding boxes in images or spatial-temporal tubes in videos. Such annotation is costly and does not scale well. Moreover, many datasets only provide images/videos with sentences without any localization information.

We thus require approaches that can work with little or no localization supervision.

Modeling compatibility between language and vision. Language and visual representations are informative of the respective modalities, however it is not obvious how to combine them or relate them to each other. In the visual grounding task we need to match a phrase to a particular region of an image. As the natural language phrases can be rather complex, e.g. "a second boy right from the car", it is challenging to correctly match such phrases to image regions.

Joint description generation and grounding. Ultimately, we would like to generate descriptions which can also be grounded in the visual data. However, typically description and grounding tasks are addressed separately, as for grounding we assume that the description is already given. Only few recent works look into predicting descriptions jointly with grounding them in the visual data.

Language and visual co-references. One aspect of video description which received little attention in the literature, is handling of co-references. When we encounter a repeating entity (a person or an object) we should be able to refer to this entity as "he","it", etc. Establishing connections between the previous scene and the current scene requires solving the "visual co-reference resolution" problem. Specifically, we need to link the repeating entities in the video and then transfer this link in the generated description through the grounding.

1.1.3.2 *Contributions*

Here we present the contributions of the thesis for the *language grounding and grounded video description*.

The first contribution is our approach GroundeR (GROUNDing by Reconstruction) which can operate in different supervision regimes: fully-supervised, semi-supervised and unsupervised, w.r.t. the localization supervision. In order to learn to select the correct bounding box given a phrase query and an image, we introduce a reconstruction loss. Specifically, we reconstruct the phrase from the "attended" (selected) region and compare the output with the ground-truth query. In short, once the reconstruction is correct, the grounding must also be correct. For details see Chapter 6.

The second contribution is a comparison of different ways of combining visual and language representations (e.g. concatenation, element-wise product) and, as a result, the proposed Multimodal Compact Bilinear pooling (MCB). MCB efficiently approximates the outer product of the two input vectors. We show that this pooling is beneficial for visual grounding as well as visual question answering (VQA), consequently winning the VQA challenge (Antol *et al.*, 2015) with real images in 2016, see Chapter 7.

The third contribution is an approach which addresses a new task of generating video descriptions with grounded and co-referenced people. It jointly generates

a video description and predicts grounding, co-reference and gender for all the generated human entities. Our approach relies on an attention mechanism which reasons about grounding and local co-reference over two adjacent sentences and clips. For details see Chapter 8.

The fourth contribution is to supply the attention mechanism in our grounded video description approach with automatically obtained localization (grounding and co-reference) supervision. The supervision comes from the linking between name mentions (e.g. Mary, Paul) and visual tracks, obtained with our weakly-supervised approach GroundeR (Chapter 8).

1.2 OUTLINE OF THE THESIS

In this section we shortly discuss each chapter of the thesis and indicate the collaborations with other researchers.

Chapter 2: Related Work. In this chapter we review related work on automatic video description, video description datasets and visual grounding, as well as other related topics.

Chapter 3: Recognizing Fine-Grained and Composite Activities using Hand Centric Features and Script Data. This chapter presents the *MPII Cooking 2* dataset of cooking videos. We describe our approaches to fine-grained activity and participating object recognition as well as composite activity recognition. In particular, for fine-grained recognition of activities and objects we propose a combination of holistic and hand-centric features.

The content of this chapter corresponds to the IJCV 2016 publication "Recognizing Fine-Grained and Composite Activities using Hand-Centric Features and Script Data" (Rohrbach *et al.*, 2016b), which is partially based on Rohrbach *et al.* (2012a) and Rohrbach *et al.* (2012b). Marcus Rohrbach was the lead author of these papers. Anna Rohrbach significantly contributed with the hand-centric approach for the fine-grained activity and object recognition, experiments and an overall discussion.

Chapter 4: Coherent Multi-Sentence Video Description with Variable Level of Detail. This chapter presents the *Tacos Multi-Level* corpus of descriptions collected for the MPII Cooking 2 video dataset (Chapter 3). We propose an approach to coherent multi-sentence video description. We also propose a new task, multi-level video description and present an approach to address it.

The content of this chapter corresponds to the GCPR 2014 publication: "Coherent Multi-Sentence Video Description with Variable Level of Detail" (Rohrbach *et al.*, 2014), which was accepted as an *Oral*. Anna Rohrbach was the lead author of the paper. This work was done in collaboration with Department of Computational Linguistics, Saarland University, who contributed to the

analysis of the collected corpus and implementation of the probabilistic input for Statistical Machine Translation (SMT) with a word lattice.

Chapter 5: Movie Description. In this chapter we propose the new large-scale *MPII Movie Description* (MPII-MD) dataset. This video description dataset is large, open domain, and relies on professionally written descriptions of movies. In particular we source movie scripts, available online, and Audio Descriptions, available for some movies to help the visually impaired people to follow the events in the movie more easily. We also propose an approach to automatic movie description, called *Visual-Labels*. Our approach makes use of sentence descriptions to extract semantic labels and learns respective visual classifiers. It then uses an LSTM network to translate the visual features into a sentence. Finally, we present the Large-Scale Movie Description Challenge, based on the MPII-MD and M-VAD (Torabi *et al.*, 2015) datasets. In this chapter we review the challenge submissions and discuss their results.

The content of this chapter is based on the following publications: CVPR 2015 publication: "A Dataset for Movie Description" (Rohrbach *et al.*, 2015b); GCPR 2015 publication: "The Long-Short Story of Movie Description" (Rohrbach *et al.*, 2015a) (accepted as an *Oral* and received an *Honorable Mention prize*) and IJCV 2017 publication: "Movie Description" (Rohrbach *et al.*, 2017b). Anna Rohrbach was the lead author of these papers. Atousa Torabi (Disney Research, Pittsburgh) contributed with the M-VAD dataset, which became part of the LSMDC.

Chapter 6: Grounding of Textual Phrases in Images by Reconstruction. This chapter presents our approach to grounding (localizing) textual phrases in images. We propose an approach GroundeR (Grounding by Reconstruction), which aims to reconstruct the query phrase from the selected/attended subset of the image. We thus ensure that we select/attend to a correct image region, which corresponds to a phrase, i.e. we ground it. We evaluate our approach in fully-, semi- and un-supervised settings on two datasets and show consistent improvement over prior work in all of them.

The content of this chapter corresponds to the ECCV 2016 publication: "Grounding of Textual Phrases in Images by Reconstruction" (Rohrbach *et al.*, 2016a), which was accepted as an *Oral* (1.8% acceptance rate). Anna Rohrbach was the lead author of the paper.

Chapter 7: Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding. In this chapter we explore different ways of combining language and visual representations. We present a Multimodal Compact Bilinear pooling, an effective and expressive way of combining two representations. We show how this pooling improves the performance of our visual grounding approach, GroundeR (Chapter 6), and additionally study the related task, Visual Question Answering (VQA).

The content of this chapter corresponds to the EMNLP 2016 publication: "Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding" (Fukui *et al.*, 2016). Akira Fukui, Dong Huk Park, Daylen Yang and Anna Rohrbach have equal contribution to the paper. Akira Fukui, Dong Huk Park and Daylen Yang contributed to the VQA experiments, while Anna Rohrbach contributed to the visual grounding experiments and overall discussion. The proposed approach to VQA has taken the first place in an open-ended VQA challenge (Antol *et al.*, 2015) with real images in 2016.

Chapter 8: Generating Descriptions with Grounded and Co-Referenced People. This chapter aims to integrate the video description and visual grounding problems. We present a new task of video description with grounded and co-referenced people. More specifically, we want to generate a video description, while jointly localizing (grounding) the human entities in video and co-referencing their occurrence across two consecutive clips/sentences. Our proposed approach address all these tasks jointly, additionally performing gender recognition for the described human entities. We also supply our approach with supervision, by performing name-visual track linking relying on our GroundeR approach (Chapter 6).

The content of this chapter corresponds to the CVPR 2017 publication (Rohrbach *et al.*, 2017a). Anna Rohrbach was the lead author of the paper.

Chapter 9: Conclusions and future perspectives. Finally, in this chapter, we summarize the thesis and discuss some future research directions in the areas on video description and visual grounding.

RELATED WORK 2

As discussed in the previous chapter, this thesis focuses on three research directions: *video description of fine-grained cooking activities*, *large-scale movie description* and *language grounding and grounded video description*. Essentially, all three directions are concerned with video description, while the third one focuses more on visual grounding. Therefore, in this chapter we present the related work, structured around: video description datasets (Section 2.1) and approaches (Section 2.2), and visual grounding (Section 2.3). Some of the following chapters present the related work for specific topics covered in these chapters.

2.1 VIDEO DESCRIPTION DATASETS AND BENCHMARKS

Machine learning in general, and deep learning in particular, benefit from large datasets, such as ImageNet (Deng *et al.*, 2009), a large-scale dataset for object class recognition. While the video description task has been explored as early as 2002 (Kojima *et al.*, 2002), there were no large datasets for this task until recently. Most early datasets were also rather constrained visually (DARPA, 2011). This is one reason why early approaches were limited to specific scenarios and relied on manually defined rules, rather than learned to generate novel descriptions from data. More recently, video depicting daily activities, such as cooking (Regneri *et al.*, 2013), and YouTube videos (Chen and Dolan, 2011) became a popular source of data. These datasets tend to be more diverse and realistic, but still limited in size. In 2015-2016 a number of large-scale open domain video description datasets have been proposed (Xu *et al.*, 2016; Zeng *et al.*, 2016). In the following we review the existing parallel datasets of videos and sentences, organized by the video domain. In the end we relate the contributions of this thesis to the prior work.

2.1.1 Surveillance video descriptions

Some early works Barbu *et al.* (2012); Hanckmann *et al.* (2012) use a subset of DARPA **Mind's Eye Y1** (Year 1) dataset (DARPA, 2011) for the video description task. The dataset features surveillance-type videos paired with verb phrases constructed from 48 different verbs. The corpus consists of 3,480 training and 749 test videos.

2.1.2 TRECVID video descriptions

The Text REtrieval Conference (TREC) of the U.S. National Institute of Standards and Technology (NIST) is a long-run conference which focuses on information retrieval

with yearly organized competitions. TRECVID (TREC VIDeo)[1] is an independent track which includes various video related tasks (e.g. video indexing, content-based retrieval). TRECVID features internet videos with user-provided metadata (title, keywords, description, etc). Typically, only the challenge participants have access to the data and only for the time of the competition. Subsets of the TRECVID challenge data have been used for video description research. E.g.**TRECVID 2010 Multimedia Event Detection (MED10)** task (Over *et al.*, 2010), which consists of three events: "making a cake", "batting a run", and "assembling a shelter", was used by Tan *et al.* (2011). They generate textual descriptions for 140 MED test videos that contain at least one of the aforementioned events. Another subset, **TRECVID 2012 Multimedia Event Detection (MED12)** (Over *et al.*, 2012) covers 25 event categories, while each category has around 200 videos. It was used by Das *et al.* (2013) to train their video description system. They also use the subset of the **Multimedia Event Recounting (MER12)** (Over *et al.*, 2012) data which covers 5 out of 25 events from MED12 ("cleaning an appliance", "renovating a home", etc.). Specifically, there are six test videos for each event. As the access to MER12 test data is restricted, Das *et al.* (2013) collect text descriptions for these videos in-house.

The TRECVID 2016 edition features a new task, **Video to Text Description** (Awad *et al.*, 2016). The new dataset contains over 30,000 Twitter Vine[2] videos. In the presented task around 2,000 videos are used, each video is annotated by two distinct annotators. Given a set of 2,000 video URLs and two sets of descriptions, the participants have to submit their results for at least one of the two tasks: a) Matching and Ranking, i.e. "Return for each video URL a ranked list of the most likely text descriptions that correspond to the video from each of the sets"; b) Description Generation, i.e. "Automatically generate for each video URL a text description (1 sentence) independently and without taking into consideration the existing sentences" (Awad *et al.*, 2016). The latter is evaluated with standard automatic metrics, e.g. METEOR (Lavie, 2014) and BLEU (Papineni *et al.*, 2002), as well as with an experimental Semantic Textual Similarity (STS) (Han *et al.*, 2013) metric. The test data is publicly available.

2.1.3 Description datasets of daily activities

The **TACoS** (Textually Annotated Cooking Scenes) dataset, presented by Regneri *et al.* (2013), is based on the "MPII Composites" cooking video corpus (Rohrbach *et al.*, 2012b). TACoS augments the cooking videos with multiple textual descriptions collected by crowd-sourcing via Amazon Mechanical Turk (AMT)[3]. The dataset covers 127 videos from "MPII Composites", each described by 20 turkers. The descriptions are collected by letting a turker watch a video, then stop at a particular point and enter a sentence describing what has just happened in the scene. In total, the dataset contains 18,227 sentences linked to 7,206 unique video intervals.

[1] http://trecvid.nist.gov/

[2] A hosting service for user-generated six-second-long looping videos, https://vine.co/

[3] http://www.mturk.com

Das *et al.* (2013) present **YouCook**, a dataset which consists of 88 (49 training and 39 test) long YouTube cooking videos. The featured videos cover different cooking styles, e.g. baking, grilling, diverse kitchen environments and have dynamic camera changes. The ground-truth descriptions for the videos are collected via AMT, on average there are 8 multi-sentence descriptions per video. The training set also includes annotations for 48 participating objects and 7 activities.

Charades is a recent dataset of Sigurdsson *et al.* (2016), recorded by people in their homes while performing common household activities. The data collection process consists of three steps: script generation, script-guided video recording, verification and annotation. All steps are performed via AMT crowdsourcing. In total the dataset includes 9,848 videos with 27,847 sentence descriptions and features 267 people.

Several datasets are based on YouTube instruction videos, obtaining video descriptions directly from speech or user-provided descriptions. Alayrac *et al.* (2016) present a dataset of instruction videos for five tasks: changing a car tire, performing Cardio Pulmonary resuscitation (CPR), jumping a car, repotting a plant and making coffee. For each task they obtain 30 videos with English language speech transcripts from YouTube. They manually clean up the spelling and punctuation of the transcripts. The time alignment is obtained through the closed caption timings. Malmaud *et al.* (2015) collect a dataset of 180k cooking videos with aligned recipes. The videos are obtained from YouTube and also have English transcripts, which are not manually corrected. All the videos have accompanying textual descriptions, which contain the recipes. Malmaud *et al.* (2015) automatically extract the relevant parts of the descriptions which describe recipe steps or ingredients. Additionally they extract a set of 1.4M short video clips automatically annotated with an action and a noun phrase.

2.1.4 Description datasets of open domain web video

YouTube videos are frequently used to construct vision datasets, due to their diversity and availability. On the downside many YouTube videos suffer from rather low quality. The **Microsoft Video Description (MSVD)** corpus (Chen and Dolan, 2011) is a popular video description dataset sourced from YouTube. Initially it was collected for the tasks of language paraphrasing and translation, thus it has multi-lingual descriptions. Typically, in the video description research, only the English part of the corpus is used. It contains 1,970 short videos, each paired with around 16 sentence descriptions collected via AMT.

Habibian *et al.* (2014) propose the **VideoStory46K Dataset**, which consists of 45,826 videos from YouTube. The videos are comparatively long, 58.4 seconds on average, and in total there are 743 hours of videos. Each video is provided with a short user-generated caption. There are no precise temporally aligned descriptions provided with the video.

MSR-VTT (MSR-Video to Text) is a recent dataset presented by Xu *et al.* (2016). It is collected by querying a commercial video search engine with 257 queries from

20 categories (e.g. music, sports, news). The initial release of MSR-VTT contains 7,180 videos split into 10,000 clips (10-30 seconds on average) and the total duration is 41.2 hours. Each clip is annotated with about 20 natural sentences, which makes 200K clip-sentence pairs in total. Based on the MSR-VTT dataset the MSR **Video to Language Challenge**[4] was proposed as part of the Multimedia Grand Challenge 2016. The challenge includes both the automatic (BLEU@4 (Papineni *et al.*, 2002), METEOR (Lavie, 2014), ROUGE-L (Lin, 2004), and CIDEr-D (Vedantam *et al.*, 2015)) as well as human evaluation. The latter is done on a subset of the test set.

VTW (Video Titles in the Wild) (Zeng *et al.*, 2016) is a dataset which focuses on longer YouTube videos (1.5 minutes on average). VTW includes 18,100 videos with 1-3 sentence long user provided descriptions and editor provided titles (44,603 sentences in total). Unlike the standard video description works, Zeng *et al.* (2016) aim to generate concise video titles for these long videos.

2.1.5 Description datasets of image sequences

Li *et al.* (2016) present a new dataset, **Tumblr GIF (TGIF)**, that contains 100K animated GIFs from Tumblr[5] paired with 120K natural language descriptions obtained via crowdsourcing. GIFs are generated by users to represent concise and dynamic visual messages. They are rather short (3.10 seconds on average) and have no accompanying audio.

VIsual STorytelling (VIST) (Huang *et al.*, 2016) is a dataset of image sequences from Flickr data (Thomee *et al.*, 2015) accompanied by textual descriptions. The descriptions are collected in three ways. First are the standard descriptive captions for individual images. Second are the descriptions of images in sequence. And third are the stories for images in sequence, which use less descriptive and more narrative language. VIST contains 20,211 image sequences with 81,743 unique photos. The stories are centered around topics like a *party, amusement park, church* etc. Although the data does not contain videos but instead image sequences, it allows for studying multi-sentence story-driven description generation, also relevant for videos.

2.1.6 Movie scripts, audio descriptions and books

Movie scripts have been used for automatic discovery and annotation of scenes and human actions in videos (Duchenne *et al.*, 2009; Laptev *et al.*, 2008; Marszalek *et al.*, 2009), as well as a resource to construct activity knowledge bases (Tandon *et al.*, 2015; de Melo and Tandon, 2016). Others, like e.g. Bojanowski *et al.* (2013, 2014); Duchenne *et al.* (2009); Laptev *et al.* (2008); Marszalek *et al.* (2009), proposed datasets focused on extracting several activities from movies using movie scripts. Most of the movies are part of the "Hollywood2" dataset (Marszalek *et al.*, 2009) which contains 69 movies and 3669 clips. Another line of work (Cour *et al.*, 2009; Everingham *et al.*,

[4] http://ms-multimedia-challenge.com/challenge
[5] http://www.tumblr.com

2006; Ramanathan *et al.*, 2014; Sivic *et al.*, 2009; Tapaswi *et al.*, 2012) proposed datasets for character identification targeting TV shows.

Other prior work has looked at supporting Audio Descriptions (AD) production using scripts as an information source (Lakritz and Salway, 2006) and automatically finding scene boundaries (Gagnon *et al.*, 2010). ADs have been used to understand which characters interact with each other (Salway *et al.*, 2007). Salway (2007) analyse the linguistic properties on a non-public corpus of ADs from 91 movies. Their corpus is based on the original sources to create the ADs and contains different kinds of artifacts not present in actual description, such as dialogs and production notes.

Tapaswi *et al.* (2015) propose a dataset for book to movie alignment. It consists of the first season of the TV series Game of Thrones and the respective book, further denoted as GOT, and the Harry Potter and the Sorcerer's Stone book and movie, denoted as HP. The provided alignment between the book and the video is rather coarse, at a chapter/scene level. In total the dataset covers 73 GOT and 17 HP chapters, and 369 GOT / 138 HP movie scenes. In a related effort, Zhu *et al.* (2015b) propose a **MovieBook** dataset. It covers more movies (11) and introduces a more precise book-to-movie alignment, namely at a sentence or paragraph level. In total the MovieBook dataset contains 2,070 movie shot to sentence correspondences. Finally, Tapaswi *et al.* (2016) propose **MovieQA**, a dataset which focuses on answering questions about movies. Besides the QA data, the dataset features 408 subtitled movies with plot synopses sourced from Wikipedia. 199 movies also have aligned movie scripts and 60 have the AD sourced from our MPII-MD dataset (Chapter 5).

2.1.7 Relations to our work

In this section we relate prior work to the contributions of this thesis and, specifically, to the proposed video description datasets: TACoS Multi-Level, MPII Movie Description and Large Scale Movie Description Challenge (LSMDC) benchmark.

To the best of our knowledge the TACoS Multi-Level dataset (Chapter 4) is the only video/sentence dataset that provides descriptions at three levels of detail (detailed, short and single-sentence). Similar to TACoS (Regneri *et al.*, 2013), which is based on cooking videos from *MPII Composites* (Rohrbach *et al.*, 2012b), TACoS Multi-Level is based on the extended version, *MPII Cooking 2* (Chapter 3). However, unlike TACoS, our dataset contains the precise temporal alignment of sentences to video (not only the endpoint), descriptions at three levels of detail, and it is by a factor of 4 larger in number of sentences.

The recent large-scale video description datasets like MSR-VTT (Xu *et al.*, 2016), TGIF (Li *et al.*, 2016), VTW (Zeng *et al.*, 2016) all rely on web content, while our proposed MPII Movie Description (MPII-MD) dataset focuses on movies (Chapter 5). MPII-MD leverages movie scripts and Audio Descriptions (AD) aligned to movies. It is most similar to the concurrently published Montreal Video Annotation Dataset (M-VAD) (Torabi *et al.*, 2015). There are three differences between our and their corpus. First, MPII-MD consists both of movie scripts and ADs, while M-VAD only uses ADs. Second, we manually align every sentence to the corresponding activity in

the video, while M-VAD relies on automatic AD detection and uses its timestamps, leading to less precise alignment. Last, we use Blu-ray HD movies, while M-VAD uses DVDs. Both corpora are presented jointly as the LSMDC challenge in Chapter 5. LSMDC is the largest dataset to date in terms of a number of video clips (128K). While many existing datasets (e.g. MSVD, MSR-VTT, TGIF) focus on short clips described with a single sentence and do not allow studying longer video or multi-sentence description, VTW, TACoS Multi-Level and LSMDC allow for multi-sentence description and story understanding.

So far we have organized the LSMDC challenge twice, at the corresponding ICCV15 and ECCV16 workshops. The recent MSR Video to Language Challenge, presented in 2016, also raised high interest in the community. Additionally, TRECVID 2016 introduced a pilot challenge track Video to Text Description. All of this indicates that video description is a relevant and important problem, well established in the computer vision community. All challenges employ automatic evaluation metrics, while LSMDC and MSR Video to Language Challenge also perform human evaluation. To facilitate video understanding research and allow for purely automatic evaluation, the last edition of LSMDC in 2016 introduced the movie annotation and retrieval track (Torabi *et al.*, 2016) as well as the movie fill-in-the-blank track (Maharaj *et al.*, 2017).

Other works, datasets, and challenges are already building upon our data. Gao *et al.* (2016a) study the physical causality of action verbs using crowdsourcing to collect causality attributes for the TACoS Multi-Level sentences. Zhu *et al.* (2015b) learn a visual-semantic embedding from our movie clips and ADs to relate movies to books. Bruni *et al.* (2016) learn a joint embedding of videos and descriptions and use this representation to improve activity recognition on the Hollywood 2 dataset Marszalek *et al.* (2009). Tapaswi *et al.* (2016) use our AD transcripts for building their MovieQA dataset, which asks natural language questions about movies, requiring an understanding of visual and textual information, such as dialogue and AD, to answer the question. Zhu *et al.* (2015a) present a fill-in-the-blank challenge for audio description of the current, previous, and next sentence description for a given clip, requiring to understand the temporal context of the clips.

2.2 VIDEO DESCRIPTION APPROACHES

The first works on video description with natural language go back to Kojima *et al.* (2002). Early works typically employed manually defined templates or retrieval approaches to video description (Barbu *et al.*, 2012; Das *et al.*, 2013; Krishnamoorthy *et al.*, 2013). A few works proposed to generate novel sentences by means of learning the language models (Rohrbach *et al.*, 2013b). After we witnessed an explosive interest to image captioning around 2015 (Fang *et al.*, 2015; Karpathy and Fei-Fei, 2015), it consequently steered the interest to video description (Venugopalan *et al.*, 2015c; Yao *et al.*, 2015). Many recent approaches rely on recurrent neural networks to learn the language representation and to generate novel descriptions.

In the following we review prior work on video description, grouped according to the language generation mechanisms. We start with manually defined templates and grammars, then discuss retrieval approaches and, finally, the more recent "translation" approaches. We also review approaches to generate multi-sentence video descriptions. Afterwards we relate these prior works to the contributions of the thesis.

2.2.1 Manually defined templates and grammars

Most of the early works on video description rely on manually defined templates to generate sentences. Typically, as the first step, they detect certain visual concepts (e.g. Subject, Verb, Object) and then use these predictions to "fill in" the predefined templates. Other works manually define more sophisticated language grammars, typically limited to small vocabularies. Such approaches, although they might guarantee a perfect language grammar, are limited by the hand-designed rules and predefined visual concept detectors.

Kojima *et al.* (2002) tackle a surveillance setting, where a person is seen entering an office room and interacting with objects. They propose a manually defined action hierarchy modeled with the "case frames" (e.g. predicate, agent, location, object). Kojima *et al.* build a recognition system for the case frames based on body, head and hand movements. Lastly they apply a set of rules to translate the case frames into natural language sentences.

Tan *et al.* (2011) exploit a video as well as an audio channel to learn audio-visual concepts for three types of TRECVID 2010 Multimedia Event Detection (MED10) events (Over *et al.*, 2010). Such concepts can refer to human actions (e.g. walking, running), scenes (e.g. kitchen, crowd) and audio (e.g. cheering). Tan *et al.* generate sentences using the predefined templates based on the predicted concepts.

Barbu *et al.* (2012) extract human body pose and track objects in DARPA Mind's Eye Y1 (DARPA, 2011) videos. They use Hidden Markov Models (HMMs) to recognize human actions based on the extracted tracks. Based on a predicted action and associated tracks Barbu *et al.* employ templates to generate sentence descriptions.

Yu and Siskind (2013) propose a framework of learning the semantics of words through the video which also allows them to generate descriptions for new videos. Their approach, "Sentence Tracker", tracks multiple objects in a video mentioned in a sentence description. They employ HMMs to model verbs as well as other parts of speech that apear in sentences. Each part of speech is grounded in specific visual features. Finally, the sentence/video pairs are jointly scored to obtain the highest total likelihood. Siddharth *et al.* (2014) show how the "Sentence Tracker" can be applied to three video understanding tasks, namely sentence-guided focus of attention (tracking), video description generation, and video retrieval. The experiments are carried out on a small dataset with a vocabulary of 17 words, modeled with regular expressions or finite-state recognizers (FSMs). In their follow-up work, Yu and Siskind (2015a) investigate whether the information that events are *absent* in the video can benefit their approach. They provide the "Sentence Tracker" with "positive"

and "negative" sentences, and propose a discrimination score which ensures that the positive sentences are scored higher. Another difference from the work of Yu and Siskind (2013) is that here the word meaning is learned in a weakly supervised manner, without explicit word-to-video annotations.

Krishnamoorthy *et al.* (2013) predict Subject, Verb, Object (SVO) triplets for videos by relying on pre-trained object detectors (Felzenszwalb *et al.*, 2010) for subjects and objects as well as motion descriptions (Laptev *et al.*, 2008) for verbs. They also expand the set of detected verbs by including synonym verbs from WordNet (Fellbaum, 1998). Next, they score the obtained SVO triplets with respect to an SVO language model and generate multiple sentences for the best triplet with different templates. Finally, they choose one most likely sentence w.r.t. an n-gram language model. Guadarrama *et al.* (2013) scale the approach of Krishnamoorthy *et al.* (2013) by relying on stronger object detectors of Li *et al.* (2010) and Dense Trajectories (Wang *et al.*, 2013a). They also employ the "hedging-your-bets" strategy (Deng *et al.*, 2012) to predict more abstract descriptions in case of uncertainty. Additionally, the usage of external linguistic knowledge from web-scale textual corpora allows them to do "zero-shot" verb prediction, namely predicting verbs that were not seen during training. Thomason *et al.* (2014) rely on similar visual recognition as Guadarrama *et al.* (2013) but enhance it by integrating large scale object classifiers (Deng *et al.*, 2012) and scene classifiers (Xiao *et al.*, 2010). Thomason *et al.* use a factor graph to combine visual prediction for objects, activities and scenes with the language statistics mined from large corpora to estimate the most likely subject, verb, object, and place.

Sun and Nevatia (2014) propose a Semantic Aware Transcription (SAT) framework based on Random Forest classifiers. SAT uses object and action detection responses as input and models the semantic relationships between SVO labels. Specifically, it relies on a continuous skip-gram language model (Mikolov *et al.*, 2013) to group semantically similar words during training.

Xu *et al.* (2015b) jointly address the language generation and video/language retrieval tasks. They learn a joint embedding for a deep video model and a compositional semantic language model. Sentences are modeled by subject, verb, and object (SVO) triplets which are represented with Word2Vec (Mikolov *et al.*, 2013). The entire sentence representation is obtained via a recursive neural network. Sentences are generated with the SVO templates.

2.2.2 Retrieval based approaches

Another line of work approaches video description as a retrieval task, namely they retrieve sentences from the training set. Although this guarantees perfect grammar, such approaches can not compose novel descriptions and are limited to the descriptions in the training set, thus the retrieved descriptions are likely not to be entirely correct with respect to the test video.

Das *et al.* (2013) propose a system which consists of three parts: a low-level topic model which predicts keywords, a mid-level concept detectors and high-level

semantic verification. The high-level system ensures that the low-level detections are consistent with the mid-level concept predictions, and retrieves the most likely training sentence based on a ranking w.r.t. low- and mid-level predictions.

The recent winners of the TRECVID 2016 Video to Text Description challenge, Dong *et al.* (2016b), propose Word2VisualVec, a deep neural network model, which learns to match sentences to videos. Specifically, they project the language Word2Vec (Mikolov *et al.*, 2013) representation into a deep video feature space via a multilayer perceptron.

Kaufman *et al.* (2016) present a retrieval-based approach which took the first place in the LSMDC16 Movie Description track. The problem which they address is more general. They want to establish correspondences between test videos and reference videos with associated (task specific) semantics, so that the semantics is transferred to test videos. Special cases of such task are video description and video summarization. The nearest neighbor video is retrieved from the reference set via the unified space using Canonical Correlation Analysis (CCA). The CCA space is learned over visual and semantic features, while optimizing the semantics-appearance similarity and temporal coherency.

2.2.3 Translation approaches

Most recent approaches treat video description as a translation problem, i.e. they aim to learn the mapping between the two “languages", video and text. The first work to apply this idea to video description is by Rohrbach *et al.* (2013b), who propose a two-step learning approach. First they predict an intermediate semantic representation (SR) modeled with a CRF. Next, they use statistical machine translation (SMT) (Koehn *et al.*, 2007) to translate the SR to a sentence. The approach of Rohrbach *et al.* (2013b) learns from a parallel corpus of videos, low-level semantic annotations and sentence descriptions.

More recent works take inspiration from the encoder-decoder approaches to neural machine translation (Cho *et al.*, 2014; Sutskever *et al.*, 2014). If we apply the translation paradigm to video description, the encoder corresponds to a visual feature extractor, e.g. a convolutional neural network (CNN). The decoder, typically a recurrent neural network (RNN), or specifically, the Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997), generates sentence descriptions from the encoded visual representations. Inspired by this idea, Donahue *et al.* (2015) propose Long-term Recurrent Convolutional Networks (LRCN) to describe videos using LSTM. They rely on precomputed CRF scores of Rohrbach *et al.* (2014), thus still depending on low-level semantic annotations. Venugopalan *et al.* (2015c) extend this work to extract CNN features from frames which are mean-pooled over time, removing the need for low-level annotations. They show the benefit of pre-training the LSTM network for image captioning and fine-tuning it to video description. Later Venugopalan *et al.* (2015a) propose S2VT, an encoder-decoder framework, where a single LSTM encodes the input video frame by frame and decodes it into a sentence. This approach improves over the prior work which does mean-pooling of video

features over time. In their follow-up work, Venugopalan *et al.* (2016) explore the benefits of pre-trained word embeddings and language models for generation on large external text corpora.

To handle the challenging scenario of movie description, Yao *et al.* (2015) propose a soft-attention based model which selects the most relevant temporal segments in a video, incorporates 3D convolutional neural networks (3D-CNN) and generates a sentence using an LSTM. Ballas *et al.* (2016) leverage multiple convolutional maps from different CNN layers to improve the visual representation for activity and video description. Later Yao *et al.* (2016) conduct an interesting study on performance upper bounds for both image and video description tasks on available datasets, including our LSMDC dataset.

Shetty and Laaksonen (2015) evaluate different visual features as input for an LSTM generation framework. Specifically they use dense trajectory features (Wang *et al.*, 2013a) extracted for the clips and CNN features extracted at center frames of the clip. They find that training concept classifiers on MS COCO with the CNN features, combined with dense trajectories provides the best input for the LSTM. Later Shetty and Laaksonen (2016) propose to train an evaluator network to select the best caption from multiple models, trained with different feature combinations (at video frame level and video segment level).

Pan *et al.* (2016b) propose a Long Short-Term Memory with visual-semantic Embedding (LSTM-E) framework. It consists of mean pooled 2-/3-D CNN representations and LSTM trained jointly with a visual-semantic embedding to ensure better coherence between video and text. Yu *et al.* (2017b) jointly address video description, retrieval and fill-in-the-blank tasks. The main components of their approach are a concept word detector, which predicts a set of concepts for a given video, and an LSTM with semantic attention mechanism (over the word concepts), similar to the one of You *et al.* (2016).

Pan *et al.* (2016a) extend the video encoding idea by presenting a hierarchical recurrent neural encoder (HRNE) to exploit temporal structure of videos. HRNE introduces a second LSTM layer which receives input of the first layer, but skips several frames, reducing its temporal depth. Pan *et al.* also incorporate a temporal attention mechanism, similar to Yao *et al.* (2015). Baraldi *et al.* (2017) propose to leverage the hierarchical structure of the video by means of a novel time boundary-aware LSTM. The proposed LSTM can identify discontinuities (cuts) in the video and modify the temporal connectivity accordingly, e.g. by resetting the hidden state and memory. This enables the network to better adapt to the input data. Peris *et al.* (2016) propose to use Bidirectional Recurrent Neural Networks to model relationships in two temporal directions. Their model is based on the model of Yao *et al.* (2015). Peris *et al.* obtain expressive visual representations by combining CNNs for object detection and scene classification.

Multiple approaches have been proposed as part of the MSR Video to Language Challenge, based on the MSR-VTT dataset. Ramanishka *et al.* (2016) propose a video description model, which fuses multiple modalities (e.g. visual, audial), based on the S2VT model of Venugopalan *et al.* (2015a). They also include high level video

category information in the model (e.g. cooking, sports), available in the MSR-VTT dataset, and show that a "committee" of expert models for each category outperforms one joint model. Jin *et al.* (2016) focus on multi-modal fusion of visual, audio, speech and meta modalities. An LSTM is used to decode the visual representation into a sentence. Dong *et al.* (2016a) employ early embedding and late re-ranking for video description. Early embedding enriches the input to LSTM with automatically predicted video tags. Late reranking re-scores generated sentences with respect to tag matching or semantics to promote the most relevant captions.

Looking in a slightly alternative direction, Li *et al.* (2015) study the problem of summarizing a long video to a single concise description by using ranking based summarization of multiple generated candidate sentences. Specifically, they generate sentence descriptions for each video frame, construct an adjacency graph, connecting all pairs of sentences, and prune the graph with a ranking based summarization method, obtaining the final description.

2.2.4 Multi-sentence video description

Some works go beyond the "single clip - single sentence" scenario by producing multi-sentence video descriptions. Gupta *et al.* (2009) learn AND/OR graphs to capture the causal relationships of actions given visual and textual data. At test time they find the most fitting graph to produce template-based, multi-sentence descriptions. Khan *et al.* (2011) first produce multiple sentences and then rely on paraphrasing and merging to get the minimum needed number of sentences. Using a simple template, Tan *et al.* (2011) generate a sentence every 10 seconds of the video based on concept detections. They recognize a high level event and remove inconsistent concepts. To generate multiple sentences for a video, Das *et al.* (2013) segment the video based on the similarity of concept detections in neighboring frames but rely on manually defined verbs instead of predicting them.

More recently, Yu *et al.* (2016a) propose to use two stacked RNNs where the first models words within a sentence and the second models sentences within a paragraph. The paragraph generator takes the sentential embeddings produced by the sentence generator as input and assigns the new initial state to the sentence generator. Shin *et al.* (2016) generate story-like video descriptions by temporally segmenting the video with respect to the localized actions, and generating a consistent narrative (multiple sentences) by means of multiple natural language processing techniques. Specifically they perform backward coreference resolution and introduce connective words, such as "then".

2.2.5 Relations to our work

In this section we put our approach to multi-sentence and multi-level video description as well as the Visual-Labels approach to movie description in context of prior work.

Our video description approach, presented in Chapter 4, falls in the group of translation approaches. Although it builds upon the work of Rohrbach *et al.* (2013b), there are multiple differences between them. While Rohrbach *et al.* (2013b) generate single sentences, the focus of our work is to produce multi-sentence descriptions for an entire video at multiple levels of detail. In contrast to Rohrbach *et al.* (2013b) who rely on pre-segmented video snippets, we segment the video automatically. Furthermore, we exploit the probabilistic output of the CRF and incorporate it in SMT using a word-lattice (Dyer *et al.*, 2008).

Regarding multi-sentence video description, unlike prior works (Das *et al.*, 2013; Khan *et al.*, 2011; Tan *et al.*, 2011) we use agglomerative clustering of visual attribute classifiers trained to capture the desired granularity. Our visual attributes include fine-grained activities and participating objects. Similar to Tan *et al.* (2011), we recognize the high level event (in our case a composite activity) to make sure that the individual sentences are consistent. However, Tan *et al.* work in a much simpler setting of just 3 high level events with manually defined relations to all existing concepts.

We are not aware of any work in computer vision approaching descriptions at different levels of detail. There is some relation with the work of Guadarrama *et al.* (2013), who predict more abstract words if the uncertainty is too high for a more specific prediction. Our approach is complementary, as our goal is to produce descriptions at different levels of detail rather than to decrease uncertainty.

Our movie description approach, Visual Labels (Chapter 5), is another representative of translation approaches. Unlike most recent end-to-end approaches, we argue that the movie description task requires a targeted visual representation, learned on the movie domain, while pre-trained CNN representations from other domains might be not sufficient. Visual Labels builds on two ideas: the visual classifiers of Rohrbach *et al.* (2013b) and the LSTM decoder of Donahue *et al.* (2015). Unlike Rohrbach *et al.* (2013b), who rely on low-level semantic annotations of the video, we extract labels from sentence descriptions automatically, using our semantic parser (Chapter 5), focusing on three semantic groups of labels (verbs, objects and places). For sentence generation we rely on the LSTM implementation of (Donahue *et al.*, 2015), based on Caffe Jia *et al.* (2014). Additionally, we analyze different aspects and variants of this architecture for movie description, e.g. dropout strategies and placements and ensemble of multiple networks with different random initializations. Finally, our approach Visual Labels is ranked second in the LSMDC 2016 Movie Description challenge, according to human judges, only losing to the recently proposed approach of Kaufman *et al.* (2016).

2.3 VISUAL GROUNDING

Visual grounding is a rather broad term as discussed in the previous chapter. In this thesis we focus on two instances of the visual grounding problem. First, is the task of localizing natural language phrases in images and video (Plummer *et al.*, 2015)

and a very related task of referring expression comprehension (localization) (Mao *et al.*, 2016). Second, is the task of joint generation and grounding of descriptions for images and videos (Xu *et al.*, 2015a; Zanfir *et al.*, 2016). In the following we review the works addressing both directions. We conclude with relating the contributions of the thesis to prior work.

2.3.1 Grounding natural language in images and video

For grounding language in images, the approach of Kong *et al.* (2014) is based on a Markov Random Field which aligns 3D cuboids to words in textual descriptions of of RGB-D scenes. Their approach is limited to nouns of 21 object classes relevant to indoor scenes. Recently, Plummer *et al.* (2015) presented a new dataset, Flickr30k Entities, which augments the Flickr30k dataset (Young *et al.*, 2014) with bounding boxes for all noun phrases present in textual descriptions. Plummer *et al.* propose an approach based on a Canonical Correlation Analysis (CCA) (Gong *et al.*, 2014) embedding, learned from phrases and associated visual features. Later Plummer *et al.* (2016) improve their CCA approach with more proposal regions and additional object detectors as well as size and color features. Wang *et al.* (2016a) propose Deep Structure-Preserving Embedding for image-sentence retrieval; by formulating it as a ranking problem they also apply it to phrase localization. For each phrase they retrieve the closest image region using the learned embedding space.

The Spatial Context Recurrent ConvNet (SCRC) (Hu *et al.*, 2016b) uses an RNN-based caption generation framework to score the phrase on the set of proposal boxes, to select the box with the highest score. In addition to local descriptors from the proposal boxes, they take the global context and geometric configuration of the boxes into account. A similar approach is taken by Mao *et al.* (2016), who also score the proposal boxes based on the local and global visual features. Additionally to phrase localization (referred to as *comprehension* in their work), they also address the generation of referring expressions in the same framework. They also present a new dataset of images and localized referring expressions, based on MS COCO, named Google Refexp (GRef). Yu *et al.* (2016b) propose two more datasets for referring expression localization, RefCOCO and RefCOCO+, collected following Kazemzadeh *et al.* (2014). Both datasets are based on MS COCO images. While RefCOCO does not have any restriction on the language of referring expressions, the RefCOCO+ has one constraint: no location words can be used, thus expressions are more focused on appearance. Yu *et al.* also propose their approach to comprehension (localization) and generation of referring expressions. They integrate contextual features from other regions of the image to provide visual comparison of a target object to other objects.

A number of recent works state that it is necessary to look beyond single objects and also reason about relationships between object pairs. Following Wang *et al.* (2016a), Wang *et al.* (2016b) formulate a structured matching problem for phrases and image regions. They aim to fullfill two constraints: an image region can only be matched to one phrase and the relation between two phrases should result in

a similar relation for corresponding image regions. They only consider a subset of relations which include possessive pronouns. At the same time, Nagaraja *et al.* (2016) propose to ground referring expressions in two image regions: a target region and a context region. E.g. in the expression "a monitor above the keyboard" the "monitor" is a target, while the "keyboard" is the context. Their approach is closely related to the work of Mao *et al.* (2016), with a difference that not one but two regions are used as input. The context region is not known at training time, so it is discovered using multiple instance learning (MIL). Hu *et al.* (2017) propose Compositional Modular Networks (CMNs), a modular end-to-end architecture for referring expression localization. CMN decomposes the expression into parts (subject, relation, object) and grounds them in the image. The expression parsing is done with three respective soft attention maps. Two types of neural modules are used to perform word to image alignment: the localization module (to score the regions) and the relationship module (to score the region pairs).

Yu *et al.* (2017a) address referring expression generation and localization in a joint Speaker-Listener-Reinforcer framework. The idea is similar to the one by Andreas and Klein (2016), where a *speaker* aims to generate a discriminative caption for an image, so that a *listener* can distinguish it from another image. Here, the listener has to perform the grounding of the referring expression, and the additional reward-based *reinforcer* module guides the sampling for the speaker module.

Luo and Shakhnarovich (2017) propose to use a referring expression comprehension model to train a better generation model, by allowing the comprehension model to rerank the candidate expressions. Their comprehension model is based on our grounding approach, presented in Chapter 6. They introduce several changes, such as a different phrase representation (mean-pooled bidirectional LSTM initialized with Word2Vec (Mikolov *et al.*, 2013)), a dot product to combine visual and language representations, and an alternative loss formulation.

Zhang *et al.* (2016) study the top-down task-driven attention in CNNs. They propose Excitation Backprop, a back-propagation method, which follows the probabilistic Winner-Take-All formulation. Zhang *et al.* introduce the contrastive top-down attention by amplify the discriminative class-specific neurons, which helps them enhance the discriminativeness of the attention maps. They train a large scale tag classifier and and obtain attention maps for individual words from Flickr30k Entities test phrases. The averaged attention maps for each phrase are then used to score the region proposals.

A few works address visual grounding of linguistic structures other than natural language phrases. Johnson *et al.* (2015) use a Conditional Random Field (CRF) to ground scene graphs in images. The scene graphs capture image semantics, representing objects, their attributes, and relationships between them. They are used as a proxy between textual queries and images to perform image retrieval. Sadeghi *et al.* (2015) localize relation phrases of the type Subject-Verb-Object (SVO) at a large scale in order to verify their correctness, while relying on concept detectors (for S, O, SV, VO, SVO) from Divvala *et al.* (2014). Recently, Lu *et al.* (2016) detect visual relationships in a form of (object, predicate, object) with bounding boxes in images.

They rely on the RCNN object detector (Girshick *et al.*, 2014), a visual module that scores pairs of objects w.r.t. predicates, and a language model, which estimates a likelihood of relationships. In total they relate 100 objects with 70 predicates in their system. Karpathy *et al.* (2014a) ground sentence fragments, in a form of dependency-tree relations, to image regions, from a pre-trained object detector, using multiple instance learning and a ranking objective. Later Karpathy and Fei-Fei (2015) simplify this objective to just the maximum score and replace the dependency tree with a learned bidirectional recurrent network.

In the video domain some of the representative works on spatial-temporal language grounding are Yu and Siskind (2013) and Lin *et al.* (2014a). The approach of Yu and Siskind, who ground a sentence in the object tracks, was discussed in Section 2.2. Lin *et al.* ground individual words in a query to object tracks in a video. They build a semantic graph for the query and try to match it to objects detected in the video. Both works are limited to a small set of nouns.

2.3.2 Grounded image and video description

Recently, latent attention mechanisms have been explored for image and video description. The idea is to select (attend to) a subset of visual features while generating a word of a description. Xu *et al.* (2015a) first propose two variants of attention mechanisms to ground each word to spatial CNN image features. First is the deterministic *soft* attention, which weights multiple images regions according to the attention weights, second is the stochastic *hard* attention, which selects a single image region. Most follow-up works adopt the soft attention mechanism, which is also easier to train with standard back-propagation, while the hard attention includes non-differentiable sampling.

You *et al.* (2016) extend this approach to semantic attention over attributes. They run the learned attribute detectors on images. During caption generation they apply a soft-attention mechanism over the detected bounding boxes that represent attribute words. Recently, Yang *et al.* (2016b) extend the standard attentive encoder-decoder framework (Xu *et al.*, 2015a) to "encoder-reviewer-decoder" by introducing a *reviewer* module. This module updates the encoder hidden states with an attention mechanism, and produce global fact vectors which become input to the attention mechanism in the decoder.

Unlike others, Lu *et al.* (2017) argue that not all words in the image caption can and should be visually grounded (e.g. articles and prepositions). They introduce an adaptive attention model, which learns when to look at the image and when to rely on a language model during sentence generation. Their LSTM model is extended with an additional "visual sentinel" latent representation in the decoder and a respective gate, providing an option to either to rely on image or on the visual sentinel during decoding. They also propose a new spatial attention model, inspired by Residual Networks (He *et al.*, 2016).

In the video domain, Yao *et al.* (2015) apply the soft attention mechanism across video frames, allowing the description model to focus on relevant video segments, as

discussed earlier in Section 2.2. Zanfir *et al.* (2016) extend it to spatio-temporal object proposals in video. They also model video semantics with SVO triplets and represent videos as classifier responses over different S, V and O classes. Additionally they rely on the pre-trained object classifier (Simonyan and Zisserman, 2015) and detector (Ren *et al.*, 2015). All the different semantic visual representations are provided as input into LSTM, along with the spatio-temporally weighted visual representations (Simonyan and Zisserman, 2015).

Most recent works do not evaluate the correctness of the obtained localizations or attention maps, but exceptions exist. Liu *et al.* (2017) look into evaluating and improving attention correctness for image captioning. They propose an evaluation metric to measure the attention correctness, which captures the agreement between human annotations and automatic attention maps. They also propose multiple models which can integrate different forms of attention supervision, from explicit word-level localization supervision, to weak object class-level localization supervision. Ramanishka *et al.* (2017) propose an approach, Caption-Guided Visual Saliency, to analyze the mapping between spatial/spatio-temporal regions in images/video and words in captions, while the latter can be either generated or provided independently. They predict saliency maps for images and videos in a top-down fashion, based on the captions, exploiting the implicit dependencies captured by the LSTM. Saliency is estimated for each word by computing the decrease in probability of predicting this word, when given only a particular spatial/temporal region.

Johnson *et al.* (2016) take a different direction and build a model for dense captioning, which describes the entire image by jointly predicting large number of bounding boxes and a corresponding short phrase for each box. Lin *et al.* (2015a) parse the visual 3D scene into a scene graph, transform it into a sequence of semantic trees, and from these generate coherent multi-sentence descriptions, where the nouns are grounded in 3D cuboids.

2.3.3 Relations to our work

This section discusses our approaches to visual grounding and grounded video description with respect to prior work.

For the first problem, phrase localization, the main advantage of our approach GroundeR (Chapter 6) over prior work is its applicability to un- and semi-supervised training regimes (in terms of localization supervision). We believe that our approach of encoding the phrase optimizes the better objective for grounding than scoring the phrase with a text generation pipeline as e.g. done by Hu *et al.* (2016b) or Mao *et al.* (2016). Luo and Shakhnarovich (2017) base their localization approach on ours but introduce several modifications, like e.g. initializing language representation with pre-trained Word2Vec. As shown by Plummer *et al.* (2016), taking into account object size and color benefits grounding performance. We believe our approach would also benefit from such additional features. We also think that modeling context and relationships between objects, as done by e.g. Hu *et al.* (2017); Nagaraja *et al.* (2016); Wang *et al.* (2016a); Yu *et al.* (2016b), is beneficial for phrase localization, but we leave

this to future work.

Regarding different ways of combining language and visual representations, we propose Multimodal Compact Bilinear pooling (Chapter 7) and show its superiority to other methods, e.g. concatenation, elementwise product, for visual grounding and visual question answering. Although most neural approaches rely on representation concatenation (Hu *et al.*, 2016b; Mao *et al.*, 2016; Nagaraja *et al.*, 2016; Yu *et al.*, 2017a), alternatives exist. E.g. Luo and Shakhnarovich (2017) use a dot product, while Hu *et al.* (2017) use element-wise multiplication, followed by L2 normalization.

As for the second problem, grounded video description, our focus is to localize people in a video when we mention them in a generated sentence (Chapter 8). More specifically, we address four tasks jointly: description generation, people grounding, people local coreference resolution, and prediction of their gender. For that we employ a soft attention mechanism which jointly reasons about grounding and visual co-reference over people head tracks. Similar to Liu *et al.* (2017), we provide supervision to our attention mechanism, by automatically linking character mentions in text to visual tracks, for what we rely on our weakly supervised approach GroundeR (Chapter 6). The most related works are by Ramanishka *et al.* (2017); Zanfir *et al.* (2016), who also ground the words while generating video descriptions. The main difference is that they aim to ground all the words in a generated description, while we not only ground but also resolve local co-reference, and predict the gender of the described people. Unlike these works, we evaluate all the predictions of our approach, including grounding of people in video.

3 RECOGNIZING FINE-GRAINED AND COMPOSITE ACTIVITIES USING HAND-CENTRIC FEATURES AND SCRIPT DATA

Activity recognition has shown impressive progress in recent years. However, the challenges of detecting fine-grained activities and understanding how they are combined into composite activities has been largely overlooked. In this chapter we approach both tasks and present a dataset which provides detailed annotations to address them. The first challenge is to detect fine-grained activities, which are defined by low inter-class variability and are typically characterized by fine-grained body motions. We explore how human pose and hands can help to approach this challenge by comparing two pose-based and two hand-centric features with state-of-the-art holistic features. To attack the second challenge, recognizing composite activities, we leverage the fact that these activities are compositional and that the essential components of the activities can be obtained from textual descriptions or scripts. We show the benefits of our hand-centric approach for fine-grained activity classification and detection. For composite activity recognition we find that decomposition into attributes allows sharing information across composites and is essential to attack this hard task. Using script data we can recognize novel composites without having training data for them.

In Chapter 4 we address the video description task, using the proposed dataset.

3.1 INTRODUCTION

Human activity recognition in video is a fundamental problem in computer vision. State-of-the-art methods (e.g. Tang *et al.*, 2012; Wang *et al.*, 2013b; Wang and Schmid, 2013; Karpathy *et al.*, 2014b) achieve near perfect results for simple actions (e.g. KTH dataset, Schuldt *et al.*, 2004) and robustly recognize actions in realistic settings such as Hollywood movies (Marszalek *et al.*, 2009), videos from YouTube (Liu *et al.*, 2009), or sport scenes (Rodriguez *et al.*, 2008).

While impressive progress has been made, we argue that most works are addressing only a part of the overall activity recognition challenge. Many application scenarios, such as human-robot interaction or elderly care require to understand complex activities (e.g. *does the person prepare food?*), consisting of multiple fine-grained activities and object manipulations (e.g. *is it fried and what is in it?*). Frequently it is important to recognize both, the individual steps and the high level composite activities. Consequently we approach both problems in this chapter: recognizing fine-grained activities and recognizing composite activities. *Fine-grained activities* are defined as a set of activities which are visually very similar, i.e. have a low inter-class

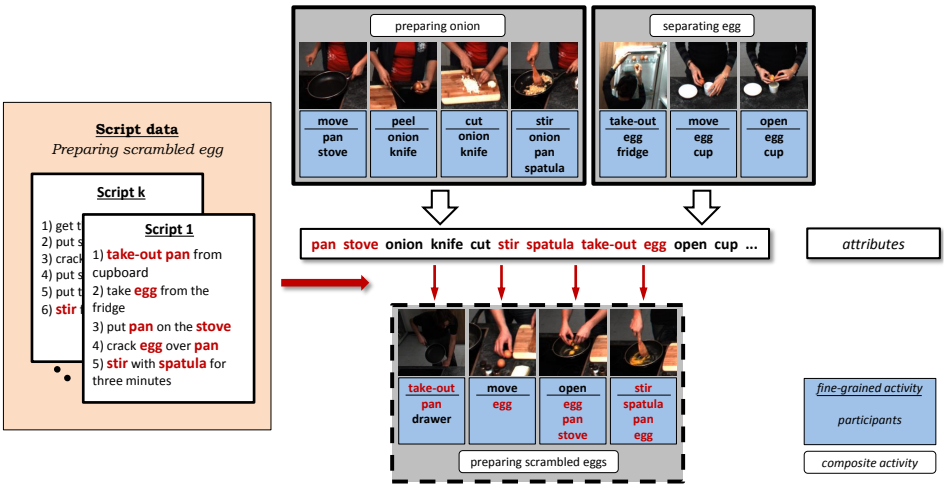


Figure 3.1: Sharing or transferring attributes of composite activities using script data. Composite activities (gray boxes) are composed of activities and their participants (light-blue boxes), modeled as attributes. These attributes can be transferred to unseen composite activities (dashed-line box) with the help of script data which allows estimating the relevant attributes (red). Our activities have the additional challenge of being fine-grained, we thus refer to them as fine-grained activities.

variability. *Composite activities* are activities which can be temporally decomposed into multiple shorter activities, i.e. they consist of multiple steps. We note that both the terms are not exclusive, i.e. composite activities can also be fine-grained. In fact some of our composites are very similar. However, in our work we consider composite activities which consist of fine-grained activities.

When surveying the field we also noticed a lack of datasets allowing to pursue the challenges of fine-grained and composite activity recognition. Specifically this is reflected in the following limiting factors of current benchmark databases. First, while datasets with large numbers of activities exist, the typical inter-class variability is high. This seems rather unrealistic for many domains such as surveillance or elderly care where we need to differentiate between consequentially different but visually similar activities e.g. *hug someone* versus *hold someone* or *throw in garbage* versus *put in drawer*. Second, the activities considered so far are full-body activities, e.g. *jumping* or *running*. This appears rather untypical for many applications where we want to differentiate between more small motion and frequently hand centric activities. Consider e.g. the *cutting activity* in domains such *cooking* (see Figure 3.1), *handicraft work* or *surgeries*, as well as different *repairing* activities in the domain of *house keeping* or *machine maintenance* with subtle difference in motion and low inter-class variability. As a third limitation we found that many available databases contain videos of few second length and focus on simple basic-level activities such as *walking* or *drinking*. In contrast, the recognition of longer-term, complex, and

composite activities such as *assembling furniture*, *food preparation*, or *surgeries* have been rarely addressed in computer vision. Notable exceptions exist (see Section 3.2) even though these have other limiting factors such as small number of classes.

We recorded, annotated, and publicly released a large dataset in a kitchen scenario which addresses the discussed limitations. It allows us to work on the challenges of fine-grained and composite activity recognition as follows.

Recognizing fine-grained activities is challenging due to their low inter-class variability. In contrast to fine-grained object recognition challenges where the same object category typically is also visually consistent, activities of the same category are frequently very diverse, i.e. have a high intra-class variability. Consider e.g. the activities *peeling*, which can be very different depending of the participating object: *peeling a carrot* versus *peeling a pineapple*. At the same time, we have to handle small differences between categories, i.e. low inter-class variability, consider e.g. *mix* versus *stir* or *slice* versus *cut dice*. This typically requires to understand the difference between fine-grained body motions. To approach both of these challenges we propose to focus on body pose and hands. As can be seen in Figures 3.1 and 3.2 many fine-grained activities, especially in our kitchen scenario, are hand-centric. Here it is not only important to understand the activity but also the participating object, e.g. *open egg* versus *open tin*. We thus propose to focus on the hand regions for extracting visual features. However, hand detection is a challenging problem in itself in real-world scenarios due to a large variability in shape and frequent partial occlusions (Mittal *et al.*, 2011; Gkioxari *et al.*, 2013). To get reliable hand detections, we integrate a hand detector into an articulated pose estimation. Consequently we use the hand position to extract color Sift and Dense Trajectories (Wang *et al.*, 2013a) and learn detectors for fine-grained activities and their participating objects. Recently, Jhuang *et al.* (2013) showed that exploiting body pose in form of body joints can be beneficial for full-body activities. We explore two approaches based on body pose tracks, motivated from work in the sensor-based activity recognition community (Zinnen *et al.*, 2009).

For recognizing composite activities, state-of-the-art methods, which build on discriminative learning from low-level activity features, experience scalability issues due to the typically highly diverse composite activities and little training data. A promising approach towards scaling activity recognition methods to a large number of complex activities is to use intermediate representations that are shared and transferred across activities by exploiting their compositional nature. We exploit this technique and propose building on an attribute-based representation, with attributes denoting the fine-grained activities and the participating objects. For example in Figure 3.1 the composite activity *preparing scrambled egg* shares the attributes *stir* and *spatula* with the composite activity *preparing onion* and the attributes *open* and *egg* with the composite activity *separating egg*. Instead of learning a holistic model for each composite activity we learn models for a large set of attributes shared across composite activity classes. Such approaches have been shown effective to recognize previously unseen object categories (Lampert *et al.*, 2013) and have also been applied to activity recognition (Liu *et al.*, 2011). A major challenge to recognize

everyday activities is that these composite activities can often be performed in a wide variety of ways, and it is practically infeasible to create a visually annotated training set with all possible alternatives. Instead, we collect a large number of textual descriptions (scripts) for composite activities to compute the association strength between attributes and composite activities. Using this script data we can handle the inherent variation of composites and even recognize unseen composite activities. As illustrated in Figure 3.1, the attributes in red are determined to be important for *preparing scrambled eggs* using script data and can be transferred from known composites such as *separating egg* and *preparing onion*.

Our main contributions are as follows. First, we propose several hand- and pose-based activity recognition approaches to recognize fine-grained activities and their object participants. We benchmark them together with state-of-the-art activity recognition features on our dataset. Second, we contribute an attribute-based approach which shares knowledge across composite activities and exploits textual script data to handle their large variability and allows transfer to unseen composite activities. Third, we recorded and annotated a video dataset called *MPII Cooking 2*. It provides challenges for classification and detection of fine-grained activities and their participants, human pose estimation, and composite activity recognition (optionally) using script data. In addition to activity recognition, which is the focus of our work, the dataset is also being used for 3D human pose estimation (Amin *et al.*, 2013), multi-frame pose estimation (Cherian *et al.*, 2014), discovering object categories from activities (Srikantha and Gall, 2014), grounding semantic similarities of natural language sentences in video (Regneri *et al.*, 2013), and for generating natural language descriptions (Rohrbach *et al.*, 2013b).

The remaining chapter is structured as follows. We first make an extensive review of related datasets, activity recognition approaches, and the use of text data for visual recognition in Section 3.2. Then we introduce our MPII Cooking 2 dataset in Section 3.3 which we benchmark in the subsequent sections. In Section 3.4 we make a quantitative comparison of our pose-recognition and hand detection with related work on the pose challenge of our dataset. Using the pose-estimation and hand detections we define several visual features and discuss fine-grained activity detection in Section 3.5. In Section 3.6 we present our approach to combine the fine-grained activities to composite activities and integrate script data. In Section 3.7 we evaluate fine-grained and composite activity recognition and then we conclude with the most important findings and directions for future work in Section 3.8.

3.2 RELATED WORK

We first present an overview of the different video activity recognition datasets (Section 3.2.1) and then review recent approaches to activity recognition (Section 3.2.2), putting a focus on works which use human pose as a cue. Next we discuss works which use textual information for improved recognition of activities (Section 3.2.3). We conclude by relating them to our work (Section 3.2.4).

3.2.1 Activity datasets

Even when excluding single image action datasets such as the Stanford-40 Action Dataset (Yao *et al.*, 2011b) or the Pascal Action Classification Challenge (Everingham *et al.*, 2011), the number of proposed activity datasets is quite large (Chaquet *et al.* (2013) survey 68 datasets). Here, we focus on the most important ones with respect to database size, usage, and similarity to our proposed dataset (see Table 3.1). We distinguish four broad categories of datasets: full body pose, movie and web, surveillance, and assisted daily living datasets – our dataset falls in the last category.

The full body pose datasets are defined by actors performing full body actions. KTH (Schuldt *et al.*, 2004), USC gestures (Natarajan and Nevatia, 2008), and similar datasets (Singh and Nevatia, 2011) require classifying simple full body and mainly repetitive activities. The MSR actions (Yuan *et al.*, 2009) pose a detection challenge limited to three classes. In contrast to these full body pose datasets, our dataset contains more and in particular fine-grained activities.

The second category consists of movie clips or web videos with challenges such as partial occlusions, camera motion, and diverse subjects. UCF50[1] and similar datasets (Liu *et al.*, 2009; Niebles *et al.*, 2010; Rodriguez *et al.*, 2008) focus on sport activities. Kuehne *et al.*'s evaluation suggests that these activities can already be discriminated by static joint locations alone (Kuehne *et al.*, 2011). UCF50 has been extended to UCF 101 (Soomro *et al.*, 2012), significantly increasing the number of categories to 101 and including 2.4 million frames at a rather low resolution of 320x240. The Sports-1M dataset exceeds all datasets with respect to number of clips (1.1 million) and categories (487 different sports), which are, however, only weakly labeled. Hollywood2 (Marszalek *et al.*, 2009), HMDB51 (Kuehne *et al.*, 2011), and ASLAN (Kliper-Gross *et al.*, 2012) have very diverse activities. Especially HMDB51 (Kuehne *et al.*, 2011) is an effort to provide a large scale database of 51 activities while reducing the database bias. Although it includes similar, fine-grained activities, such as *shoot bow* and *shoot gun* or *smile* and *laugh*, most classes have a large inter-class variability and the videos are low-resolution. ASLAN (Kliper-Gross *et al.*, 2012) focuses on a larger number of activities but with little training data per category. The task is to identify similar videos rather than categorising them. A significantly larger video collection is evaluated during the TRECVID challenge (Over *et al.*, 2012). The 2012 challenge consisted of 291h of short videos from the Internet Archive (archive.org) and more than 4,000h of multi-media (audio and video) data. The challenge covers different tasks including semantic indexing and multi-media event recognition of 20 different event categories such as *making a sandwich* and *renovating a home*. Large parts of the data are, however, only available to the participants during the challenge. Although our dataset is easier in respect to camera motion and background, it is challenging with respect to a smaller inter-class variability.

The datasets Coffee and Cigarettes (Laptev and Pérez, 2007) and High Five (Patron-Perez *et al.*, 2010) are different to the other movie datasets by promoting activity detection rather than classification. This is clearly a more challenging

[1] http://vision.eecs.ucf.edu/data.html

Dataset	cls,det	classes	clips /videos	subjects	# frames	resolution
Full body pose datasets						
KTH (Schuldt *et al.*, 2004)	cls	6	2,391	25	≈200,000	160x120
USC gestures (Natarajan and Nevatia, 2008)	cls	6	400	4		740x480
MSR action (Yuan *et al.*, 2009)	cls,det	3	63	10		320x240
Movie and web video datasets						
Hollywood2 (Marszalek *et al.*, 2009)	cls	12	1,707 /69			
UCF 101 (Soomro *et al.*, 2012)	cls	101	13,320		≈2,400,000	320x240
Sports-1M (Karpathy *et al.*, 2014b)	cls	487	1.1 mil			
HMDB51 (Kuehne *et al.*, 2011)	cls	51	6,766			height:240
ASLAN (Kliper-Gross *et al.*, 2012)	cls	432	3,631 /1,571			
Coffee and Cigarettes (Laptev and Pérez, 2007)	det	2	264 /11			
High Five (Patron-Perez *et al.*, 2010)	cls,det	4	300 /23			
Surveillance datasets						
PETS 2007 (Ferryman, 2007)	det	3	10		32,107	768x576
UT interaction (Ryoo and Aggarwal, 2009)	cls,det	6	120	6		
VIRAT (Oh *et al.*, 2011)	det	23	17		1920x1080	
Assisted daily living datasets						
TUM Kitchen (Tenorth *et al.*, 2009)	det	10	20 /4		36,666	384x288
CMU-MMAC (la Torre *et al.*, 2009)	cls,det	>130	26			1024x768
URADL (Messing *et al.*, 2009)	cls	17	150 /30	5	≤ 50,000	1280x720
MPII Cooking 2 (our dataset)	cls,det	67/ 59	14,105 /273	30	2,881,616	1624x1224

Table 3.1: Overview of activity recognition datasets: We list if datasets allow for classification (cls), detection (det); number of activity classes; number of clips extracted from full videos (only one listed if identical), number of subjects, total number of frames, and resolution of videos. We leave fields blank if unknown or not applicable.

problem as one not only has to classify a pre-segmented video but also to detect (or localize) an activity in a continuous video. As these datasets have a maximum of four classes, our dataset goes beyond these by distinguishing a large number of classes.

The third category of datasets is targeted towards surveillance. The PETS (Ferryman, 2007) or SDHA2010[2] workshop datasets contain real world situations from surveillance cameras in shops, subway stations, or airports. They are challenging as they contain multiple people with high partial occlusion. The UT interaction (Ryoo and Aggarwal, 2009) requires to distinguish 6 different two-people interaction activities, such as *punch* or *shake hands*. The VIRAT (Oh *et al.*, 2011) dataset is a recent attempt to provide a large scale dataset with 23 activities on nearly 30 hours of video. Although the video is high-resolution people are only of 20 to 180 pixel height. Overall the surveillance activities are very different to ours which are challenging

[2]http://cvrc.ece.utexas.edu/SDHA2010/

with respect to fine-grained hand motion.

Next we discuss the domain of *Assisted daily living (ADL) datasets*, which also includes our dataset. The University of Rochester Activities of Daily Living Dataset (URADL) (Messing *et al.*, 2009) provides high-resolution videos of 10 different activities such as *answer phone*, *chop banana*, or *peel banana*. Although some activities are very similar, the videos are produced with a clear script and contain only one activity each. In the TUM Kitchen dataset (Tenorth *et al.*, 2009) all subjects perform the same composite activity (*setting a table*) and rather similar actions with limited variation. Roggen *et al.* (2010) and la Torre *et al.* (2009) present recent attempts to provide several hours of multi-modal sensor data (e.g. body worn acceleration and object location). But unfortunately people and objects are (visually) instrumented, making the videos visually unrealistic. In the CMU-MMAC dataset (la Torre *et al.*, 2009) all subjects prepare the identical five dishes with very similar ingredients and tools. In contrast to this our dataset contains 59 diverse dishes, where each subject uses different ingredients and tools in each dish. The authors also record an egocentric view. Similarly to (Farhadi *et al.*, 2010a; Fathi *et al.*, 2011; Stein and McKenna, 2013) the camera view mainly shows hands and manipulated cooking ingredients. Also recorded in an egocentric view, Pirsiavash and Ramanan (2012) propose a dataset of 18 diverse daily living activities, not restricted to the cooking domain, recorded in different houses in non-scripted fashion.

Overall our dataset fills the gap of a large database with on the one hand a detection challenge of fine-grained activities and on the other hand a recognition challenge of highly variable composite activities.

3.2.2 Advances in activity recognition

Activity recognition for still images has been advanced e.g. by jointly modeling people and objects (Yao and Li, 2012) or scenes and objects (Li and Li, 2007). In the following we focus on recognizing activities in video, distinguishing three aspects: holistic features for activity recognition, exploiting body pose, and modelling the temporal structure of activities.

To create a discriminative feature representation of a video, many approaches first detect space-time interest points (Chakraborty *et al.*, 2011; Laptev, 2005) or sample them densely (Wang *et al.*, 2009a) and then extract diverse descriptors in the image-time volume, such as histograms of oriented gradients (HOG) and histograms of oriented flow (HOF) (Laptev *et al.*, 2008) or local trinary patterns (Yeffet and Wolf, 2009). Messing *et al.* (2009) found improved performance by tracking Harris3D interest points (Laptev, 2005). The state-of-the-art Dense Trajectories approach from Wang *et al.* (2013a) uses this idea: it tracks dense feature points and extracts strong video features around these tracks, namely HOG, HOF, and Motion Boundary Histograms (MBH, Dalal *et al.*, 2006). They report state-of-the art results on several datasets including KTH (Schuldt *et al.*, 2004), UCF YouTube (Liu *et al.*, 2009), Hollywood2 (Marszalek *et al.*, 2009), and HMDB51 (Kuehne *et al.*, 2011). Recently, Wang and Schmid (2013) improved their approach by removing

background flow and by ensuring that detected humans do not contribute to the background motion estimation. Additionally they replace the BoW encoding with Fisher vectors. The computational effort of this approach can be significantly reduced by replacing dense flow with motion information from video compression (Kantorov and Laptev, 2014). As alternative to manually defined activity features, Taylor *et al.* (2010), Baccouche *et al.* (2011), Le *et al.* (2011), and Ji *et al.* (2013) use deep learning with convolutional neural networks to learn an activity feature representation. So far these approaches cannot reach the manually defined Dense Trajectories even when learning on a database of over a 1 million videos (Karpathy *et al.*, 2014b).

Human body poses and their motion frequently characterize human activities and interactions. This has been exploited in Microsoft's Kinect, which uses human pose as a game controller but relies on a depth sensor to recognize human pose (Shotton *et al.*, 2011). Earlier work in human pose based activity recognition employed motion capture systems using physical on-body markers to reliably capture human poses (Campbell and Bobick, 1995). Such an approach is impractical for recording realistic data. Recently a number of hand and pose-centric approaches have been proposed for activity recognition for more realistic video recordings (Fathi *et al.*, 2011; Packer *et al.*, 2012; Yao *et al.*, 2011a; Sung *et al.*, 2011; Raptis and Sigal, 2013; Jhuang *et al.*, 2013) as well as in static images (Yang *et al.*, 2011; Yao and Li, 2012). Packer *et al.* demonstrate impressive results in recognition of kitchen activities using body poses recovered from depth images. Fathi *et al.* (2011) propose a hand-centric approach for learning effective models of activities from egocentric video by observing regularities in hand-object interactions. Hand poses have been shown to facilitate extraction of appearance features for activity recognition in static images (Karlinsky *et al.*, 2010). Pose-based models are effective for activity recognition when body poses can be estimated reliably, as e.g. in depth images (Packer *et al.*, 2012; Sung *et al.*, 2011). Mittal *et al.* (2011) and Gkioxari *et al.* (2013) aim for specialized representations for hands, but do not apply them to pose estimation or activity recognition. Jhuang *et al.* (2013) study the benefits of pose estimation for activity recognition on a subset of the HMDB dataset (Kuehne *et al.*, 2011). They show that ground truth pose, estimated over time can significantly outperform the holistic Dense Trajectories features (Wang *et al.*, 2013a); this is also true for estimated pose using (Yang and Ramanan, 2013) but only on a subset where the full body is visible.

Although several interesting techniques have been proposed to model the temporal structure of videos, they typically perform only below or on par with bag-of-word based approaches: A simple temporal structure is encoded in the template-based Action MACH by Rodriguez *et al.* (2008), Brendel and Todorovic (2011) model temporal and spatial structure by segmenting the space-temporal volume, and Niebles *et al.* (2010) model activities as a temporal composition of primitive actions and discriminatively learn such models. While Niebles *et al.* fix anchor points and the length of the temporal segments before training, Tang *et al.* (2012) learn all parameters from data using a variable-duration hidden Markov model. An AND/OR graph structure can be used to combine different features at its nodes (Tang *et al.*, 2013) or model co-occurring and consecutive actions (Gupta *et al.*, 2009). Recently Pirsiavash and

Ramanan (2014) have shown how to efficiently parse activity videos with segmental grammars.

3.2.3 Natural language text for activity recognition

Natural language descriptions have shown beneficial for image segmentation (Socher and Fei-Fei, 2010) or recognizing object categories (Wang *et al.*, 2009b; Elhoseiny *et al.*, 2013). Similar to our work, Elhoseiny *et al.* use classifiers trained on the known classes. Representing the text descriptions with tf∗idf (term frequency times inverse document frequency) vectors for relevant encyclopedic entries, they compare a regression, a domain adaptation, and a newly proposed constrained optimization formulation to learn a function from the textual vector to the visual classifier space. On two fine-grained visual recognition datasets, CU200 Birds (Welinder *et al.*, 2010) and Oxford Flower-102 (Nilsback and Zisserman, 2008), they show the benefit of their constraint optimization approach. Semantic similarity from linguistic resources has also been used to allow zero-shot recognition in images via attributes and direct similarity (Rohrbach *et al.*, 2010) and by learning an embedding into a linguistic word vector space (Socher *et al.*, 2013; Frome *et al.*, 2013). Additionally to transferring knowledge one can exploit the unlabeled instances to improve recognition, assuming a transductive setting. For this, Fu *et al.* (2013) exploit the test-data distribution by performing a single round of self-training by averaging over the k-nearest neighbors.

Teo *et al.* (2012) improve activity recognition by adding object detectors, which are selected based on the linguistic co-occurrence statistics in the newswire Gigaword Corpus. A similar idea is pursued by Motwani and Mooney (2012), who mine and cluster verbs from descriptions of the video snippets in the MSVD dataset (Chen and Dolan, 2011). Zhang *et al.* (2011) show that tf∗idf can identify the most relevant terms in text descriptions collected for seven video scenes allowing to yields close to perfect (98%) recognition accuracy on their dataset. Ramanathan *et al.* (2013) jointly recognize actions and roles in YouTube videos using their captions. They mine a large number of YouTube descriptions and use a topic model to estimate the semantic relatedness between an action/role and a description.

Another line of work focuses on describing videos with natural language descriptions. Recently Guadarrama *et al.* (2013) generated simple sentences for the Microsoft Video Description corpus (Chen and Dolan, 2011) containing challenging web videos. Das *et al.* (2013) compose descriptions for kitchen videos of their YouCook dataset showing YouTube cooking videos. Finally, (Rohrbach *et al.*, 2013b) have shown how to learn a translation model for generating natural sentences on our dataset.

3.2.4 Relations to our work

Most of the activity recognition approaches and datasets have been evaluated on full-body motion or challenging web or movie datasets but not on fine-grained motions with low inter-class variability. We therefore evaluate the holistic Dense

Trajectories approach from Wang *et al.* (2013a) as well as two pose-based and two hand centric approaches on our MPII Cooking 2 dataset. Our pose-based approach encodes trajectories of body joints using features motivated from the sensor-based activity recognition community (Zinnen *et al.*, 2009). The features are also similar to the relational and distance features defined on joints by Jhuang *et al.*. Similarly to their work we define relational and distance metrics between joints per frame and over time. However, our activities contain very subtle motions and the people have a very similar pose for most activities, which reduces the benefits of this feature representation. Jhuang *et al.* examine the advantages of focusing Dense Trajectories (Wang *et al.*, 2013a) on body joints. In our static scene (holistic) Dense Trajectories are already restricted to human body as the features are only extracted on moving points. However, in this work we propose to focus on hands, as they are the main cue for recognizing our fine-grained activities and participating objects.

Amin *et al.* (2013) show how to improve the hand localization by leveraging multiple cameras to handle self-occlusion. Instead, we remain monocular and propose to use a specialized hand detector to improve pose estimation and activity recognition.

To improve fine-grained activities and their participating objects we train a classifier on stacked classifier scores from co-occurring activities/objects as well as from temporal context after max pooling. Classifier stacking has previously been explored e.g. by Ting and Witten (1997); Liu *et al.* (2012); Sill *et al.* (2009). Most relevant to our work, Liu *et al.* (2012) try to optimize the usage of training data and avoid over-fitting when learning stacked video classifiers. This could be beneficial when applied to our approach.

In this chapter we exploit cooking instructions (script data) to extract which activities, tools, and ingredients are relevant for a certain dish (composite activity). For this we compare co-occurrence statistics with tf$*$idf, which has also been used by Zhang *et al.* (2011) and Elhoseiny *et al.* (2013) to extract relevant concepts for video scene and object recognition. We find that tf$*$idf better discriminates different dishes and improves performance in most cases. Script data allows for zero-shot recognition, which has mainly been used for object recognition, but also for multimedia data by Fu *et al.* (2013). Fu *et al.* learn a latent attribute representation on the known classes, but then use manually defined attribute associations to transfer.

While the temporal structure, i.e. temporal ordering, seems an important component to recognize activities, so far mainly the short term structure of short video clips has been explored (e.g. Gupta *et al.*, 2009; Brendel and Todorovic, 2011; Tang *et al.*, 2012). Here we exploit temporal co-occurrence within the same time interval and context of short actions and their participating objects within the entire video using max pooling. For long term composite activities we aggregate its components with max pooling ignoring the temporal order. Nevertheless, we believe that the temporal structure of scripts (Regneri *et al.*, 2010) might form a good prior for the temporal structure of videos and vise-versa. Bojanowski *et al.* (2014) have recently shown the benefit of movie scripts as a weak supervision. They use the ordering constraints provided by the script data to localize the actions and to learn action

models.

Finally we shortly summarize how this chapter extends the prior work by Rohrbach *et al.* (2012a) and Rohrbach *et al.* (2012b). First, we updated the dataset by correcting and unifying some of the annotations and adding a few more videos. We refer to this new version as MPII Cooking 2. It supersedes both previous datasets, see Table 3.3. Second, we integrated Propagated Semantic Transfer (PST) of Rohrbach *et al.* (2013b) for composite activity recognition. Specifically, this thesis contributes with the hand-centric approaches for fine-grained recognition, namely an integration of pose-estimation and hand detector, and Hand centric features for activity recognition. We also extended qualitative and quantitative results and rerun experiments with updated version of Dense Trajectories (Wang and Schmid, 2013).

3.3 DATASET "MPII COOKING 2"

For our dataset we video-recorded human subjects cooking a diverse set of dishes, e.g. *making pizza* or *preparing cucumber*. The dishes form the *composite activities* and the individual steps taken are the *fine-grained activities*, e.g. *cut*, *pour*, or *spice*. All videos have a composite label and are annotated with time intervals. Each time interval has a fine-grained activity and the participating objects as labels. A subset of frames was annotated with human pose and hands. In the following we provide details and statistics of the dataset, Figures 3.1 and 3.2 show example frames of the dataset.

3.3.1 Dataset statistics and versions

We recorded 30 subjects in 273 videos with a total length of more than 27 hours or 2,881,616 frames. Each video contains a single subject preparing a certain dish.

The dataset was recorded in two batches. The first part contains few, but very diverse and complex dishes (see upper part of Table 3.2) and was presented in (Rohrbach *et al.*, 2012a). The second part, presented by (Rohrbach *et al.*, 2012b), focuses on composite activities and thus contains significantly more dishes/composites which are slightly shorter and simpler, see lower part of Table 3.2. The second set of composite activities are selected according to our script corpus which we describe below in Section 3.3.4. We ignored some of them which were either too elementary to form a composite activity (e.g. *how to secure a chopping board*), were duplicates with slightly different titles, or because of limited availability of the ingredients (e.g. *butternut squash*).

For this work we corrected and unified some of the annotations and added a few more videos. We refer to this new dataset version as MPII Cooking 2. It supersedes both previous datasets. Table 3.3 compares the different versions and shows different statistics about them. The table also shows the proposed training/validation/test split, which is selected in a way that for all 31 composite activities in the test set, there are at least 3 training/validation videos and there is no overlap between training,

MPII Cooking	sandwich, salad, fried potatoes, potato pancake, omelet, soup, pizza, casserole, mashed potato, snack plate, cake, fruit salad, cold drink, and hot drink
MPII Composites	**cooking pasta**, juicing {**lime**, **orange**}, making {**coffee**, **hot dog**, **tea**}, pouring beer, preparing {asparagus, **avocado**, **broad beans**, broccoli and cauliflower, **broccoli**, carrots and potatoes, **carrots**, **cauliflower**, **chilli**, **cucumber**, **figs**, **garlic**, **ginger**, **herbs**, **kiwi**, **leeks**, **mango**, **onion**, **orange**, peach, peas, **pepper**, **pineapple**, **plum**, **pomegranate**, **potatoes**, **scrambled eggs**, spinach, spinach and leeks}, **separating egg**, sharpening knives, **slicing loaf of bread**, using {microplane grater, pestle and mortar, speed peeler, **toaster**, tongs}, zesting lemon

Table 3.2: Composite activities (dishes) of MPII Cooking 2 dataset, composites marked in bold are part of the test split.

	videos	subjects	categories composites	categories attributes	ground truth time intervals	attribute instances	video duration
MPII Cooking	44	12	14	218	3,824	15,382	3-41 min
MPII Composites	212	22	41	218	8,818	33,876	1-23 min
combined	256	30	55	218	12,642	49,258	1-41 min
MPII Cooking 2	273	30	59	222	14,105	54,774	1-41 min
- Training set	201	24	58	222	10,931	42,619	1-41 min
- Validation set	17	1	17	107	445	1,662	1-8 min
- Test set	42	5	31	169	2,102	8,023	1-13 min

Table 3.3: Dataset statistics. Note that the train/val/test split do not add up to the full dataset, as some videos of the test subjects are not used as they have less than three train/val videos.

validation, and test subjects. In contrast to the earlier versions we avoid multiple test splits for simpler evaluation and to reduce the computational burden for other researchers evaluating on the dataset.

3.3.2 Dataset recording and annotation protocol

To record realistic behavior we neither asked subjects to perform certain activities nor to follow a certain recipe but we told them only which dish they should prepare. This resulted in a larger variety of how subjects prepared things. This means subjects used different tools for preparation (*knife* or *peeler* for *peeling*), took different steps (e.g. some people cooked the vegetables some did not), and did things in different temporal orders for the same dish (e.g. *washed* the vegetable before or after they *peeled* it). Before the recording the subjects were shown our kitchen and places of tools and ingredients to feel at home. During the recording subjects could ask questions in case of problems and some listened to music. We always started the recording with an empty and clean kitchen, prior to the subject entering the kitchen and ended it once the subject declared to be finished, i.e. we did not include the final cleaning process. Most subjects were university students from different disciplines recruited by e-mail and publicly posted flyers. Subjects were paid per hour and

1. get a large sharp knife	1. gather your cutting board and knife.	1. wash the cucumber
2. get a cutting board	2. wash the cucumber.	2. peel the cucumber
3. put the cucumber on the board	3. place the cucumber flat on the cutting board.	3. place cucumber on a cutting board.
4. hold the cucumber in your weak hand	4. slice the cucumber horizontally into round slices.	4. take a knife and rock it back and forth on the cucumber
5. chop it into slices with your strong hand		5. make a clean thin slice each time.

Table 3.4: Three example scripts for the composite activity *preparing cucumber*.

cooking experience ranged from beginner cookers to amateur chefs.

Composite activities are annotated on the level of each video. Fine-grained activities were annotated with a two-stage revision phase with start and end frame using the annotation tool Advene (Aubert and Prié, 2007). In addition to the activity category each annotation consists of used tools, ingredients, and locations (we refer to them as participants). Composite activities were chosen as described in Sections 3.3.1 and 3.3.4. Activity, tool, ingredient, and location categories were chosen to describe all activities the human subjects were performing. The decision was made after the recording on the base what the human subjects did. With respect to the level of detail, we do not annotate the specific motions (e.g. move arm up or down) but what effect or semantic they have (e.g. open versus close). See Table 3.7 for the chosen granularity.

We recorded in our kitchen (see Figure 3.2(a)) with a 4D View Solutions system using a Point Grey Grasshopper camera with 1624x1224 pixel resolution at 29.4fps and global shutter. The camera is attached to the ceiling, recording a person working at the counter from the front. We provide the sequences as single frames (jpg with compression set to 75) and as video streams (compressed weakly with mpeg4v2 at a bit-rate of 2500). For most videos we recorded 7 additional camera views on the kitchen, a subset was used and released by Amin *et al.* (2013). Although they are not used in this work we will make the remaining 7 views available upon publication. All fine-grained and composite activity annotations are also valid for the other cameras as each frame was synchronized across all 8 cameras.

We also provide intermediate representations of holistic video descriptors, human pose detections, tracks, and features defined on the body pose. We hope this will foster research at different levels of activity recognition.

The dataset provides furthermore human body pose annotations (see Section 3.3.3), script data (see Section 3.3.4) and there exist textual descriptions in the TACoS (Regneri *et al.*, 2013) and TACoS Multi-Level corpus (Chapter 4). The descriptions in TACoS describe what happens in a specific video and are temporally aligned to the video, i.e. they provide a textual annotation. In contrast, the scripts used in this work are collected independently of the video and thus contain domain or script knowledge, i.e. what activities and what objects are likely used for a certain dish. As

Figure 3.2: Single frames from the dataset depicting fine-grained cooking activities and diverse sets of tools and ingredients (participants). (a) Full scene of *slicing* in the composite activity *omelet*, and crops of (b) *take out*, (c) *dicing*, (d) *take out*, (e) *squeeze*, (f) *peel*, (g) *wash*, (h) *grate*.

they are not specific to the training videos they allow to transfer and generalize to novel test scenarios.

3.3.3 Pose challenge

A subset of frames have articulated human pose and hand annotations to learn and evaluate pose estimation approaches and hand detectors. For human pose we annotated the frames with right and left shoulder, elbow, wrist, and hand joints as well as head and torso. We have 2,994 frames of 10 subjects for training of pose annotation and an additional of 4,250 training images with hand points used for training the hand detector. For testing we sample 1,277 frames from all activities with 7 subjects as test set for the pose challenge. All training and test frames are from MPII Cooking (Rohrbach *et al.*, 2012a) and thus avoid an overlap with the test subjects and test composites in MPII Cooking 2.

3.3.4 Mining script data for composite activities

Linguistics and psychology literature knows prototypical sequences of certain activities as so-called *scripts* (Schank and Abelson, 1977; Barr and Feigenbaum, 1981). Scripts describe a certain scenario which corresponds to composite activities in our case. Scenarios (e.g. *eating in a restaurant*) are temporally ordered events (*the*

patron enters restaurant, he takes a seat, he reads the menu,...) and subjects (*patron, waiter, food, menu,...*). Written event sequences for a scenario can be collected on a large scale using crowd-sourcing (Regneri *et al.*, 2010). We make use of this method to collect scripts for our composite activities and assembling a large number of written sequences for each of those.

We collect natural language sequences similar to Regneri *et al.* (2010) using Amazon Mechanical Turk. For each composite activity, we asked the subjects to give tutorial-like sequential instructions for executing the respective kitchen task. The instructions had to be divided into sequential steps with at most 15 steps per sequence. We select 53 relevant kitchen tasks as composite activities by mining the tutorials for basic kitchen tasks on the webpage "Jamie's Home Cooking Skills"[3]. All those tasks/scenarios are about processesing ingredients or using certain kitchen tools. In addition to the data we collected in this experiment, we use data from the OMICS corpus (Singh *et al.*, 2002) and Regneri *et al.* (2010) for 6 kitchen-related composite activities. This results in a corpus with 59 composite activities and 2,124 sequences in sum, having a total of 12,958 individual event descriptions. Note that for practical reasons we only recorded videos for 35 of these composite activities as discussed in Section 3.3.1. They are listed in Table 3.2 under "MPII Composites".

This script corpus provides much more variation than the limited number of video training examples can capture. Of course this also poses a challenge, because we need to overcome the problem of different wordings and coordinated events: Table 3.4 shows three examples we collected for the composite activity *preparing cucumber*. They differ in verbalization (e.g. *slice*, *chop*, and *make a slice*) and granularity (*getting* something is often left out). Further, the sequences reflect different ways of preparing the vegetable, some include *peeling* it, some do not *wash* it, and so on. Some sentences contain conjugated events (*take a knife and rock it...*). While we clean the data to a certain degree by fixing spelling mistakes and resolving pronouns with the method from Bloem *et al.* (2012), we end up with both challenges and blessings of a noisy but big script corpus.

In Section 3.6.4 we will describe how we extract semantic relatedness from this data.

3.4 HAND DETECTION AND POSE ESTIMATION

One goal of this chapter is to investigate the applicability of state-of-the-art pose estimation methods in the context of activity recognition. Therefore, in this section we propose our new pose estimation method based on Andriluka *et al.* (2011) and benchmark it on our dataset together with state-of-the-art pose estimation methods. Another goal is to demonstrate the importance of hand-based features for recognizing activities and their participants. For this we need to localize hands, which is in itself a challenging task due to partial occlusions, obstruction by manipulated objects, and variability of hand postures. In order to achieve high quality hand localization we

[3] http://www.jamieshomecookingskills.com

leverage two complementary sources of information. We exploit the characteristic appearance of hands in order to train an effective hand detector. We then integrate observations from this detector in our pose estimation approach to take advantage of the context provided by the other body parts. As another finding, we show that localization of all body parts benefits significantly from our specialized hand detector.

In the following we introduce our hand detector (Section 3.4.1) and pose estimation method (Section 3.4.2) as well as how we combine them (Section 3.4.3). In Section 3.4.4 we evaluate our proposed approaches as well as state-of-the-art pose estimation methods on our dataset.

3.4.1 Hand detection based on local appearance

As a basis for our hand detector we rely on the deformable part models (DPM, Felzenszwalb *et al.*, 2010). We discuss several design choices in order to achieve best performance.

Detection of left and right hands. We aim for a hand detector that can correctly distinguish the left and right hand of a person. The rationale behind this is that for many activities left and right hands have different roles (e.g. for a cutting activity the dominant hand is typically holding a knife while the supporting hand is holding the object that is being cut). Further, we would like to avoid situations when two strong hypotheses for one of the hands are chosen over two hypotheses for both hands. We achieve this by dedicating separate DPM components to left and right hands and jointly training them within the same detector (see examples in Figure 3.3). Note that in contrast to the default setting mirroring is switched off in DPM. At test time we pick the best scoring hypothesis among the components corresponding to left and right hands.

Component initialization. We capture the variance of hand postures by decomposing the hands' appearance into multiple modes and representing each mode with a specific DPM component. We found that a rather large number of components is necessary to achieve good detection performance. We initialize the components by clustering the HOG descriptors of the training examples using K-means as in Divvala *et al.* (2012). The detection further improves by first clustering the training examples by hand orientation and then by HOG.

Body context. We improve the hand localization by augmenting the hand detector with the context provided by a person detector. We rely on the person detector to constrain the search for hands to the image locations within the extended person bounding box and also constrain the scale of the hands detector to the scale of the person hypothesis.

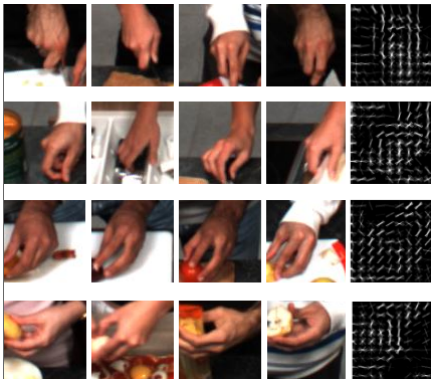


Figure 3.3: Examples of training images assigned to 4 different hand components, each row shows images from one component. Rows 1 and 2 correspond to right hand components, and rows 3 and 4 to left hand components.

3.4.2 Pose estimation

We base our pose estimation approach on the pictorial structures (PS) approach (Fischler and Elschlager, 1973; Felzenszwalb and Huttenlocher, 2005). In PS the body is represented as a collection of rigid parts linked via a set of pairwise part relationships. Unlike the original model we define a flexible variant of the PS model (FPS) that consists of $N = 10$ parts corresponding to head, torso, as well as left and right shoulders, elbows, wrists and hands. Denoting the configuration of parts as $L = l_1, \ldots, l_N$, and image observations as D, the posterior over the part configuration is given by

$$p(L|D) \propto \prod_{(i,j)\in E} p(l_i|l_j) \cdot \prod_{i=1}^{i=N} p(D|l_i), \tag{3.1}$$

where E is a set of connected part pairs. We build on the publicly available PS implementation from Andriluka *et al.* (2011). In this model the pairwise connections between parts form a tree structure, which permits efficient and exact inference. The pairwise terms represent the spatial relationships between part positions and are modeled as Gaussians with respect to relative position and orientation of parts. The appearance of individual parts is represented with boosted part detectors and shape context image features. Conceptually the formulation of Andriluka *et al.* (2011) is similar to flexible mixture of parts model (FMP, Yang and Ramanan, 2011). The FMP model represents appearance of each body part with a set of HOG templates. Pairwise terms are adapted depending on the particular template. Parameters of appearance templates and pairwise terms of the FMP model are jointly trained using max-margin objective. The model of Andriluka *et al.* (2011) relies on a single appearance template for all parts. Parameters of pairwise terms are estimated using maximum likelihood independently from appearance terms. We extend this model by incorporating color features into the part likelihoods by stacking them with shape context features prior to part detector training. We encode the color as a multidimensional histogram in RGB space using 10 bins for each color dimension

which results in 1000 dimensional feature vectors. We then concatenate color and shape context features and train boosted part detectors for each part using the combined representation. We use standard AdaBoost for training and rely on the same weak learners as in Andriluka *et al.* (2011).

3.4.3 Combining hand detection and pose estimation

We extend the image observations in Eq. 3.1 with detection hypotheses for left and right hands, which we obtain using the corresponding components of our hand detector. We denote the set of hand hypotheses produced by our hand detector by $H = \{(d_k, s_k)|k = 1, \ldots, K\}$, where d_k is the image position and s_k the detection score. Based on this sparse set of detections we obtain a dense likelihood map for the hand part l_h using a kernel density estimate:

$$p(H|l_h) = \sum_{k=1}^{K} w_k \exp(-\sigma^2 \|d_k - l_h\|^2), \tag{3.2}$$

where $w_k = s_k - m$ is a positive weight associated with each hand hypothesis computed by shifting the detection score by the minimal score value m. There is no specific upper/lower bound for the scores s_k, but since DMP relies on SVM formulation the scores tend to be centered around 0 with confident negative examples having score less than -1. In practice we set $m = -1$ and ignore all detections with a smaller score than m.

3.4.4 Evaluation: pose estimation and hand detection

We first evaluate the results on the upper-body pose estimation task. In order to identify the best 2D pose estimation approach we use our 2D body joint annotations (see Section 3.3.3). For evaluating these methods we adopt the PCP measure (percentage of correct parts) proposed by Ferrari *et al.* (2008). The results are shown in Figure 3.5. The first three lines compare three state-of-the-art methods: the cascaded pictorial structures (CPS, Sapp *et al.*, 2010), the flexible mixture of parts model (FMP, Yang and Ramanan, 2011) and the implementation of pictorial structures model (PS, Andriluka *et al.*, 2011), using their published pose models. Lines 4 and 5 show the models of Yang and Ramanan and Andriluka *et al.* retrained on our data. Overall the model of Andriluka *et al.* performs best, achieving 66.0 PCP for all body-parts. We attribute the improvement of PS over FMP to the following. The FMP model encodes different orientation of parts via different appearance templates, whereas the PS model uses a single template that is rotation invariant and is evaluated at all orientations. The FMP model has a larger number of parameters because appearance templates are not shared across different part orientations. A larger number of parameters means that it is easier to overfit the FMP model than the PS model. This could explain the performance differences after retraining on our data. It could also be that finer discretization of body part orientations in the PS model compared to

Method	Torso	Head	upper arm r	upper arm l	lower arm r	lower arm l	All
Original models							
CPS Sapp *et al.* (2010)	67.1	0.0	53.4	48.6	47.3	37.0	42.2
FMP Yang and Ramanan (2011)	63.9	72.1	60.2	59.6	42.1	46.7	57.4
PS Andriluka *et al.* (2009)	58.0	45.5	50.5	57.2	43.3	38.8	48.9
Trained on our data							
FMP Yang and Ramanan (2011)	79.6	67.7	60.7	60.8	50.1	50.3	61.5
PS Andriluka *et al.* (2009)	80.1	80.0	67.8	69.6	48.9	49.6	66.0
FPS	78.5	79.4	61.9	64.1	62.4	61.0	67.9
FPS + data	79.3	85.0	64.3	64.6	60.0	59.8	68.8
FPS + data + hand det	79.6	84.9	70.9	70.0	73.5	70.2	74.9
FPS + data + color	80.7	85.8	69.1	67.4	69.3	65.5	73.0
FPS + data + hand det + color	**81.3**	**86.1**	**72.4**	**71.3**	**74.4**	**70.3**	**75.9**

Table 3.5: 2D upper body pose estimation results on the "Pose Challenge" of our dataset. The numbers correspond to the "percentage of correct parts" (PCP).

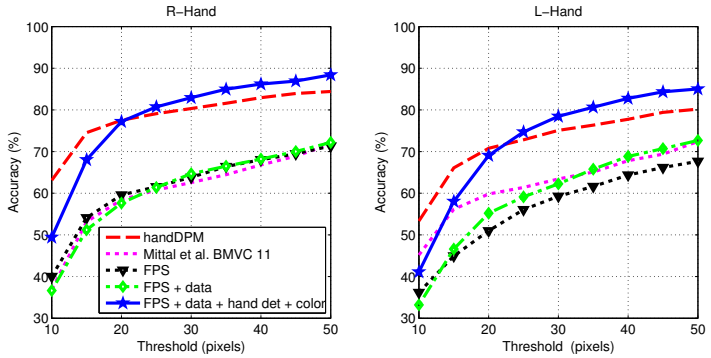


Figure 3.4: Accuracy of different methods for detection of right and left hands for a varying distance (in pixels) from the ground truth position.

the FMP model is important for good performance. As described above we base our model (FPS) on PS, adding to it flexible part configuration.

The bottom part of the Figure 3.5 shows that this as well as our other improvements (more training data comparing to Rohrbach *et al.* (2012a), color features, and hand detections) in the model each helps to improve performance. Overall, compared to PS, we achieve an improvement from 66.0 to 75.9 PCP and most notably an improvement from 48.9 to 74.4 and from 49.6 to 70.3 for lower arms, which are most important for recognizing hand-centric activities. We also would like to point to the benefit which hand detectors have to pose estimation (compare line 7 vs 8 and 9 vs 10).

Next we discuss the hand detection results. Our final hand detector *handDPM* is based on 32 components with 16 components allocated to each hand. The components are initialized by first grouping the training examples of each hand into 4

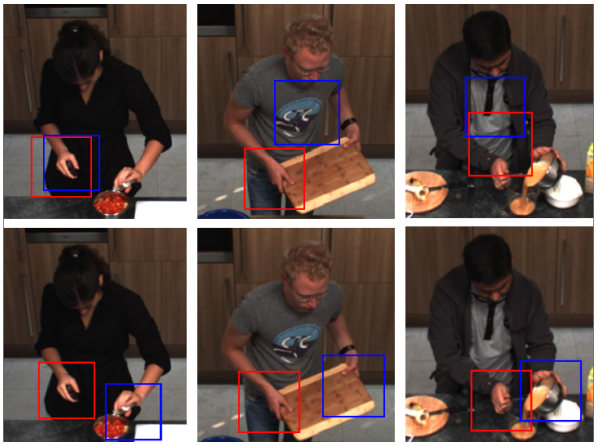


Figure 3.5: Pose helps to resolve failure cases of hand localization (upper row - handDPM, lower row is FPS+data+hand det+color).

discrete orientations, and then clustering their HOG descriptors. In the experiments on hand localization we use a metric that reflects the localization accuracy and measures the percentage of hand hypotheses within a given distance from the ground truth. We visualize the results by plotting the localization accuracy for a range of distances.

Figure 3.4 presents the evaluation of the localization accuracy of both hands. We observe that our hand detector (handDPM, red-dashed curve) alone already significantly improves over the proposed FPS approach (black-dotted-triangles). The performance further improves when hand detection hypotheses are integrated within the pose estimation model (blue-solid-stars). However, the improvement is moderate, likely because the pose estimation approach is not optimized specifically for hand detection and has to compromise between localization of hands and other body parts. Some qualitative examples are shown in Figure 3.5.

We also compare our hand detector to a state-of-the-art hand detector of Mittal *et al.* (2011) using the code made publicly available by the authors. We perform the best-case evaluation and assign the hand hypothesis returned by the approach to the closest left and right hand in the ground-truth, as the hand detector does not differentiate between left and right hands. For a fair comparison we also filter the hand detections of Mittal *et al.* (2011) at irrelevant scales and image locations using body context as explained before. Our detector significantly improves over the hand detector of Mittal *et al.* (2011), which in addition to hand appearance also relies on color and context features, whereas our hand detector uses hand regions only. Note that there are significant differences between localization accuracy of left and right hands. We attribute this to the fact that the majority of people in our database are right handed. Since people perform many activities with their dominant hand, the pose of the right hand is more likely to be constrained by various activities due to the use of tools such as a knife or peeler. The left hand's pose is far less deterministic and the hand is often occluded behind the counter or while holding various objects.

3.5 FINE-GRAINED ACTIVITY RECOGNITION AND DETECTION

In this section we focus on fine-grained activity recognition to approach the challenges typical e.g. for assisted daily living. Along with the activities we want to recognize their participating objects. To better understand the state-of-the-art for this challenging task we benchmark three types of approaches on our new dataset. The first type (Section 3.5.1) uses features derived from upper body model motivated by the intuition that human body configurations and human body motion should provide strong cues for activity recognition. For body pose estimation we rely on our approach described in Sections 3.4.2 and 3.4.3. The second type (Section 3.5.2) are the state-of-the-art Dense Trajectories (Wang *et al.*, 2013a) which have shown promising results on various datasets. It is a holistic approach in a sense that it extracts visual features on the entire frame. As the third type (Section 3.5.3) we present our hand-centric visual features, targeted at recognizing our hand-centric activities and the participating objects which are typically in the hand neighbourhood. For this we propose a hand detector (Sections 3.4.1, 3.4.3). Finally, we discuss our approaches to activity classification and detection in Section 3.5.4.

3.5.1 Pose-based approach

Pose-based activity recognition approaches were shown to be effective using inertial sensors (Zinnen *et al.*, 2009). Inspired by Zinnen *et al.* (2009) we build on a similar feature set, computing it from the temporal sequence of 2D body configurations.

We employ a person detector (Felzenszwalb *et al.*, 2010) and estimate the pose of the person within the detected region with 50% border around. This allows us to reduce the complexity of the pose estimation and simplifies the search to a single scale. To extract the trajectories of body joints we rely on search space reduction (Ferrari *et al.*, 2008) and tracking. To that end we first estimate poses over a sparse set of frames (every 10-th frame in our evaluation) and then track over a fixed temporal neighborhood of 50 frames forward and backward. For tracking we match SIFT features for each joint separately across consecutive frames. To discard outliers we find the largest group of features with coherent motion and update the joint position based on the motion of this group. This approach combines the generic appearance model learned at training time with the specific appearance (SIFT) features computed at test time.

Given the body joint trajectories we compute two different feature representations. First is a manually defined statistics over the body model trajectories, which we refer to as *body model features* (BM). Second is Fourier transform features (FFT) from Zinnen *et al.* (2009), which have shown effective for recognizing activities from body worn wearable sensors.

Body model features (BM). For the BM features we compute the *velocity* of all joints (similar to gradient calculation in the image domain). We bin it in an 8-bin

histogram according to its direction, weighted by the speed (in pixels/frame). This is similar to the approach by Messing *et al.* (2009) which additionally bins the velocity's magnitude. We repeat this by computing *acceleration* of each joint. Additionally we compute *distances* between the right and corresponding left joints as well as between all 4 joints on each body half. Similar to the joint trajectories (i.e. trajectories of x,y values) we build corresponding "trajectories" of distance values by stacking the values over temporally adjacent frames. For each distance trajectory we compute statistics (mean, median, standard deviation, minimum, and maximum) as well as a rate of change histogram, similar to velocity. Last, we compute the angle trajectories at all inner joints (wrists, elbows, shoulders) and use the statistics (mean etc.) of the angle and angle speed trajectories. This totals to 556 dimensions.

Fourier transform features (FFT). The FFT feature contains 4 exponential bands, 10 cepstral coefficients, and the spectral entropy and energy for each x and y coordinate trajectory of all joints, giving a total of 256 dimensions.

Feature representation. For both features (BM and FFT) we compute a separate codebook for each distinct sub-feature (i.e. velocity, acceleration, exponential bands etc.) which we found to be more robust than a single codebook. We set the codebook size to twice the respective feature dimension, which is created by computing k-means from all features (over 80,000). We compute both features for trajectories of length 20, 50, and 100 (centered at the frame where pose was detected) to allow for different motion lengths. The resulting features for different trajectory lengths are combined by stacking and give a total feature dimension of 3,336 for BM and 1,536 for FFT.

3.5.2 Holistic approach

Most approaches for activity recognition are based on a bag-of-words representations. We pick the state-of-the-art Dense Trajectories approach (Wang *et al.*, 2011, 2013a) which extracts histograms of oriented gradients (HOG), flow (HOF Laptev *et al.*, 2008), and motion boundary histograms (MBH Dalal *et al.*, 2006) around densely sampled points, which are tracked for 15 frames by median filtering in a dense optical flow field. The x and y *trajectory* speed is used as a fourth feature. Using their code and parameters which showed state-of-the-art performance on several datasets we extract these features on our data. Following Wang *et al.* (2013a) we generate a codebook for each of the four features of 4,000 words using k-means from over a million sampled features.

3.5.3 Hand-centric approach

In domains where people mainly perform hand-related activities it seems intuitive to expect that hand regions contain important and relevant information for recognizing those activities and the participating objects. Thus, in addition to using the holistic

and pose-based features, we suggest to focus on the hand regions. To obtain the hand locations we rely on our hand detector described in Section 3.4.1 as well as on the pose estimation method with integrated hand candidates (Section 3.4.3). In order to increase the robustness of the method we use both location candidates (provided by the handDPM detector and the final pose model) and sum the obtained features.

Hand-Trajectories. We want to represent different type of information: hand motion, hand shape, and shape variations over time, as well as the appearance of objects manipulated by the hands. We propose to densely sample the neighborhood of each hand and to track those points over time. For tracking and also representing the point trajectories with powerful features we adapt the approach of Wang *et al.* (2013a). We focus only on densely sampled points around the estimated hand positions instead of sampling the entire video frame. We specify a bounding box around each hand detection and densely sample points inside of it. In our experiment we use 120×140 pixels bounding box around hands to include the information about the hands' context. We use 8 pixels grid spacing for points sampling and finally we get 136 interest point tracks for each frame. After extracting the features along computed tracks we create codebooks that contain 4000 words per feature.

Hand-cSift. Color information is another important cue for recognizing activities and even more prominent for recognizing the participating objects. Similar to the previous approach we densely sample the points in the hands' neighborhood and extract color Sift features on 4 channels (RGB+grey). We quantize them in a codebook of size 4000.

3.5.4 Fine-grained activity classification and detection

Activity classification. Given a long video we assume that it consists of multiple time intervals. Each such interval t depicts a single fine-grained activity and its participating objects (e.g. *dry, hands, towel*). In the following we refer to both, activities and participants, as activity attributes $a_i, (i \in \{1,\dots,n\})$, i.e. a_i can be any attribute including *cut*, *knife*, or *cucumber*. We train one-vs-all SVM classifiers on the features described in the previous sections given the ground truth intervals and labels. The classifiers provide us with real valued confidence score functions $f_i^{base} : \mathbb{R}^N \mapsto \mathbb{R}$ for attribute a_i and feature vectors of dimension N. Combining different features is achieved by concatenating, i.e. stacking, the corresponding feature vectors.

Activity detection. While we use ground truth intervals for training the activity classifiers, we use a sliding window approach to find the correct interval of detection. To efficiently compute features of a sliding window we build an integral histogram over the histogram of the codebook features. We use non maximum suppression over different window lengths and start with the maximum score and remove all overlapping windows. In the detection experiments we use a minimum window

size of 30 with a step size of 6 frames; we increase window and step size by a factor of $\sqrt{2}$ until we reach a window size of 1800 frames (about 1 minute). Although this will still not cover all possible frame configurations, we found it to be a good trade-off between performance and computational costs.

3.6 MODELING COMPOSITE ACTIVITIES

In the previous section we discussed how we recognize fine-grained activities (such as *peeling* or *washing*) and their object participants (such as *grater*, *knife*, or *cucumber*). Now we focus on exploiting the temporal context and on recognizing different composite activities, e.g. *preparing a cucumber* or *cooking pasta*.

For this, we first show how we exploit temporal context and co-occurrence to improve the recognition of fine-grained activities and their object participants (Section 3.6.1). Then, we model composite activities as a flexible combination of attributes, where attributes refer jointly to the fine-grained activities and their object participants (Section 3.6.2). We then show how to use prior knowledge (Section 3.6.3) to improve the recognition of composite activities, overcoming the notorious lack of training data and handling the large variability of composite activities. In Section 3.6.4 we discuss how to mine the semantic relatedness from script data. Finally, in Section 3.6.5 we introduce an automatic approach to temporal video segmentation, which removes the necessity to manually annotate the ground truth intervals in a video.

3.6.1 Recognizing activity attributes using context and co-occurrence

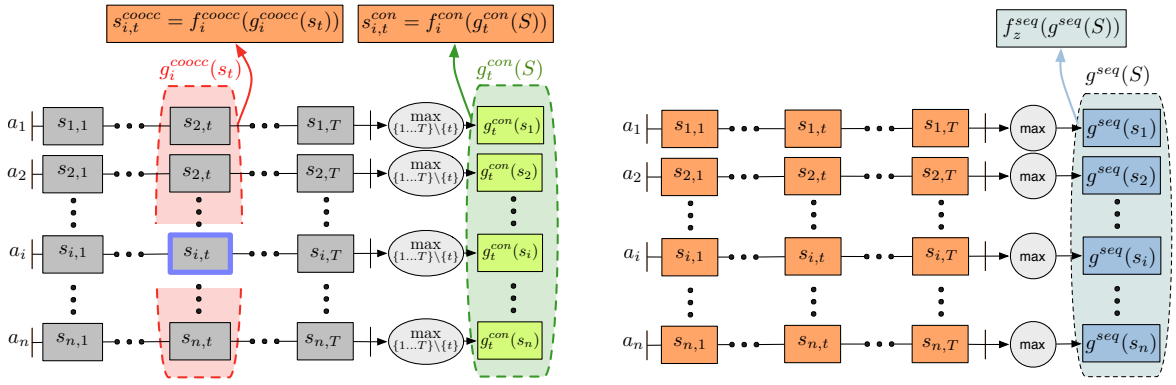
For a time interval t we want to classify if a particular fine-grained activity and its participants are present. We refer to activities and participants as activity attributes a_i. We distinguish three types of attribute classifiers. The first type of is given by the classifiers introduced in the previous section providing us with confidence score functions $f_i^{base} : \mathbb{R}^N \mapsto \mathbb{R}$ for each attribute a_i. Let us denote the score of a given feature vector x_t at time interval t as:

$$s_{i,t} = f_i^{base}(x_t). \tag{3.3}$$

Together these score constitute a matrix S of dimensions $n \times T$ (# attributes $\times$ #timestamps). Based on these scores, we define features for context (in the same video sequence) as well as features for co-occurrence of other attributes (in the same time interval t).

Contextual features formalize the intuition that adjacent time frames have strongly related attributes: e.g. if a *cucumber* is *peeled* in one time interval, then *cutting* the *cucumber* is probably also present in the same video sequence. As visualized in Figure 3.6(a) we define a context feature $g_t^{con} : \mathbb{R}^{n \times T} \mapsto \mathbb{R}^n$ at time t by max pooling the scores of each attribute over all time intervals except t:

$$g_t^{con}(S) = \max_{u \in \{1,\dots,T\} \setminus \{t\}} s_u \tag{3.4}$$



(a) Activity attribute recognition using **con**textual and **co-occ**urrence attributes vectors.

(b) Composite activity classification using max-pooled activity attributes.

Figure 3.6: Our approach to recognition of attributes (a) and composite activities (b).

where max is an element-wise operator over all columns $s_u \in \mathbb{R}^n$ of matrix S.

Similarly, activity attributes happening at the same time interval t are related, e.g. if we *peel* something it is more likely to observe also *carrot* or *cucumber* rather than *cauliflower*. We thus define the co-occurrence as a feature $g^{coocc}_i : \mathbb{R}^n \mapsto \mathbb{R}^{n-1}$ by stacking all attribute scores at time t excluding $s_{i,t}$:

$$g^{coocc}_i(s_t) = [s_{1,t}; ...; s_{i-1,t}; s_{i+1,t}; ...; s_{n,t}], \tag{3.5}$$

where $s_t \in \mathbb{R}^n$ is a column of matrix S.

Based on these features we train activity attribute SVM classifiers using the features individually or by stacking them. Specifically we obtain corresponding confidence score functions for context: $f^{con}_i : \mathbb{R}^n \mapsto \mathbb{R}$ and co-occurrence: $f^{coocc}_i : \mathbb{R}^{n-1} \mapsto \mathbb{R}$, where i denotes that a separate function for each attribute a_i is trained. We define corresponding scores as:

$$s^{con}_{i,t} = f^{con}_i(g^{con}_t(S)) \tag{3.6}$$

and

$$s^{coocc}_{i,t} = f^{coocc}_i(g^{coocc}_i(s_t)). \tag{3.7}$$

This formulation can be easily extended to other attribute representations depending on the task and available features.

3.6.2 Composite activity classification using activity attributes

We now want to classify composite activities that span an entire video sequence, given attribute classifier scores. We note that we can use any of the scores introduced in the previous section ($s_{i,t}$, $s^{con}_{i,t}$, $s^{coocc}_{i,t}$ or their stacked combination). In the following for simplicity we refer to these scores as $s_{i,t}$ and corresponding matrix as S. In this approach we rely on the representation that captures likelihoods of the presence

or absence of a particular attribute and leave modeling the temporal ordering of attributes for future work. We define a feature for the video sequence as g^{seq} : $\mathbb{R}^{n \times T} \mapsto \mathbb{R}^n$ by max pooling the scores of each attribute over all time intervals (see Figure 3.6(b)):

$$g^{seq}(S) = \max_{t \in \{1,\ldots,T\}} s_t \tag{3.8}$$

where max is an element-wise operator over all columns $s_t \in \mathbb{R}^n$ of matrix S.

To decide on the class z of a sequence d we use the feature g^{seq} and classify it using a nearest neighbor classifier (*NN*) or a one-versus-all SVM given a set of labeled training sequences. The SVM classifier provides us with the following confidence function for all composite classes z: $f_z^{seq} : \mathbb{R}^n \mapsto \mathbb{R}$, where the final score is defined as:

$$s_{z,d}^{seq} = f_z^{seq}(g^{seq}(S_d)), \tag{3.9}$$

where S_d is the score matrix for sequence d. The following sections describe alternatives to NN and SVM to incorporate prior knowledge mined from script data.

3.6.3 Script data for recognizing composite activities

Composite activities show a high diversity which is practically impossible to capture in a training corpus. Our system thus needs to be robust against many activity variants that are not present in the training data. The use of attributes allows to include external knowledge to determine relevant attributes for a given composite activity. For this we assume associations between attribute a_i and composite activity class z in a matrix of weights $w_{z,i}$, with Z being the number of composite activity classes. The vectors w_z are L1 normalized, i.e. $\sum_{i=1}^{n} w_{z,i} = 1$. Our system extracts those associations from script data (see Section 3.6.4), but the approach generalizes to other arbitrary external knowledge sources. We explore three options to use such information which we detail in the following.

Script data: We compute the confidence $f_z^{scriptdata} : \mathbb{R}^n \mapsto \mathbb{R}$ of a sequence being of the composite activity z using the attribute-based feature representation $g^{seq}(S)$ introduced in Equation (3.8). Given the weights $w_{z,i}$ we compute a weighted sum:

$$f_z^{scriptdata}(g^{seq}(S)) = \sum_{i=1}^{n} w_{z,i} g_i^{seq}(S). \tag{3.10}$$

For a specific sequence d with corresponding score matrix S_d we get the following score:

$$s_{z,d}^{scriptdata} = f_z^{scriptdata}(g^{seq}(S_d)). \tag{3.11}$$

This formulation is similar to the sum formulation used by Rohrbach *et al.* (2011) for image recognition with attributes, which itself is an adaption of the direct attribute prediction model introduced by Lampert *et al.* (2013). Note that the weight matrix retrieved from script data is sparse (most $w_{z,i} = 0$). When mining from other

corpora one might need to threshold the weights $w_{z,i}$, setting all others to zero, to achieve good performance as done e.g. by Rohrbach *et al.* (2011).

NN+script data: When training data is available we can use a nearest neighbor classifier. Often, only a handful of attributes are likely to be indicative for a composite activity class, while the majority of other attributes will provide irrelevant, potentially noisy information. When searching for nearest neighbors such irrelevant attributes might dominate the distance, resulting in suboptimal performance. To reduce this effect we rely on the script data to constrain the attribute feature vector to the relevant dimensions.

More specifically, we replace the L2 norm for computing the distance of nearest neighbor with the following training class dependent weighted L2 norm. It takes weights of class-attribute associations into account. It is defined between the test attribute vector of unseen class $g^{seq}(S_{test})$ and the training attribute vector $g^{seq}(S^z_{train})$ of class z as:

$$Dist(S_{test}, S^z_{train}) = \left(\sum_{i=1}^{n} w_{z,i} \left(g_i^{seq}(S_{test}) - g_i^{seq}(S^z_{train}) \right)^2 \right)^{0.5} . \tag{3.12}$$

To enhance robustness further, we binarize all association weights $w_{z,i}$ by setting all non-zero weights to 1 (and L1-normalize w_z). This reduces the distance computation to the relevant attributes, normalized by the total number of relevant attributes.

Propagated semantic transfer (PST): As the third approach to integrate external knowledge from script data we use Propagated semantic transfer (PST), proposed by Rohrbach *et al.* (2013a), and summarize shortly in the following. The approach builds on Equation (3.10) and uses label propagation to exploit the distances within the unlabeled data, i.e. it assumes a transductive setting where all test data is available when predicting a single test label.

We can incorporate (partially) labeled training data $l_{z,d} \in \{0, 1, \emptyset\}$ for class z and sequence d. $\emptyset$ denotes that we do not have a label for this sequence and class. We combine the labels with the predictions in the following way, using only the most reliable predictions $s^{scriptdata}_{z,d}$ (top-δ fraction) per class z:

$$s^{PST}_{z,d} = \begin{cases} \gamma l_{z,d} & \text{if } l_{z,d} \in \{0,1\} \\ (1-\gamma) s^{scriptdata}_{z,d} & \text{if among top-}\delta\text{ fraction} \\ & \text{of predictions for class } z \\ 0 & \text{otherwise.} \end{cases} \tag{3.13}$$

γ provides a weighting between the true labels and the predicted labels. In the zero-shot case we only use predictions and $\gamma = 0$. The parameters $\delta, \gamma \in [0, 1]$ are chosen, similar to the remaining parameters, on the validation set. For zero-shot we use the unlabeled training data as additional data for label propagation.

For computing the distance between the sequences we use the feature representation $g^{seq}(S)$, as for the *NN*-classifier, which is much lower dimensional than the

raw video feature representation and provides more reliable distances as shown by Rohrbach *et al.* (2013a). We build a k-NN graph by connecting the k closest neighbours. We set the weights of the graph edges between sequences d and e to $exp(-0.5\sigma^{0.5}\|g^{seq}(S_d) - g^{seq}(S_e)\|)$, where σ is set to the mean of the distances to the nearest neighbours. We initialize this graph with the scores $s^{PST}_{z,d}$ and propagate them using label propagation from Zhou *et al.* (2004).

3.6.4 Prior knowledge from script data

We want to quantify what activities and objects typically occur in a composite activity by leveraging the script data we collected (see Section 3.3.4). In order to use prior knowledge from textual script data, we have to match the (controlled) attribute labels from the video annotations to the (freely) written script instances (Section 3.6.4.1). Based on the matched attributes we compute two different word frequency statistics (Section 3.6.4.2).

3.6.4.1 *Label matching*

To transfer any kind of knowledge from the script corpus to the attributes in the video annotation, we need to match attribute labels to natural language descriptions. The annotated attribute labels are standard English verbs (for activities, *wash*) and nouns (for participating objects, *carrot*), sometimes with additional particles (*take apart* and *take out*). As the script instances contain freely written natural language sentences, they do not necessarily have any correspondence with the attribute label annotations. We compare two strategies for mapping annotations to script data sentences:

- **literal**: we look for the exact matching of the attribute label within the data.
- **WordNet**: we look for attribute labels and their synonyms. We take synonyms as members of the same *synset* according to the WordNet ontology (Fellbaum, 1998) and restrict them to words with the same part of speech, i.e. we match only verbal synonyms to activity predicates and only nouns to object terms.

3.6.4.2 *Statistics computed on the script data*

We compute two different association scores between attribute labels a_i and composite activities z. For this we concatenate all scripts for a given composite z to a single document δ_z.

- **freq**: word frequency $freq(a_i, \delta_z)$ for each attribute a_i and composite activity z.
- **tf∗idf** (term frequency $*$ inverse document frequency, Salton and Buckley, 1988) is a measure used in Information Retrieval to determine the relevance

of a word for a document. Given a document collection $D = \{\delta_1, ..., \delta_z, ..., \delta_m\}$, tf∗idf for a term or attribute a_i and a document δ_z is computed as follows:

$$tfidf(a_i, \delta_z) = freq(a_i, \delta_z) * log \frac{|D|}{|\{\delta \in D : a_i \in \delta\}|}, \tag{3.14}$$

where $\{\delta \in D : a_i \in \delta\}$ is the set of documents containing a_i at least once. tf∗idf represents the distinctiveness of a term for a document: the value increases if the term occurs often in the document and rarely in other documents.

We set $w_{z,i} = freq(a_i, \delta_z)$ or $w_{z,i} = tfidf(a_i, \delta_z)$ and L1-normalize all vectors w_z. These weights $w_{z,i}$ are then used in Equations (3.10) and (3.12) and subsequently also in the PST approach.

3.6.5 Automatic temporal segmentation

While we assume a segmented video during training time to learn attribute classifiers as described in Section 3.5.4, we want to segment the video automatically at test time. To avoid noisy and small segments we follow the approach discussed in more detail in Chapter 4, namely we employ agglomerative clustering. We start with uniform intervals of 60 frames and describe each interval with an attribute-classifier score vector. We combine neighbouring intervals based on the cosine similarity of their score vectors and stop when we reach a threshold (found on the validation set). We aim for a segmentation with granularity similar to original manual annotation. After this a separately trained visual background classifier removes irrelevant or noisy segments. In our experiments we show that this leads to composite recognition results, similar to using the ground truth intervals for the attributes.

3.7 EVALUATION

In this section we evaluate our approaches to fine-grained and composite activity recognition. We start with the fine-grained activity classification and detection and compare three types of approaches described in Section 3.5, namely pose-based, hand-centric and holistic approaches. Next we evaluate our approaches for composite activity recognition introduced in Section 3.6, evaluating our attributes enhanced with context and co-occurrence, the recognition of composite cooking activities using different levels of supervision, and the zero-shot approach using script data.

3.7.1 Experimental Setup

This section details our experimental setup. We will release evaluation code to reproduce and compare with our results. See Table 3.3 for the information on our training/validation/test split. We estimate all hyper parameters on the validation set and then retrain the models on the training and validation set with the best parameters.

3.7.1.1 *Experimental setup fine-grained activity classification and detection*

In the fine-grained recognition task we want to distinguish 67 fine-grained activities and 155 participating objects (see Tables 3.7, 3.8 for the lists of activities and objects). To learn the visual classifiers we use the annotated ground truth intervals provided with the dataset. We train one-vs-all SVMs using mean SGD (Rohrbach *et al.*, 2011) with a χ^2 kernel approximation (Vedaldi and Zisserman, 2010). For detection we use the midpoint hit criterion to decide on the correctness of a detection, i.e. the midpoint of the detection has to be within the ground-truth. If a second detection fires for one ground-truth label, it is counted as false positive. In the following we report the mean over the average precision (AP) of each class. Combining features is achieved by stacking the bag-of-word histograms.

3.7.1.2 *Experimental setup composite activity recognition*

For localizing attributes within composite activities we rely on our automatic segmentation (Section 3.6.5). We aim to recognize 31 composite activities (see bold names in Table 3.2).

We distinguish two cases for training the attributes with respect to composites.

Attribute training on all composites. We use all available 218 training+validation videos for training the attribute classifiers. See left half of Tables 3.10, 3.11, and 3.12.

Attribute training on disjoint composites. We use all available videos apart from those showing the test composite categories (in total 92 videos). This means that attributes and composites are trained on disjoint sets of composite categories and thus also on disjoint sets of videos. This tests how well novel composite categories can be recognized without additional attribute labels. See right half of Tables 3.10, 3.11, and 3.12.

Next, we have two cases for training the composites.

With training data for composites. We train on the 126 training+validation videos whose category is in the set of the 31 test categories. Note that in case of *Attribute training on all composites* the training videos are also part of the attribute training. See top part of Table 3.11.

No training data for composites. Here we do not rely on any training labels for the composite activities. See bottom part of Table 3.11 and all of Table 3.12. Combined with *Attribute training on disjoint composites* this is zero-shot recognition.

3.7.2 Fine-grained activity classification and detection

Activity classification. We start with the classification results on fine-grained activities and their participants (Table 3.6).

Approach	Activities	Objects	All
Pose-based approaches			
(1) BM	18.9	13.8	15.7
(2) FFT	19.0	16.2	17.2
(3) Combined	24.1	19.0	20.8
Hand-centric approaches			
(4) Hand-cSift	23.0	23.8	23.5
(5) Hand-Trajectories	45.1	31.5	36.4
(6) Combined	43.5	34.2	37.5
Holistic approach			
(7) Dense Trajectories	44.5	31.3	36.1
Combinations			
(8) Dense Traj,BM,FFT	43.1	30.7	35.2
(9) Dense Traj,Hand-Traj	52.2	37.7	42.9
(10) Dense Traj,Hand-Traj,-cSift	51.2	39.3	43.7

Table 3.6: Fine-grained activity and object classification results, mean AP in % (see Section 3.7.2 for discussion).

The body model features on the joint tracks (BM) achieve a mean average precision (AP) of 18.9% for activities and 13.8% for objects. Comparing this to the FFT features, we observe that FFT performs slightly better, improving over BM the AP by 0.1% and 2.4% respectively. The combination of BM and FFT features (line 3 in Table 3.6) yields a significant improvement, reaching AP of 24.1% for activities and 19.0% for objects. We attribute this to the complementary information encoded in the features. While BM encodes among others velocity-histograms of the joint-tracks and statistics between tracks of different joints, FFT features encode FFT coefficients of individual joints. Still, this is a relatively low performance. It can be explained, on one hand, by failures of the pose estimation method and, on the other hand, the pose-based features might not contain enough information to successfully distinguish the challenging fine-grained activities and participating objects. Next we look at the performance of our proposed hand-centric features. Color Sift features, densely sampled in the hand neighborhood, allow us to improve the object recognition AP to 23.8% (Hand-cSift), indicating their better suitability in particular for recognizing objects. Dense Trajectories features computed around hands (denoted as Hand-Trajectories) reach 45.1% and 31.5% recognition AP for activities and objects, respectively. Combining both features leads to a small disimprovement for activities, however it helps to further improve the object recognition performance to 34.2%. Overall our hand-centric approach reaches the recognition AP of 37.5% for activities and objects together. The state-of-the-art holistic approach of Dense Trajectories (Wang *et al.*, 2013a) obtains 44.5% and 31.3% recognition AP for activities and objects. If compared to our hand-centric features, this is slightly below the Hand-Trajectories,

Activity	Dense Traj	Hand Traj	Combi +cSift
add	19.8	16.3	24.0
arrange	61.9	32.1	33.8
change temperature	69.1	78.1	75.4
chop	36.6	35.4	48.3
clean	32.0	33.0	33.3
close	76.3	68.8	77.0
cut apart	33.8	36.2	33.5
cut dice	39.3	45.7	44.9
cut off ends	21.4	52.0	31.9
cut out inside	2.2	0.8	2.0
cut stripes	12.9	13.0	15.4
cut	28.3	44.9	27.2
dry	81.9	85.1	84.5
enter	100.0	100.0	100.0
fill	94.3	90.8	86.2
gather	25.7	23.8	35.7
grate	66.7	100.0	100.0
hang	85.8	57.2	81.4
mix	10.3	5.4	52.9
move	75.7	75.7	78.3
open close	60.8	65.7	64.7
open egg	50.0	28.1	39.2
open tin	-	-	-
open	22.0	22.0	34.5
package	0.4	1.6	1.8
peel	55.0	67.2	58.6
plug	41.6	32.6	81.0
pour	44.8	44.9	45.1
pull apart	38.7	53.8	45.2
pull up	79.2	21.7	75.6
pull	1.3	9.1	1.2
puree	-	-	-
purge	0.1	0.1	0.6
push down	30.7	7.6	28.0
put in	55.5	50.8	58.0
put lid	87.3	85.3	90.0
put on	6.2	5.6	1.2
read	5.1	5.4	5.6
remove from package	19.3	34.3	31.5
rip open	2.8	45.0	100.0
scratch off	30.7	33.1	31.9
screw close	77.3	77.5	77.5
screw open	78.7	69.4	79.2
shake	73.0	75.7	77.3
shape	-	-	-
slice	47.2	71.3	57.4
smell	49.7	15.7	33.0
spice	88.6	89.0	89.2
spread	87.1	77.1	96.7
squeeze	90.1	92.9	91.9
stamp	-	-	-
stir	91.2	81.9	91.7
strew	1.7	2.4	2.4
take apart	1.6	32.1	53.3
take lid	66.2	76.8	71.7
take out	94.1	93.9	95.1
tap	3.3	4.2	6.2
taste	9.4	21.0	22.0
test temperature	11.3	11.8	35.1
throw in garbage	96.7	96.0	97.1
turn off	7.4	21.1	33.0
turn on	27.8	30.6	48.5
turn over	-	-	-
unplug	8.7	3.8	20.0
wash	93.4	93.9	93.7
whip	-	-	-
wring out	3.3	4.5	5.3

Table 3.7: Fine-grained activities classification performance of Dense Trajectories, Hand Trajectories, and their combination including Hand-cSift (line 10 in Table 3.6) for 67 fine-grained activities. AP in %. "-" denotes that the category is not part of the test set and not evaluated.

which are restricted to the areas around hands. This supports our hypothesis that the most relevant information for recognizing our fine-grained activities is contained in the hand regions. We also consider several feature combinations (lines 8, 9, 10 in Table 3.6). Combining Dense Trajectories with the pose-based features does not improve the recognition performance. However, combining them with Hand-Trajectories improves the activity recognition by 7.7% and object recognition by 6.4% (line 7 vs 9 in Table 3.6). Finally, adding the Hand-cSift features allows to reach the impressive 43.7% recognition AP for activities and objects together.

The detailed comparison of Dense Trajectories, Hand-Trajectories and the final feature-combination (line 10 in Table 3.6) can be found in Tables 3.7 and 3.8. Hand-Trajectories loose to Dense Trajectories on activities that include "coarser" motion, e.g. *push down*, *hang* or *plug*, and corresponding objects such as *hook* or *teapot*. Note that Hand-Trajectories outperform the Dense Trajectories for 35 activity classes, while in the opposite direction this holds only 25 times (for objects, respectively 65 vs

43 times). This shows again that the hand-centric features consistently outperform the holistic features in both tasks. Some example cases where the hand-centric approach is significantly better, are such activities as *rip open*, *take apart*, and *grate* and such objects as *cauliflower*, *oven*, and *cup*. At the same time the final feature combination (line 10 in Table 3.6) consistently outperforms both aforementioned features in about 60% of cases. We demonstrate some qualitative results comparing Dense Trajectories to the final feature combination in Table 3.13. We also looked closer at the performance of other features. e.g. the combined pose features (line 3 in Table 3.6) perform well on "coarser", full-body activities, such as *throw in garbage*, *take out*, *move*, while rather poorly on more fine-grained activities. On the other hand the Hand-cSift features are good in recognizing objects with distinct shapes/colors, e.g. *pineapple*, *carrot*, *bowl*, etc.

Activity detection. Next we look at the detection performance (Table 3.9), which is inherently more challenging than the classification task. Here the BM features reach 8.3% overall AP and FFT get 9.3%. Their combination (line 3 in Table 3.9) gets 11.4% overall AP, while Hand-cSift only reaches 10.7%. Hand-Trajectories alone get 16.6% AP and combined with Hand-cSift they reach 22.5%, while the Dense Trajectories get 24.4% AP. As we can see for this task our hand-centric features perform worse than holistic and even pose-based features (line 3 vs 4 in Table 3.9). We believe the reason for this is that for correct segmentation of the video into activity intervals we need more holistic information, which the hand-centric features cannot provide, while pose-based and holistic features can capture it better. Similarly, when combining Dense Trajectories with the pose-based features (line 8 in Table 3.9) we observe a small improvement, supporting our hypothesis that pose indeed helps to capture the detection boundaries. On the other hand, combining Dense Trajectories with our hand-centric features significantly improves the performance, in particular by 4.7% for activities and by 3.7% for objects (line 6 vs 9 in Table 3.9). Combining the obtained features with the Hand-cSift further improves the results and we reach the 28.6% overall AP. The improvement obtained after combining holistic and hand-centric features can be explained by the increased classification AP within the obtained intervals. We thus conclude that for activity detection we require holistic information, which can come e.g. from the human pose. Combining the holistic and hand-centric features is still beneficial and significantly improves the performance.

3.7.3 Context and co-occurrence for fine-grained activities

While so far we looked at individual fine-grained activities, we now evaluate the benefit from co-occurrence and context as introduced in Section 3.6.1. Table 3.10 provides the results for recognizing activities and their participants, modeled as attributes. We evaluate in two settings. The left two columns of Table 3.10 show the results for training on all composites in training set, while the right two columns are trained only on composites absent in test set (Disjoint Composites), i.e. the second is a more challenging problem, as there is less training data and the attributes are tested

Object	Dense Traj	Hand Traj	Combi+cSift
arils	19.8	57.8	12.5
avocado	2.5	4.3	3.8
bottle	57.1	49.3	57.7
bowl	34.7	33.1	49.0
bread	3.7	6.5	8.9
bread-knife	3.0	4.0	8.1
broccoli	2.0	2.3	5.7
bun	1.2	2.3	8.5
bundle	0.5	1.1	1.4
butter	6.2	1.9	9.6
carafe	44.4	46.7	54.4
carrot	26.5	41.3	64.9
cauliflower	29.3	68.9	73.8
chefs-knife	59.9	73.3	63.1
chili	0.6	0.9	1.3
coffee	3.3	25.0	100.0
coffee-container	34.6	24.8	73.4
coffee-machine	34.7	65.1	91.2
coffee-powder	0.5	1.3	3.0
colander	63.4	62.2	77.9
counter	71.8	70.3	76.5
cream	0.9	0.5	1.4
cucumber	4.3	5.2	4.1
cup	27.0	26.7	43.6
cupboard	97.5	98.0	98.4
cutting-board	84.4	85.4	88.9
drawer	98.2	98.4	98.5
egg	12.1	3.6	7.3
eggshell	3.5	3.6	11.2
electricity-column	89.3	82.3	98.1
electricity-plug	74.3	70.6	87.7
fig	1.0	1.0	0.9
filter-basket	1.3	3.4	13.1
finger	18.4	15.4	8.8
flat-grater	31.7	27.7	40.9
fork	8.7	7.5	10.5
fridge	100.0	99.8	100.0
front-peeler	21.8	6.0	17.6
frying-pan	88.7	91.9	93.6
garbage	13.7	17.9	27.5
garlic-bulb	0.3	0.6	0.8
garlic-clove	11.7	3.6	9.3
ginger	1.9	3.3	3.6
glass	2.6	4.5	21.6
green-beans	21.1	24.6	23.2
hand	95.9	95.2	96.4
handle	100.0	9.1	100.0
hook	95.6	71.2	98.3
hot-dog	2.1	2.7	8.8
jar	5.4	14.2	17.8
ketchup	2.0	3.1	19.6
kettle-power-base	14.4	9.8	41.4
kiwi	1.1	2.9	1.5
knife	69.6	83.5	76.8
leek	10.6	19.5	17.6
lid	67.1	70.8	71.8
lime	14.2	3.7	14.6
mango	3.8	7.0	2.5
measuring-pitcher	0.7	5.0	5.3
measuring-spoon	34.1	12.6	7.3
milk	0.4	0.4	0.4
net-bag	0.3	0.2	0.7
oil	52.3	47.6	55.6
onion	19.3	20.4	22.7
orange	18.4	11.1	19.3
oven	30.7	73.4	89.3
paper-bag	20.5	10.3	33.0
paper-box	1.0	1.2	3.6
parsley	23.4	25.5	49.6
pasta	26.1	16.0	40.7
peel	40.3	28.6	35.2
pepper	3.1	14.4	6.7
pineapple	19.5	47.0	49.7
plastic-bag	36.4	37.7	43.6
plastic-bottle	4.7	2.8	9.1
plastic-box	2.6	9.0	5.3
plastic-paper-bag	0.9	14.7	19.6
plate	65.7	69.2	73.9
plum	0.7	2.5	1.3
pomegranate	5.1	0.8	2.3
pot	84.3	88.0	91.1
potato	0.4	0.4	0.6
salt	59.8	48.7	64.1
side-peeler	50.0	11.7	37.8
sink	47.0	54.0	53.9
spatula	72.9	76.2	78.2
spice	19.1	13.3	12.4
spice-holder	95.6	94.4	96.3
spice-shaker	88.3	87.3	91.5
sponge	17.2	45.4	38.2
sponge-cloth	67.1	68.1	75.0
spoon	2.8	5.9	8.9
squeezer	52.5	67.0	59.3
stone	0.2	0.7	0.7
stove	84.4	87.2	90.4
sugar	22.0	24.2	29.0
tap	70.2	71.8	79.1
tea-egg	37.2	28.7	36.1
tea-herbs	60.5	55.6	91.1
teapot	46.4	6.7	69.1
teaspoon	29.2	32.4	36.5
toaster	1.3	8.1	6.7
towel	73.2	76.9	79.2
tube	1.0	9.5	10.2
water	55.0	46.9	57.2
water-kettle	40.7	25.9	53.7
wrapping-paper	2.9	0.4	2.0
yolk	0.5	0.5	0.3

Table 3.8: Object classification performance of Dense Trajectories, Hand Trajectories, and their combination including Hand-cSift (line 10 in Table 3.6) for 108 participating objects. *(47 objects are not in the test set and thus not evaluated: apple, asparagus, bag, baking-paper, baking-tray, blender, box-grater, cheese, chive, chocolate, cooking-spoon, corn, dough, flower-pot, food, ham, hot-chocolate-powder-bag, knife-sharpener, kohlrabi, ladle, lemon, masher, mortar, mushroom, oregano, paper, peach, pear, peppercorn, pestle, philadelphia, puree, raspberries, salad, salami, seed, soup, spinach, table-knife, tin, tin-opener, tissue, tomato, tongs, top, wire-whisk, zucchini.)*

Approach	Activities	Objects	All
Pose-based approaches			
(1) BM	9.7	7.6	8.3
(2) FFT	10.5	8.7	9.3
(3) Combined	14.3	9.8	11.4
Hand-centric approaches			
(4) Hand-cSift	10.5	10.9	10.7
(5) Hand-Trajectories	21.3	14.0	16.6
(6) Combined	26.0	20.6	22.5
Holistic approach			
(7) Dense Trajectories	29.5	21.5	24.4
Combinations			
(8) Dense Traj,BM,FFT	30.7	21.5	24.8
(9) Dense Traj,Hand-Traj	34.3	25.2	28.5
(10) Dense Traj,Hand-Traj,-cSift	34.5	25.3	28.6

Table 3.9: Fine-grained activity and object detection results, mean AP in % (see Section 3.7.2 for discussion)

in a different context. The performance in the first line is equivalent to the results in Table 3.6. The very left column shows results on Dense Trajectories. More specifically using only temporal context to recognize activity attributes performance drops from 36.1% AP for the base classifier to 11.1% AP. This is the expected result, because the context is similar for all activities of the same sequence and thus cannot discriminate attributes. In contrast, when using co-occurrence only (line 4 in Table 3.10), the performance increases by 2.0% compared to the base classifiers due to the high relatedness between the attributes, namely between activities and their participants. Combining context and co-occurrence information with the base classifier gives 37.8% and 38.1%, respectively. A combination of all training modes achieves a performance of 39.3% AP, improving the base classifier's result by 3.2%. While results for Dense Trajectories are as expected i.e. adding context and co-occurrence improves performance, the performance drops slightly for the (in general) better performing combined features (second column). However, although the attribute prediction performance drops, we found that for recognizing the composites, context and co-occurrence are still useful.

In the second setting, we restrict the training dataset to composites absent in the test set (right two columns of Table 3.10), requiring the activity attributes to transfer to different composite activities. When comparing the right two the left columns, we notice a significant performance drop for all classifiers and both features. This decrease can mainly be attributed to the strong reduction of training data to about one third. The base classifier performs best and co-occurrence variants slightly below. Variants including context lead to tremendous performance drops in all

Attribute training on:	All Composites		Disjoint Composites	
	Dense Traj	Combi +cSift	Dense Traj	Combi +cSift
(1) Base (s^{base})	36.1	43.7	33.5	35.9
(2) Context only (s^{con})	11.1	12.6	6.8	8.1
(3) Base+Context	37.8	41.2	28.3	32.3
(4) Co-occ. only (s^{coocc})	38.1	41.7	32.6	35.3
(5) Base+Co-occ.	38.1	41.4	32.7	35.2
(6) Base+Cont.+Co-occ.	39.3	41.5	30.8	32.6

Table 3.10: Attribute recognition using context and co-occurrence, mean AP in %. Combi+cSift refers to Dense Traj,Hand-Traj,-cSift, see Section 3.7.3 for discussion.

combinations because the activity context changes from training to test (having different composite activities).

3.7.4 Composite cooking activity classification

After evaluating attribute recognition performance in Section 3.7.3, we now show the results for recognizing composites as introduced in Section 3.6.2. From the different attribute combination variants we only use the combination of base, context, and co-occurrence (last line in Table 3.10). Although this is not always the best choice for recognizing attributes we found it to work better or similar to alternatives for composite recognition. The results are shown in Table 3.11, which, similar to Table 3.10, shows results for training the attributes on all composites, on the left, and reduced attribute training on non-test composites on the right. In the top section of the table we use training data for the composite cooking activities. In the bottom section of the table we use *n*o training data for the composite cooking activities. This is enabled by the use of script data as motivated before. Disregarding the first line which does not use attributes at all and the second line which uses ground truth intervals for attributes, all other lines are based on attributes computed on our automatic temporal segmentation, introduced in Section 3.6.5.

Examining the results in Table 3.11 we make several interesting observations. First, training composites on attributes of fine-grained activities and objects (line 3 in Table 3.11) outperforms low-level features (line 1 in Table 3.11), supporting our claim that for learning composite activities it is important to share information on an intermediate level of attributes.

The second somewhat surprising observation is that recognizing composites based on our segmentation (line 3 in Table 3.11) outperforms using ground truth segments (line 2 in Table 3.11). We attribute this to the fact that our segmentation is coarser than the ground truth and that we additionally remove noisy and background

Attribute training on:	All Composites		Disjoint Composites	
	Dense Traj	Combi +cSift	Dense Traj	Combi +cSift
With training data for composites				
Without attributes				
(1) SVM	39.8	41.1	-	-
Attributes on gt intervals				
(2) SVM	43.6	52.3	32.3	34.9
Attributes on automatic segmentation				
(3) SVM	49.0	56.9	35.7	34.8
(4) NN	42.1	43.3	24.7	32.7
(5) NN+Script data	35.0	40.4	18.0	21.9
(6) PST+Script data	54.5	57.4	32.2	32.5
No training data for composites				
Attributes on automatic segmentation				
(7) Script data	36.7	29.9	19.6	21.9
(8) PST + Script data	36.6	43.8	21.1	19.3

Table 3.11: Composite cooking activity classification, mean AP in %. Top left quarter: fully supervised, right column: reduced attribute training data, bottom section: no composite cooking activity training data, right bottom quarter: true zero shot. See Section 3.7.4 for discussion.

segments with a background classifier. This leads to more robust attributes and consequently better composite recognition. This allows to have separate training sets for composites and attributes. This setting is explored in the top right quarter of Table 3.11. Here the training sequences for attributes are disjoint with the ones for composites, i.e. we do not require the attribute annotataions for the composite training set.

Third, the improvements we achieved for fine-grained activities and object recognition by combining hand-centric with holistic features are still evident for composites. The Combination of Dense Trajectoreis, Hand-Trajectories, and Hand-cSift (2^{nd}, 4^{th} column) outperforms in most cases Dense Trajectories only (1^{st}, 3^{rd} column), most notably in the setting "All Composites" for SVM (56.9% over 49.0% AP) and PST+Script data (43.8% over 36.6% AP).

Fourth, using our Propagated Semantic Transfer (PST) approach is in most cases superior to other variants of incorporating script data (NN+Script data/ Script data). Most notably it reaches 57.5% AP for our combined feature. This is the overall best performance and also outperforms the SVM with 56.6% AP. PST slightly drops for the last number in table (19.3%), which we found is due to rather suboptimal parameters selected on the validations set. We note that in the scenario of Disjoint Composites (top right quarter of Table 3.11) PST+Script data is outperformed by

Attribute training on:	All Composites		Disjoint Composites	
	Dense Traj	Combi +cSift	Dense Traj	Combi +cSift
No training data for composites				
Script data				
(1) freq-literal	28.2	30.5	19.8	24.1
(2) freq-WN	25.3	28.6	17.4	20.3
(3) tf∗idf-literal	35.9	31.8	20.0	23.6
(4) tf∗idf-WN	36.7	29.9	19.6	21.9

Table 3.12: Variants of script knowledge, AP in %. Combi+cSift refers to Dense Traj,Hand-Traj,-cSift. See Section 3.7.4 for discussion.

training an SVM. We attribute this to the fact that the attributes are less robust in this scenario (see Table 3.10) and the SVM can better adjust to that by learning which attributes are reliable and which not. NN and PST are based on distances between attribute score vectors, thus metric learning could be beneficial in these cases.

Fifth, script data does not only allow to achieve the maximum performance but also allows transfer (bottom part of Table 3.11) achieving in some cases results close to supervised approaches. The bottom right part of the table shows zero-shot recognition. Although here the performance cannot compete with the supervised setting, we like to point out that this is a very challenging scenario, where attributes are trained on different composites, without composite training data, and the video stream has to be segmented automatically.

Sixth, while in Table 3.11 we always used the variant tf∗idf-WN for Script data, we show different variants of Script data for the case where they are not combined with NN or PST in Table 3.12. The main observation is that freq-WN performs in all cases worst, most likely the WordNet expansions make the results noisier. While in the first column the tf∗idf-WN works best, there is overall no clear winner. However, when incorporated in PST, it is more important to select appropriate parameters for PST on the validation set rather than selecting the right variant of Script data.

Last, we want to look at an interesting comparison of the first line (SVM without attributes) versus line 8 (PST + Script data), which effectively compares the settings "only composite labels" versus "only attribute labels" (+ Script data). Although the latter does not have any labels for the actual task of composite recognition it either performs close (in case of Dense Trajectories) or slightly better (for combined features). This indicates that our PST + Script data approach is very good in transferring information from the original task it was trained on to another which is very important for adaptation to novel situations, typical for assisted daily living scenarios.

Table 3.13 provides qualitative results for three composite videos including how they are decomposed into fine-grained activities and participating objects.

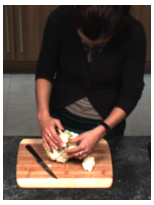
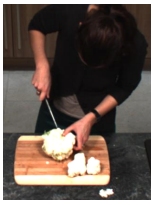
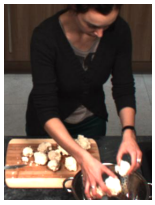
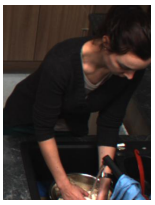
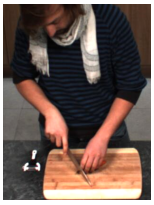
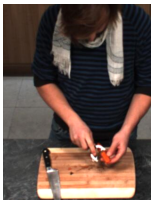
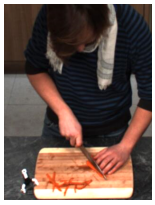
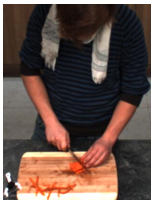
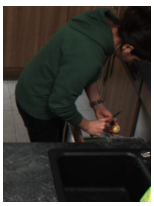
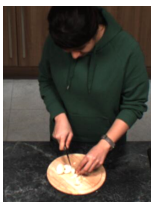
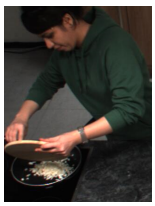
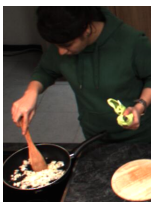
					Composite
Ground-truth	**cauliflower**, **cutting-board**, **hand**, **pull apart(A)**	**cauliflower**, **cut(A)**, **cutting-board**, **knife**	**add(A)**, **cauliflower**, **colander**, **cutting-board**, **hand**	**cauliflower**, **colander**, **hand**, **wash(A)**	**Preparing cauliflower**
Dense Traj	**hand**, **cutting-board**, **pull apart(A)**, onion, peel, cut apart(A)	**knife**, **cutting-board**, cut apart(A), counter, chefs-knife, **cut(A)**	**hand**, **cutting-board**, move(A), counter, bowl, **colander**	**hand**, **wash(A)**, plate, **colander**, onion, peel	Preparing orange
Dense Traj, Hand-Traj, -cSift	**hand**, **cutting-board**, cut apart(A), **cauliflower**, onion, **pull apart(A)**	**cauliflower**, cut apart(A), **knife**, chefs-knife, **cutting-board**, **cut(A)**	**hand**, **cutting-board**, move(A), counter, **cauliflower**, **colander**	**hand**, **wash(A)**, bowl, **colander**, **cauliflower**, onion	**Preparing cauliflower**
					Composite
Ground-truth	**carrot**, **chefs-knife**, **cut off ends(A)**, **cutting-board**	**carrot**, **front-peeler**, **peel(A)**	**carrot**, **chefs-knife**, **cut stripes(A)**, **cutting-board**	**carrot**, **chefs-knife**, **cut apart(A)**, **cutting-board**	**Preparing carrot**
Dense Traj	**cutting-board**, cut apart(A), **chefs-knife**, **cut off ends(A)**, knife, put on(A)	cutting-board, **peel(A)**, **front-peeler**, chefs-knife, knife, cucumber	**cutting-board**, **chefs-knife**, slice(A), knife, cut apart(A), cucumber	**cutting-board**, **cut apart(A)**, **chefs-knife**, knife, cauliflower, cut off ends(A)	Preparing cucumber
Dense Traj, Hand-Traj, -cSift	**cutting-board**, **cut off ends(A)**, **chefs-knife**, cut apart(A), knife, **carrot**	cutting-board, **peel(A)**, **carrot**, chefs-knife, **front-peeler**, cucumber	**cutting-board**, **chefs-knife**, slice(A), knife, **carrot**, cut apart(A)	**cutting-board**, **cut apart(A)**, **chefs-knife**, cut off ends(A), knife, **carrot**	**Preparing carrot**
					Composite
Ground-truth	**knife**, **onion**, **peel(A)**	**chop(A)**, **cutting-board**, **knife**, **onion**	**add(A)**, **cutting-board**, **frying-pan**, **knife**, **onion**	**frying-pan**, **onion**, **spatula**, **stir(A)**	**Preparing onion**
Dense Traj	**peel(A)**, hand, **onion**, throw in garbage(A), bowl, front-peeler	**cutting-board**, **knife**, cut dice(A), onion, **chop(A)**, slice(A)	hand, **frying-pan**, **cutting-board**, pot, spatula, **add(A)**	**spatula**, **frying-pan**, **stir(A)**, **onion**, add(A), egg	**Preparing onion**
Dense Traj, Hand-Traj, -cSift	**peel(A)**, hand, throw in garbage(A), **onion**, **knife**, peel	**cutting-board**, **knife**, cut dice(A), slice(A), **chop(A)**, chive	hand, **frying-pan**, **add(A)**, pot, spatula, cauliflower	**frying-pan**, **spatula**, **stir(A)**, **onion**, add(A), broccoli	**Preparing onion**

Table 3.13: Qualitative results for Dense Trajectories and its combination with hand-centric features (line 10 in Table 3.6). We show top-6 highest scoring attributes, activities(A) and objects, and composite activity predictions. Correct results are marked with bold. Many predictions are not correct according to the ground truth but very relevant, e.g. *slice* instead of *cut stripes*.

3.8 CONCLUSION

In this chapter we address two challenges that have not been widely explored so far, namely fine-grained activity recognition and composite activity recognition. In order to approach these tasks we propose the large activity database MPII Cooking 2. We recorded and annotated 273 videos of more than 27 hours with 30 human subjects performing a large number of realistic cooking activities. Our database is unique with respect to size, length, complexity of the videos, and available annotations (activities, objects, human pose, text descriptions).

To estimate the complexity of fine-grained activity recognition in our database we compare three types of approaches: pose-based, hand-centric, and holistic. We evaluate on a classification and the often neglected detection task. Our results show that for recognizing fine-grained activities and their participating objects it is beneficial to focus on hand regions as the activities are hand-centric and the relevant objects are in the hand neighbourhood.

Composite activities are difficult to recognize because of their inherent variability and the lack of training data for specific composites. We show that attribute-based activity recognition allows recognizing composite activities well. Most notably, we describe how textual script data, which is easy to collect, enables an improvement of the composite activity recognition when only little training data is available, and even allows for complete zero-shot transfer.

As part of future work we plan to validate our hand-centric approach in other domains and exploit the scripts for composite activity recognition by modeling the temporal structure of the video.

In the following chapter we explore how to generate natural language descriptions for our cooking videos.

4 COHERENT MULTI-SENTENCE VIDEO DESCRIPTION WITH VARIABLE LEVEL OF DETAIL

HUMANS can easily describe what they see in a coherent way and at varying level of detail. However, existing approaches for automatic video description focus on generating only single sentences and are not able to vary the descriptions' level of detail. In this chapter, we address both of these limitations: for a variable level of detail we produce coherent multi-sentence descriptions of complex videos. To understand the difference between detailed and short descriptions, we collect and analyze a video description corpus of three levels of detail. We rely on the videos from our MPII Cooking 2 dataset (Chapter 3). We follow a two-step approach where we first learn to predict a semantic representation (SR) from video and then generate natural language descriptions from it. For our multi-sentence descriptions we model across-sentence consistency at the level of the SR by enforcing a consistent topic. We contribute to the robust generation of sentences using a word lattice. For the visual recognition of participating objects we rely on the hand-centric approach, introduced in the previous chapter. Human judges rate our descriptions as more readable, correct, and relevant than related work.

4.1 INTRODUCTION

Describing videos or images with natural language sentences is an intriguing but difficult task. Recently it has received increased interest both in the computer vision (Das *et al.*, 2013; Farhadi *et al.*, 2010b; Guadarrama *et al.*, 2013; Kulkarni *et al.*, 2011; Rohrbach *et al.*, 2013b) and computational linguistic communities (Krishnamoorthy *et al.*, 2013; Kuznetsova *et al.*, 2012; Yu and Siskind, 2013). The focus of most works on describing videos is to generate single sentences for short video snippets at a fixed level of detail. In contrast, we want to generate coherent multi-sentence descriptions for long videos with multiple activities and allow for producing descriptions at the required levels of detail (see Fig. 4.1).

Multi-sentence description, our first task, has been explored for videos (Das *et al.*, 2013; Khan *et al.*, 2011; Tan *et al.*, 2011), but open challenges remain, e.g. finding a segmentation of appropriate granularity and generating a conceptually and linguistically coherent description. To allow reasoning across sentences we use an intermediate semantic representation (SR) which is inferred from the video. For generating multi-sentence descriptions we ensure that sentences describing the same video are about the same topic (dish in our cooking scenario) and we improve intra-sentence consistency by allowing our language model to choose from a probabilistic SR rather than a single MAP estimate.

Detailed Description: A man took a cutting board and knife from the drawer. He took out an orange from the refrigerator. Then, he took a knife from the drawer. He juiced one half of the orange. Next, he opened the refrigerator. He cut the orange with the knife. The man threw away the skin. He got a glass from the cabinet. Then, he poured the juice into the glass. Finally, he placed the orange in the sink.

Short Description: A man juiced the orange. Next, he cut the orange in half. Finally, he poured the juice into a glass.

Single Sentence Description: A man juiced the orange.

Figure 4.1: Output of our system for a video, producing coherent multi-sentence descriptions at three levels of detail, using our automatic segmentation.

The second task is generating descriptions with a varying level of detail. While this is a researched problem in natural language generation, e.g. in context of user models (Zukerman and Litman, 2001), we are not aware of any work in computer vision that studies how to select the desired amount of information to be recognized. To understand which information is required for producing a description at a needed level of detail we collected descriptions at three levels of detail for the same video and analyzed which aspects of the video are verbalized in each case.

The first contribution of this chapter is to generate coherent multi-sentence descriptions. For this task we (a) propose a model which enforces conceptual consistency across sentences (Sec. 4.3.1), (b) suggest a simple but effective (and to our knowledge novel) segmentation approach, (c) significantly improve the visual recognition based on the semantic unaries and hand-centric features to provide a consistent description (Sec. 4.4), (d) couple visual recognition and language generation using a word lattice to improve consistency within each sentence, and (e) improve linguistic cohesiveness/readability (Sec. 4.5). Our second contribution is to propose a novel task of describing videos at multiple levels of detail. To approach this task we (a) collected and aligned a corpus of descriptions of three levels of detail, which we provide on our web-page (Sec. 4.2), (b) perform a thorough analysis of the collected data (Sec. 4.2), and (c) propose an approach to handle this new task: namely by selecting the relevant video segments according to the topic and using a language model learned for the right level of detail (Sec. 4.3.2).

4.2 ANALYSIS OF HUMAN VIDEO DESCRIPTIONS AT MULTIPLE LEVELS OF DETAIL

An important goal of our work is to generate natural language descriptions for videos at different levels of detail. In this section we investigate which aspects of a

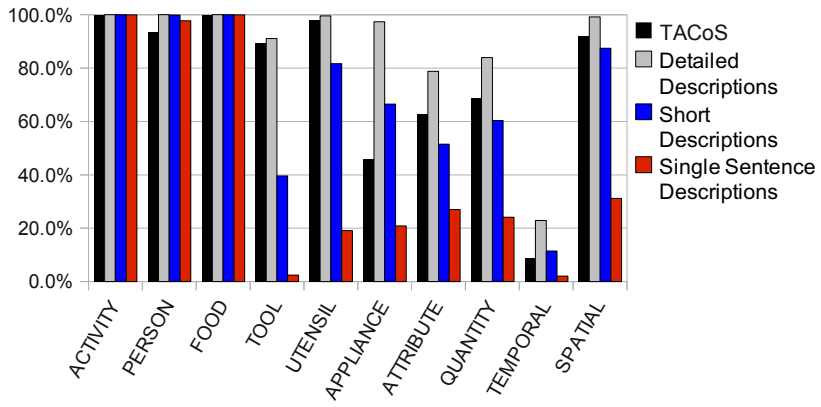


Figure 4.2: Percentage of descriptions in which each category is verbalized.

video are verbalized by humans and how descriptions of different levels of detail differ.

Data collection. We have selected a subset (185 videos) from the *MPII Cooking 2* dataset (Chapter 3) and collected text descriptions for the videos via Amazon Mechanical Turk (AMT). For each video we asked to describe it in three ways: (1) a Detailed Description with at most 15 sentences, (2) a Short Description (3-5 sentences), and (3) a Single Sentence Description. In total, we have collected a corpus with about 20 triples of descriptions for each video. which we call *TACoS Multi-Level*. Unlike Regneri *et al.* (2013), workers could freely describe videos without aligning each sentence to the video. Our data collection hence results in more natural descriptions, having a more complex sentence structure (e.g., they make use of temporal connectives and anaphora).

Analysis of human-written descriptions. First, we analyze the collected descriptions with respect to which aspects of the videos are verbalized. We assign part-of-speech (POS) tags to the collected descriptions and the ones provided by TACoS using the Stanford POS tagger (Toutanova *et al.*, 2003). Any word tagged as a verb is considered to be an *activity*, and any word tagged as an adjective is considered to represent an *attribute*. We classify all adverbials as providing *spatial* or *temporal* information using a hand-compiled list of adverbials. *quantity* information is assumed when one of the words has been tagged as a cardinal number or when a noun is a hyponym, i.e., in an *is-a* relation, of 'quantity' or 'portion' in WordNet Fellbaum (1998). We use *person*, *food*, *tool*, *utensil* or *appliance* and categories for nouns. To identify the category of a specific noun, we check whether the words are hyponyms of appropriate WordNet entries, and additionally check manually created white- and blacklists for each category. *food* is considered to be any edible item or dish. *tools* are items such as *knife* or *chopper*, while *utensils* are other kitchen utensils such as *bowl* or *cutting board*. Finally, the *appliance* category comprises non-movable items such as *stove*, *kitchen* or *sink*. Figure 4.2 shows the percentages of descriptions in which at

least one word of the respective category occurred. *activities*, *food* items and the *person* are mentioned in almost all descriptions. For *tools*, *utensils*, *appliances*, *attributes*, *quantities*, and *spatial* the occurrence frequency decreases as the descriptions become shorter. *tools*, *utensils*, and *appliances* nearly fully disappear in the Single Sentence Descriptions. The Detailed Descriptions and the descriptions from TACoS are similar except in the *appliance* category.

Next, we performed a qualitative comparison of the most frequent activities/food items verbalized in different types of descriptions. The descriptions from TACoS, the Detailed Descriptions and the Short Description mainly use verbs describing specific activities, such as *cut* or *take*. In the Single Sentence Descriptions, verbs such as *prepare*, *cook* and *make*, which summarize a set of activities, are frequently used. This indicates that when generating Single Sentence Descriptions of videos, it may not be sufficient to simply extract sentences from the longer descriptions, but some degree of abstractive summarization is needed. Regarding the food items mentioned in the collected descriptions we find the following. While the Detailed Descriptions frequently mention common ingredients such as *water*, *salt* or *spice*, this is less for the Short Descriptions, and almost never for the Single Sentence Descriptions. In the Short Descriptions humans mention the objects that are more relevant for the respective dish, which are usually the main ingredients such as *potato* or *carrot*, and skip the rest. Correspondingly, in the Single Sentence Descriptions humans only focus on the main ingredients. This suggests that knowing the dish that is being prepared is necessary in order to determine the important objects to be verbalized.

Discussion. We draw four conclusions from this analysis. (1) In the Detailed Descriptions all activities and objects are mentioned, therefore the visual recognition system should identify all of them. (2) The Short Descriptions could be obtained from Detailed Descriptions using extractive summarization techniques. However, the various levels show different relative frequency of verbalized concepts, hence it might be beneficial to learn a language model targeted to a desired level. (3) The Single Sentence Descriptions qualitatively differ from all other types, which suggests that abstractive summarization is required for this level. (4) It is important to recognize the topic (dish that is prepared, in our scenario). This would also help to generate consistent multi-sentence descriptions, another goal of this chapter.

4.3 GENERATING CONSISTENT MULTI-SENTENCE VIDEO DESCRIPTIONS AT MULTIPLE LEVELS OF DETAIL

Based on an analysis how humans describe videos we present our approach to generate consistent multi-sentence descriptions at multiple levels on detail.

4.3.1 Multi-sentence video descriptions

Assume that a video v can be decomposed into a set of I video snippets represented by video descriptors $\{x_1, ..., x_i, ..., x_I\}$, where each snippet can be described by a single sentence z_i. To reason across sentences we employ an intermediate semantic representation (SR) y_i. We base our approach for a video snippet on the translation approach proposed by Rohrbach *et al.* (2013b). We choose this approach as it allows to learn both the prediction of a semantic representation $x \rightarrow y$ from visual training data (x_i, y_i) and the language generation $y \rightarrow z$ from an aligned sentence corpus (y_i, z_i). While this chapter builds on the semantic representation of Rohrbach *et al.* (2013b), our idea of consistency is applicable to other semantic representations. The SR y is a tuple of activity and participating objects/locations, e.g. in our case $\langle$*activity, tool, object, source, target*$\rangle$. The relationship is modeled in a CRF where these entities are modeled as nodes $n \in \{1, ..., N\}$ ($N = 5$ in our case) observing the video snippets x_i as unaries. We define s_n as a state of node n, where $s_n \in S$. We use a fully connected graph and linear pairwise (p) and unary (u) terms. In addition, to enable a consistent prediction within a video, we introduce a high level topic node t in the graph, which is also connected to all nodes. In contrast to the other nodes it observes the entire video v rather than a single video snippet. For the topic node t we define a state $s_t \in T$. We then use the following energy formulation for the structured model:

$$E(s_1, ..., s_N, s_t | x_i, v) = \sum_{n=1}^{N} E^u(s_n | x_i) + E^u(s_t | v) + \sum_{l \sim m}^{l,m \in \{1,...,N,t\}} E^p(s_l, s_m) \qquad (4.1)$$

with $E^p(s_l, s_m) = w^p_{l,m}$, where $w^p_{l,m}$ are the learned pairwise weights between the CRF node-states s_l and s_m. We discuss the unary features in Sec. 4.4.

While adding the topic node makes each video snippet aware of the full video, it does not enforce consistency across snippets. Thus, at test time, we compute the conditional probability $p(s_1, ..., s_N | \hat{s_t})$, setting s_t to the highest scoring state $\hat{s_t}$ over all segments i:

$$(\hat{s_t}, \hat{i}) = \arg \max_{s_t \in T, i \in I} p(s_t | x_i, v). \qquad (4.2)$$

We learn the model by independently training all video descriptors x_i and SR labels $y_i = \langle s_1, \ldots, s_N, s_t \rangle$ using loopy belief propagation implemented by Schmidt (2013). The possible states of the CRF nodes are based on the video segment labels and topic (dish) labels of the videos provided by our approach from Chapter 5.

Segmentation. For the described approach, we have to split the video v into video-snippets x_i. Two aspects are important for this temporal segmentation: it has to find the appropriate granularity so it can be described by a single sentence and it should not contain any unimportant (background) segments which would typically not be described by humans. For the first aspect, we employ agglomerative clustering on a score-vector of semantic attribute classifiers (see Sec. 4.4). The termination threshold is selected to capture the annotation granularity (number of intervals). The

second aspect is achieved by training a background classifier on all unlabeled video segments as negative examples versus all labeled snippets as positive. We evaluate the quality of our segmentation with respect to the final task, namely generating natural language descriptions, in Sec. 4.6.

4.3.2 Multi-level video descriptions

Based on the observations discussed in Sec. 4.2, we propose to generate shorter descriptions by extracting a subset of segments from our segmentation. We select relevant segments by scoring how discriminative their predicted SR is for the predicted topic by summing the *tf*idf* scores of the node-states, computed on the training set. For the SR $\langle s_1, \ldots, s_N, s_t \rangle$, its score r equals to:

$$r(s_1, ..., s_N, s_t) = \sum_{n=1}^{N} tf^*idf\,(s_n, s_t) \tag{4.3}$$

where *tf*idf* is defined as the normalized frequency of the state s_n (i.e. activity or object) in topic s_t times the inverse frequency of its appearance in all topics:

$$tf^*idf\,(s_n, s_t) = \frac{f(s_n, s_t)}{\max_{s'_n \in S} f(s'_n, s_t)} \log\left(\frac{|T|}{\sum_{s'_t \in T} f(s_n, s'_t) > 0}\right) \tag{4.4}$$

This way we select the K highest scoring segments and use them to produce a Short Description of the video. One way to produce a description would be to simply extract sentences that correspond to selected segments from the Detailed Description. However, given that some concepts are not verbalized in shorter descriptions, we additionally explore the approach of learning a translation model targeted to the desired level of detail. For the Single Sentence Descriptions we assume that the predicted topic is sufficient to describe the video. Therefore, we reduce the SR to $\langle dish \rangle$ and learn a translation model to the single sentences.

4.4 IMPROVING VISUAL FEATURES

One conclusion drawn in Rohrbach *et al.* (2013b) is that the noisy visual recognition is a main limitation. Especially for our problem of multi-sentence generation it is important to recognize the manipulated objects to ensure consistency across sentences. We thus aim to improve the visual recognition by using the semantic unaries and hand-centric features.

Semantic unaries. The approach of Rohrbach *et al.* (2013b) uses visual attributes to obtain the features for CRF unaries. However, this approach ignores the semantic role of the attributes. E.g. a classifier for a visual attribute *knife* is learned disregarding whether a knife is a *tool* (*cut with a knife*), or an *object* (*take out knife*). The CRF unaries use the complete score vectors as features, namely: $E^u(s_n|x_i) =< w_n^u, x_i >$, where w_n^u is a vector of weights between the node-state s_n and the visual attributes' score

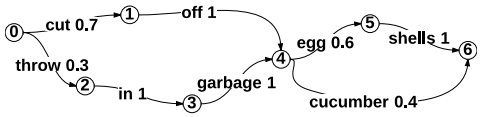


Figure 4.3: Encoding probabilistic input for SMT using a word lattice: ⟨*cut off,egg-shells*⟩ has the highest confidence but is unlikely according to the language model and other candidate paths, e.g. ⟨*cut off, cucumber*⟩ can be considered.

vector. Unlike the described method, we train SVM classifiers for visual attributes using their semantic role, e.g. we distinguish between *knife-tool* and *knife-object*. This allows us to use a score of each classifier directly as a feature for a corresponding unary: $E^u(s_n|x_i) = w^u_n x_{i,n}$. Here w^u_n is a scalar weight and $x_{i,n}$ is a score of the visual classifier. Thus we get more discriminative unaries and also reduce the number of model parameters (number of connections between node-states and visual features). The topic node unary $E^u(s_t|v)$ is defined similarly, based on the composite activity recognition features (Chapter 5) as visual descriptors of a video v.

Hand centric features for object recognition. The visual recognition approach of Rohrbach *et al.* (2013b) is based on Dense Trajectory features (Wang *et al.*, 2013a). In order to improve the object recognition, we propose to focus on hands' regions, in addition to using the holistic features that track all the moving points in the scene. This observation is intuitive, in particular in domains, where people mostly perform hand-related activities. To obtain the hand locations we use our hand detector based on appearance and body pose (Chapter 5). We densely sample the points in the hands' neighborhood, extract color Sift features (Vedaldi and Fulkerson, 2008) on 4 channels (RGB+grey) and quantize them in a codebook of size 4000. The obtained features are added as another unary to the CRF nodes.

4.5 GENERATING NATURAL DESCRIPTIONS

Probabilistic input for SMT. While the translation-based approach can achieve performance comparable with humans on ground truth SRs, this does not hold if the SRs are noisy. The approach of Rohrbach *et al.* (2013b) only takes into account the most probable prediction, the uncertainty found in the SR is not used. However, uncertain input is a known problem for SMT as speech based translation is also based on uncertain input. The work of Dyer *et al.* (2008) shows that a probabilistic input encoded in a word lattice can improve the performance of translation by decoding alternative hypotheses with lower confidence (see Fig. 4.3). A *word lattice* is a Directed Acyclic Graph allowing to efficiently decode multiple visual recognition outputs. To construct a word lattice from a set of predicted SRs ⟨*activity,tool,ingredient,source,target*⟩, we construct a word lattice for each node and then concatenate them. In case that semantic labels are empty in the SRs, we use a symbol *null+node id* to encode this information in the word lattice. SMT com-

bines scores from a phrase-based translation model, a language model, a distortion model and applies word penalties. Word lattice decoding enables us to incorporate confidence scores from the visual recognition.

Creating cohesive descriptions. As SMT generates sentences independently for each video segment, the produced descriptions seem more like a "list of sentences" rather than a "text" to readers. *Cohesion* describes the linguistic means which relate sentences on a surface level, and which do not require deep understanding of the text. Hence, we automatically post-process the descriptions such that they are more cohesive using a set of domain-independent rules: (1) we fix punctuation and create syntactic parses using the Stanford parser (Klein and Manning, 2003). (2) We combine adjacent sentences if they have the same verb but different objects. (3) We combine adjacent sentences if they have the same object but different verbs. (4) The use of referring expressions such as pronouns is a strong cohesive device. As in English, there is no appropriate pronoun for the phrase *the person*, we use gold-standard gender information and replace this phrase by appropriate nouns and pronouns. (5) We insert temporal adverbials such as *next*, *then* and *finally*.

4.6 EVALUATION

For collecting our corpus we rely on the MPII Cooking 2 dataset. This dataset is realistic and typical for assisted daily living or industrial applications which require distinguishing a large number of fine-grained activities and hand-object interaction. Besides, the dataset contains long (average 6 minutes) videos, allowing to describe them with multiple sentences and at multiple levels.

We evaluate our approach on the TACoS dataset of Regneri *et al.* (2013) and on our new corpus TACoS Multi-Level (Sec. 4.2). For TACoS we follow the setup of Rohrbach *et al.* (2013b). For the new corpus we use the training/validation/test split defined for MPII Cooking 2. Comparing to TACoS, our test split is more challenging with more videos (42 vs. 13) and more human subjects (5 vs. 1). We preprocess both corpora by substituting gender specific identifiers with "the person" and transform all sentences to past tense to ensure consistent multi-sentence descriptions.

We evaluate the generated text using BLEU@4, which computes the geometric mean of n-gram word overlaps for n=1,...,4, weighted by a brevity penalty. We also perform human evaluation of the produced descriptions asking human subjects to rate readability (without seeing the video), correctness, and relevance (with respect to the video). Readability is evaluated according to the TAC[1] definition which rates the description's grammaticality, non-redundancy, referential clarity, focus, structure and coherence. Correctness is rated per sentence with respect to the video (independent of completeness), we average the score over all sentences per description. Relevance is rated for the full descriptions and judges if the generated description captures the most important events present in the video. We select all hyperparameters (SVM,

[1] www.nist.gov/tac/2011/Summarization/Guided-Summ.2011.guidelines.html

Approach	activity	tool	object	source	target	all	dish
CRF of Rohrbach *et al.* (2013b)	59.1	79.6	36.8	71.5	78.2	21.4	-
Our CRF + Semantic unaries	59.2	81.1	39.1	73.8	77.6	23.4	-
+ Hand centric unaries	60.3	82.3	42.6	74.3	78.3	24.2	-
+ Dish unaries	60.4	82.1	48.9	74.3	78.2	26.0	49.3
number of states	78	53	138	69	49	-	31

Table 4.1: Visual recognition of SR, accuracy in % (mean over all intervals).

CRF, SMT, segmentation) on the validation set and fix them for all experiments; for our segmentation they are the initial segment size (60 frames), the similarity measure (cosine), and the termination threshold (0.982).

4.6.1 Visual recognition

We first evaluate the output of our visual recognition (SR) on MPII Cooking 2 dataset. We report accuracy of CRF nodes over all ground truth intervals on the test set in Table 4.1. The first line shows the results of Rohrbach *et al.* (2013b). We notice that the recognition of the handled object (in many cases the ingredient) is the most difficult, achieving only 36.8% compared to 59.1% or more for the other nodes. This lower performance is due to the larger number of states (last line, Table 4.1) and high intra-class variability of the ingredients. As a first step we add semantic unaries to the CRF. The performance improves for tools by 1.5% and objects by 2.3% compared to the first line. Next we add our hand centric color Sift features as second unary to the CRF nodes. This leads to an improvement for each node, especially for objects (+3.5%). Finally, we add a dish node to the CRF computing unaries with the approach from Chapter 5. This further improves recognition of *object* by an impressive 6.3%. In comparison to Rohrbach *et al.* (2013b) we achieve an overall improvement of 1.3% for *activity*, 2.5% for *tool*, 12.1% for *object* and 2.8% for *source* (line 1 vs 4). The percentage of segments where the complete SR tuple is correct (column "all") improves on each step and overall increases by 4.6%. In the next section we show that it leads to more consistent generated descriptions.

4.6.2 Multi-sentence generation

We first evaluate the effect of our improved visual recognition and the improvements in natural language sentence generation. We start with the TACoS dataset to allow a direct comparison to Rohrbach *et al.* (2013b), using the ground truth intervals provided by TACoS. The first line of Table 4.2 shows the results using the SR and SMT from Rohrbach *et al.* (2013b) (the best version, learning on predictions), which achieves a BLEU score of 23.2% when evaluated per sentence. This is an increase from

Approach	BLEU Sent	BLEU Desc	Read.	Corr.	Relev.
On gt intervals					
Rohrbach *et al.* (2013b)	23.2	55.7	2.5	3.3	2.8
Our SR	25.1	63.8	3.3	3.6	3.0
+ prob.	27.5	66.1	3.6	3.7	3.1
Human	36.0[3]	63.6[3]	4.4	4.9	4.8

Table 4.2: BLEU@4 in % on sentences (Sent) and full descriptions (Desc). Human judgments (Readability, Correctness, Relevance) from 1-5 (5 is best): TACoS.

22.1% reported by Rohrbach *et al.* (2013b) due to converting the TACoS corpus to past tense, making it more uniform. The BLEU score evaluated per description is 55.7%[2] and human judges score these descriptions with 2.5 for readability, 3.3 for correctness, and 2.8 for relevance on a scale from 1-5, where 5 is best. Using our improved SR (line 2 in Table 4.2) consistently improves the quality of the descriptions. Judges rate especially the readability much higher (+0.8) which is due to our increased consistency introduced by the dish node. Also correctness (+0.3) and relevance (+0.2) are rated higher, and the BLEU score improves by 1.9% and 8.1%.

Next, we evaluate the effect of using probabilistic input for SMT (line 3 in Table 4.2). Again all scores increase. Most notably the BLEU by 2.3% and readability by 0.3. While learning on predictions can recover from systematic errors of the visual recognition, using probabilistic input for SMT allows to recover from errors made at test time by choosing a less likely SR but more likely sentence according to the language model, e.g. *"The person got out a knife and a cutting board from the pot"* is correctly changed to *"The person took out a pot from the drawer"*. While the probabilistic input helps in many cases, we found that it sometimes generates sentences that diverge from the video content.

Now we validate our approach on the Detailed Descriptions of the TACoS Multi-Level corpus (Table 4.3). The upper part of the Table shows the results on the ground truth intervals provided by the collected descriptions. Here and in the following "Our" denotes the proposed approach with the improved SR and probabilistic input. The performance agrees with the results on TACoS. While we make significant improvements over Rohrbach *et al.* (2013b), there is still a gap to human description, showing the difficulty of the task and the dataset[3]. In the bottom part of Table 4.3 we evaluate our automatic segmentation and make the following observation: according to human judges, the performance drops only slightly compared to ground truth intervals and it is still higher than the result of Rohrbach *et al.* (2013b) on ground truth intervals. This indicates the good quality of our automatic segmentation.

In lines 2 and 5 of Table 4.3 we evaluate the impact of the linguistic post-

[2]The BLEU score per description is much higher than per sentence as the n-grams can be matched to the full descriptions.

[3]The BLEU score for human description is not fully comparable due to one reference less, which typically has a strong effect on the BLEU score.

Approach	BLEU Sent	BLEU Desc	Read.	Corr.	Relev.
On gt intervals					
Rohrbach *et al.* (2013b)	24.9	60.3	2.8	3.7	3.3
Our	26.9	65.1	3.2/3.4	4.1	3.6
Human	47.8[3]	62.3[3]	4.9	5.0	5.0
On our segmentation					
Rohrbach *et al.* (2013b)	-	48.3	2.5	3.5	3.1
Our	-	51.0	2.9/3.2	4.0	3.3

Table 4.3: BLEU@4 in % on sentences (Sent) and full descriptions (Desc). Human judgments (Readability, Correctness, Relevance) from 1-5 (5 is best): Detailed Descriptions.

Approach	BLEU Sent	BLEU Desc	Read.	Corr.	Relev.
On gt intervals					
Rohrbach *et al.* (2013b)	23.3	52.3	3.6	3.6	3.2
Our	24.7	54.6	3.8/4.0	3.9	3.7
Human	43.9[3]	56.6[3]	4.9	4.9	4.9
On our segmentation					
Our on Det Desc		53.4	-	-	-
Our on Short Desc		54.3	3.9/4.1	3.7	3.4

Table 4.4: BLEU@4 in % on sentences (Sent) and full descriptions (Desc). Human judgments (Readability, Correctness, Relevance) from 1-5 (5 is best): Short Descriptions.

processing (Sec. 4.5) on readability: the score improves from 3.2 to 3.4 and 2.9 to 3.2, respectively (all other reported numbers obtained without post-processing).

4.6.3 Multi-level generation

On the Short Descriptions the results on ground truth intervals (upper part of Table 4.4) agree with the previously discussed results. To produce a Short Description using our segmentation, we select the 3 most relevant segments, as described in Sec. 4.3. We decide for 3 segments as the average length of Short Descriptions is 3.5 sentences. In the last two lines of the Table 4.4 we compare training our system on the Detailed vs. Short Descriptions. As expected the language model trained on the Short Descriptions performs better (+0.9% BLEU) supporting our hypothesis that it is beneficial to learn a language model for a desired level of detail.

Table 4.5 shows the results for the Single Sentence Descriptions. The second line corresponds to our approach of using the dish prediction from the segmentation

Approach	BLEU	Read.	Corr.	Relev.
Upper bound				
Human	53.2[3]	4.9	4.9	4.7
On our segmentation				
Our on Sing Sent Desc	57.7	4.9	3.4	3.3
Our on Det Desc	15.2	-	-	-
Our on Short Desc	21.0	5.0	3.3	2.6

Table 4.5: BLEU@4 in % on sentences (Sent) and full descriptions (Desc). Human judgments (Readability, Correctness, Relevance) from 1-5 (5 is best): Single Sentence Descriptions.

to translate it into a sentence (Sec. 4.3.2). We also investigated a retrieval and a template baselines that rely on the dish prediction. They achieve lower BLEU score but nearly identical human judgments, indicating that the dish prediction is the most important aspect for the Single Sentence Descriptions. The last two lines compare the extractively produced descriptions, where the single (most relevant) segment was selected. The model trained on the Short Descriptions performs better than the one trained on the Detailed Descriptions, however it is far below the Single Sentence Descriptions with respect to relevance (-0.6) and BLEU (-36.7%), showing the significant difference between these types of descriptions.

4.7 CONCLUSION

This chapter addresses the challenging task of coherent multi-sentence video descriptions. We show that inferring the high level topic helps to ensure consistency across sentences. Using semantic unaries and hand centric features we improve visual recognition, especially for the most challenging semantic category, namely manipulated objects, which consecutively leads to better descriptions.

We also address the so far unexplored task of producing video descriptions at multiple levels of detail with our collected corpus of human descriptions. In an analysis we found that with decreasing length of description, the verbalized information is 'compressed' according to the topic of the video. Based on this we propose a method to extract most relevant segments of the video.

We believe that these results transfer to other domains as our approach is not specific to the kitchen setting. We plan to validate that as part of future work by exploring other domains. While we make a first step to couple visual recognition and language generation by using probabilistic input for SMT on the sentence level, we believe that a direction for future work is to reason jointly about visual recognition and language generation for multi-sentence descriptions.

In the following chapters we move to an open domain scenario. We tackle movie description in Chapter 5 and address local co-reference resolution of described people in Chapter 8.

5 MOVIE DESCRIPTION

In the previous chapter we have addressed video description in a cooking scenario. We now look at the open domain scenario, namely movies. Audio Description (AD) provides linguistic descriptions of movies and allows visually impaired people to follow a movie along with their peers. Such descriptions are by design mainly visual and thus naturally form an interesting data source for computer vision and computational linguistics. In this chapter we propose a novel dataset which contains transcribed ADs, which are temporally aligned to full length movies. In addition we also collected and aligned movie scripts used in prior work and compare the two sources of descriptions. We introduce the *Large Scale Movie Description Challenge* (LSMDC) which contains a parallel corpus of 128,118 sentences aligned to video clips from 200 movies (around 150 hours of video in total). The goal of the challenge is to automatically generate descriptions for the movie clips. First we characterize the dataset by benchmarking different approaches for generating video descriptions. Comparing ADs to scripts, we find that ADs are more visual and describe precisely what *is shown* rather than what *should happen* according to the scripts created prior to movie production. Furthermore, we present and compare the results of several teams who participated in the challenges organized in the context of two workshops at ICCV 2015 and ECCV 2016.

In Chapter 8 we present our approach to grounded video description, where besides describing video we also aim to localize the described people.

5.1 INTRODUCTION

Audio descriptions (ADs) make movies accessible to millions of blind or visually impaired people[1]. AD — sometimes also referred to as Descriptive Video Service (DVS) — provides an audio narrative of the "most important aspects of the visual information" (Salway, 2007), namely actions, gestures, scenes, and character appearance as can be seen in Figures 5.1 and 5.2. AD is prepared by trained describers and read by professional narrators. While more and more movies are audio transcribed, it may take up to 60 person-hours to describe a 2-hour movie (Lakritz and Salway, 2006), resulting in the fact that today only a small subset of movies and TV programs are available for the blind. Consequently, automating this process has the potential to greatly increase accessibility to this media content.

In addition to the benefits for the blind, generating descriptions for video is an

[1] In this chapter we refer for simplicity to "the blind" to account for all blind and visually impaired people which benefit from AD, knowing of the variety of visually impaired and that AD is not accessible to all.

AD: Abby gets in the basket.

Script: After a moment a frazzled Abby pops up in his place.

Mike leans over and sees how high they are.

Mike looks down to see – they are now fifteen feet above the ground.

Abby clasps her hands around his face and kisses him passionately.
For the first time in her life, she stops thinking and grabs Mike and kisses the hell out of him.

Figure 5.1: Audio description (AD) and movie script samples from the movie "Ugly Truth".

interesting task in itself, requiring the combination of core techniques from computer vision and computational linguistics. To understand the visual input one has to reliably recognize scenes, human activities, and participating objects. To generate a good description one has to decide what part of the visual information to verbalize, i.e. recognize what is salient.

Large datasets of objects (Deng *et al.*, 2009) and scenes (Xiao *et al.*, 2010; Zhou *et al.*, 2014) have had an important impact in computer vision and have significantly improved our ability to recognize objects and scenes. The combination of large datasets and convolutional neural networks (CNNs) has been particularly potent (Krizhevsky *et al.*, 2012). To be able to learn how to generate descriptions of visual content, parallel datasets of visual content paired with descriptions are indispensable (Rohrbach *et al.*, 2013b). While recently several large datasets have been released which provide images with descriptions (Hodosh *et al.*, 2014; Lin *et al.*, 2014b; Ordonez *et al.*, 2011), video description datasets focus on short video clips with single sentence descriptions and have a limited number of video clips (Xu *et al.*, 2016; Chen and Dolan, 2011) or are not publicly available (Over *et al.*, 2012). TACoS Multi-Level (Chapter 4) and YouCook (Das *et al.*, 2013) are exceptions as they provide multiple sentence descriptions and longer videos. While these corpora pose challenges in terms of fine-grained recognition, they are restricted to the cooking scenario. In contrast, movies are open domain and realistic, even though, as any other video source (e.g. YouTube or surveillance videos), they have their specific characteristics. ADs and scripts associated with movies provide rich multiple sentence descriptions. They even go beyond this by telling a story which means they facilitate the study of how to extract plots, the understanding of long term semantic dependencies and human interactions from both visual and textual data.

Figures 5.1 and 5.2 show examples of ADs and compare them to movie scripts. Scripts have been used for various tasks (Cour *et al.*, 2008; Duchenne *et al.*, 2009; Laptev *et al.*, 2008; Liang *et al.*, 2011; Marszalek *et al.*, 2009), but so far not for video description. The main reason for this is that automatic alignment frequently fails

AD: Buckbeak rears and attacks Malfoy.

Script: In a flash, Buckbeak's steely talons slash down.

Malfoy freezes.

Looks down at the blood blossoming on his robes.

Hagrid lifts Malfoy up.

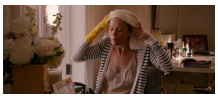
As Hagrid carries Malfoy away, the hippogriff gently nudges Harry.
Buckbeak whips around, raises its talons and - seeing Harry - lowers them.

AD: Another room, the wife and mother sits at a window with a towel over her hair.

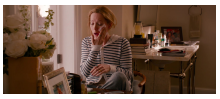
Script: Debbie opens a window and sneaks a cigarette.

She smokes a cigarette with a latex-gloved hand.

She holds her cigarette with a yellow dish washing glove.



Putting the cigarette out, she uncovers her hair, removes the glove and pops gum in her mouth.
She puts out the cigarette and goes through an elaborate routine of hiding the smell of smoke.



She pats her face and hands with a wipe, then sprays herself with perfume.

She *puts some weird oil in her hair* and uses a wet nap on her neck *and clothes and brushes her teeth*.

She pats her face and hands with a wipe, then sprays herself with perfume.

She sprays cologne *and walks through it.*

AD: They rush out onto the street.

Script: Valjean and Javert hurry out across the factory yard and down the muddy track beyond to discover -

A man is trapped under a cart.

A heavily laden cart has toppled onto the cart driver.

Valjean is crouched down beside him.

Valjean, *Javert and Javert's assistant* all hurry to help, but they can't get a proper purchase in the spongy ground.

Javert watches as Valjean places his shoulder under the shaft.
He throws himself under the cart at this higher end, and braces himself to lift it from beneath.

Javert's eyes narrow.

Javert stands back and looks on.

Figure 5.2: Audio description (AD) and movie script samples from the movies "Harry Potter and the Prisoner of Azkaban", "This is 40", and "Les Miserables". Typical mistakes contained in scripts marked in *red italic*.

due to the discrepancy between the movie and the script. As scripts are produced prior to the shooting of the movie they are frequently not as precise as the AD (Figure 5.2 shows some typical mistakes marked in red italic). A common case is that part of the sentence is correct, while another part contains incorrect/irrelevant information. As can be seen in the examples, AD narrations describe key visual elements of the video such as changes in the scene, people's appearance, gestures, actions, and their interaction with each other and the scene's objects in concise and precise language. Figure 5.3 shows the variability of AD data w.r.t. to verbs (actions) and corresponding scenes from the movies.

In this chapter we present a dataset which provides transcribed ADs, aligned to full length movies. AD narrations are carefully positioned within movies to fit in the natural pauses in the dialogue and are mixed with the original movie soundtrack

Figure 5.3: Some of the diverse verbs / actions present in our Large Scale Movie Description Challenge (LSMDC).

by professional post-production. To obtain ADs we retrieve audio streams from DVDs/Blu-ray disks, segment out the sections of the AD audio and transcribe them via a crowd-sourced transcription service. The ADs provide an initial temporal alignment, which however does not always cover the full activity in the video. We discuss a way to fully automate both audio-segmentation and temporal alignment, but also manually align each sentence to the movie for all the data. Therefore, in contrast to Salway (2007) and Salway *et al.* (2007), our dataset provides alignment to the actions in the video, rather than just to the audio track of the description. In addition we also mine existing movie scripts, pre-align them automatically, similar to Cour *et al.* (2008) and Laptev *et al.* (2008), and then manually align the sentences to the movie.

As a first study on our dataset we benchmark several approaches for movie description. We first examine nearest neighbor retrieval using diverse visual features which do not require any additional labels, but retrieve sentences from the training data. Second, we adapt the translation approach of Rohrbach *et al.* (2013b) by automatically extracting an intermediate semantic representation from the sentences using semantic parsing. Third, based on the success of Long Short-Term Memory networks (LSTMs) (Hochreiter and Schmidhuber, 1997) for the image captioning problem (Donahue *et al.*, 2015; Karpathy and Fei-Fei, 2015; Kiros *et al.*, 2015a; Vinyals *et al.*, 2015) we propose our approach*Visual-Labels*. It first builds robust visual classifiers which distinguish verbs, objects, and places extracted from weak sentence annotations. Then the visual classifiers form the input to an LSTM for generating movie descriptions.

The main contribution of this chapter is the Large Scale Movie Description Challenge (LSMDC)[2] which provides transcribed and aligned AD and script data sentences. The LSMDC was first presented at the Workshop "Describing and Understanding Video & The Large Scale Movie Description Challenge (LSMDC)", collocated with ICCV 2015. The second edition, LSMDC 2016, was presented at the

[2]https://sites.google.com/site/describingmovies/

"Joint Workshop on Storytelling with Images and Videos and Large Scale Movie Description and Understanding Challenge", collocated with ECCV 2016. Both challenges include the same public and blind test sets with an evaluation server[3] for automatic evaluation. LSMDC is based on the MPII Movie Description dataset (MPII-MD) and the Montreal Video Annotation Dataset (M-VAD) (Torabi *et al.*, 2015) which were initially collected independently but are presented jointly in this work. We detail the data collection and dataset properties in Section 5.3, which includes our approach to automatically collect and align AD data. In Section 5.4 we present several benchmark approaches for movie description, including our *Visual-Labels* approach which learns robust visual classifiers and generates description using an LSTM. In Section 5.5 we present an evaluation of the benchmark approaches on the M-VAD and MPII-MD datasets, analyzing the influence of the different design choices. Using automatic and human evaluation, we also show that our Visual-Labels approach outperforms prior work on both datasets. In Section 5.5.5 we perform an analysis of prior work and our approach to understand the challenges of the movie description task. In Section 5.6 we present and discuss the results of the LSMDC 2015 and LSMDC 2016.

5.2 RELATED WORK

Related work which focuses on video description approaches and datasets has been presented earlier in Chapter 2 of the thesis, specifically, in Section 2.2 and Section 2.1. In the following we shortly review recent approaches to image description.

Prior work on image description includes Farhadi *et al.* (2010b); Kulkarni *et al.* (2011); Kuznetsova *et al.* (2012); Li *et al.* (2011); Kuznetsova *et al.* (2014); Mitchell *et al.* (2012); Socher *et al.*. Recently image description has gained increased attention with work such as that of Chen and Zitnick (2015); Donahue *et al.* (2015); Fang *et al.* (2015); Karpathy and Fei-Fei (2015); Kiros *et al.* (2014, 2015a); Mao *et al.* (2015); Vinyals *et al.* (2015); Xu *et al.* (2015a). Much of the recent work has relied on Recurrent Neural Networks (RNNs) and in particular on Long Short-Term Memory networks (LSTMs). New datasets have been released, such as the Flickr30k (Young *et al.*, 2014) and MS COCO Captions (Chen *et al.*, 2015), where Chen *et al.* (2015) also presents a standardized protocol for image captioning evaluation. Other work has analyzed the performance of recent methods, e.g. Devlin *et al.* (2015) compare them with respect to the novelty of generated descriptions, while also exploring a nearest neighbor baseline that improves over recent methods.

5.3 DATASETS FOR MOVIE DESCRIPTION

In the following, we present how we collect our data for movie description and discuss its properties. The Large Scale Movie Description Challenge (LSMDC) is based

[3] https://competitions.codalab.org/competitions/6121

on two datasets which were originally collected independently. The MPII Movie Description Dataset (MPII-MD) was collected from Blu-ray movie data. It consists of AD and script data and uses sentence-level manual alignment of transcribed audio to the actions in the video (Section 5.3.1). In Section 5.3.2 we discuss how to fully automate AD audio segmentation and alignment for the Montreal Video Annotation Dataset (M-VAD), initially presented by Torabi *et al.* (2015). M-VAD was collected with DVD data quality and only relies on AD. Section 5.3.3 details the Large Scale Movie Description Challenge (LSMDC) which is based on M-VAD and MPII-MD, but also contains additional movies, and was set up as a challenge. It includes a submission server for evaluation on public and blind test sets. In Section 5.3.4 we present the detailed statistics of our datasets, also see Table 5.1. In Section 5.3.5 we compare our movie description data to other video description datasets.

5.3.1 The MPII Movie Description (MPII-MD) dataset

In the following we describe our approach behind the collection of ADs (Section 5.3.1.1) and script data (Section 5.3.1.2). Then we discuss how to manually align them to the video (Section 5.3.1.3) and which visual features we extracted from the video (Section 5.3.1.4).

5.3.1.1 *Collection of ADs*

We search for Blu-ray movies with ADs in the "Audio Description" section of the British Amazon[4] and select 55 movies of diverse genres (e.g. drama, comedy, action). As ADs are only available in audio format, we first retrieve the audio stream from the Blu-ray HD disks. We use MakeMKV[5] to extract a Blu-ray in the .mkv file format, and then XMediaRecode[6] to select and extract the audio streams from it. Then we semi-automatically segment out the sections of the AD audio (which is mixed with the original audio stream) with the approach described below. The audio segments are then transcribed by a crowd-sourced transcription service[7] that also provides us the time-stamps for each spoken sentence.

Semi-automatic segmentation of ADs. We are given two audio streams: the original audio and the one mixed with the AD. We first estimate the temporal alignment between the two as there might be a few time frames difference. The precise alignment is important to compute the similarity of both streams. Both steps (alignment and similarity) are estimated using the spectograms of the audio stream, which is computed using a Fast Fourier Transform (FFT). If the difference between the two audio streams is larger than a given threshold we assume the mixed stream contains AD at that point in time. We smooth this decision over time

[4] www.amazon.co.uk
[5] www.makemkv.com/
[6] www.xmedia-recode.de/
[7] CastingWords transcription service, http://castingwords.com/

using a minimum segment length of 1 second. The threshold was picked on a few sample movies, but had to be adjusted for each movie due to different mixing of the AD stream, different narrator voice level, and movie sound. While we found this semi-automatic approach sufficient when using a further manual alignment, we describe a fully automatic procedure in Section 5.3.2.

5.3.1.2 *Collection of script data*

In addition to the ADs we mine script web resources[8] and select 39 movie scripts. As starting point we use the movie scripts from "Hollywood2" (Marszalek *et al.*, 2009) that have highest alignment scores to their movie. We are also interested in comparing the two sources (movie scripts and ADs), so we are looking for the scripts labeled as "Final", "Shooting", or "Production Draft" where ADs are also available. We found that the "overlap" is quite narrow, so we analyze 11 such movies in our dataset. This way we end up with 50 movie scripts in total. We follow existing approaches (Cour *et al.*, 2008; Laptev *et al.*, 2008) to automatically align scripts to movies. First we parse the scripts, extending the method of (Laptev *et al.*, 2008) to handle scripts which deviate from the default format. Second, we extract the subtitles from the Blu-ray disks with SubtitleEdit[9]. It also allows for subtitle alignment and spellchecking. Then we use the dynamic programming method of (Laptev *et al.*, 2008) to align scripts to subtitles and infer the time-stamps for the description sentences. We select the sentences with a reliable alignment score (the ratio of matched words in the near-by monologues) of at least 0.5. The obtained sentences are then manually aligned to video in-house.

5.3.1.3 *Manual sentence-video alignment*

As the AD is added to the original audio stream between the dialogs, there might be a small misalignment between the time of speech and the corresponding visual content. Therefore, we manually align each sentence from ADs and scripts to the movie in-house. During the manual alignment we also filter out: a) sentences describing movie introduction/ending (production logo, cast, etc); b) texts read from the screen; c) irrelevant sentences describing something not present in the video; d) sentences related to audio/sounds/music. For the movie scripts, the reduction in number of words is about 19%, while for ADs it is under 4%. In the case of ADs, filtering mainly happens due to initial/ending movie intervals and transcribed dialogs (when shown as text). For the scripts, it is mainly attributed to irrelevant sentences. Note that we retain the sentences that are "alignable" but contain minor mistakes. If the manually aligned video clip is shorter than 2 seconds, we symmetrically expand it (from beginning and end) to be exactly 2 seconds long. In the following we refer to the obtained alignment as a "2-seconds-expanded" alignment.

[8] http://www.weeklyscript.com, http://www.simplyscripts.com, http://www.dailyscript.com, http://www.imsdb.com

[9] www.nikse.dk/SubtitleEdit/

5.3.1.4 *Visual features*

We extract video clips from the full movie based on the aligned sentence intervals. We also uniformly extract 10 frames from each video clip. As discussed earlier, ADs and scripts describe activities, objects and scenes (as well as emotions which we do not explicitly handle with these features, but they might still be captured, e.g. by the context or activities). In the following we briefly introduce the visual features computed on our data which are publicly available[10].

IDT We extract the improved dense trajectories compensated for camera motion (Wang and Schmid, 2013). For each feature (Trajectory, HOG, HOF, MBH) we create a codebook with 4,000 clusters and compute the corresponding histograms. We apply L1 normalization to the obtained histograms and use them as features.

LSDA We use the recent large scale object detection CNN (Hoffman *et al.*, 2014) which distinguishes 7,604 ImageNet (Deng *et al.*, 2009) classes. We run the detector on every second extracted frame (due to computational constraints). Within each frame we max-pool the network responses for all classes, then do mean-pooling over the frames within a video clip and use the result as a feature.

PLACES and HYBRID Finally, we use the recent scene classification CNNs (Zhou *et al.*, 2014) featuring 205 scene classes. We use both available networks, *Places-CNN* and *Hybrid-CNN*, where the first is trained on the Places dataset (Zhou *et al.*, 2014) only, while the second is additionally trained on the 1.2 million images of ImageNet (ILSVRC 2012) (Russakovsky *et al.*, 2015). We run the classifiers on all the extracted frames of our dataset. We mean-pool over the frames of each video clip, using the result as a feature.

5.3.2 The Montreal Video Annotation Dataset (M-VAD)

One of the main challenges in automating the construction of a video annotation dataset derived from AD audio is accurately segmenting the AD output, which is mixed with the original movie soundtrack. In Section 5.3.1.1 we have introduced a way of semi-automatic AD segmentation. In this section we describe a fully automatic method for AD narration isolation and video alignment. AD narrations are typically carefully placed within key locations of a movie and edited by a post-production supervisor for continuity. For example, when a scene changes rapidly, the narrator will speak multiple sentences without pauses. Such content should be kept together when describing that part of the movie. If a scene changes slowly, the narrator will instead describe the scene in one sentence, then pause for a moment, and later continue the description. By detecting those short pauses, we are able to align a movie with video descriptions automatically.

In the following we describe how we select the movies with AD for our dataset (Section 5.3.2.1) and detail our automatic approach to AD segmentation (Section 5.3.2.2). In Section 5.3.2.3 we discuss how to align AD to the video and obtain high quality AD transcripts.

[10] mpii.de/movie-description

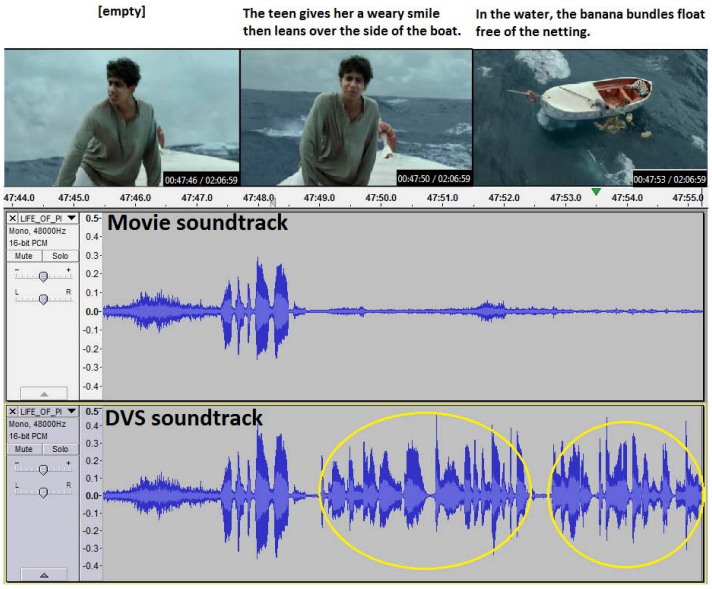


Figure 5.4: AD dataset collection. From the movie "Life of Pi". Line 2 and 3: Vocal isolation of movie and AD soundtrack. Second and third rows shows movie and AD audio signals after voice isolation. The two circles show the AD segments on the AD mono channel track. A pause (flat signal) between two AD narration parts shows the natural AD narration segmentation while the narrator stops and then continues describing the movie. We automatically segment AD audio based on these natural pauses. At first row, you can also see the transcription related to first and second AD narration parts on top of second and third image shots.

5.3.2.1 *Collection of ADs*

To search for movies with AD we use the movie lists provided in "An Initiative of the American Council of the Blind"[11] and "Media Access Group at WGBH"[12] websites, and buy them based on their availability and price. To extract video and audio from the DVDs we use the DVDfab[13] software.

5.3.2.2 *AD narrations segmentation using vocal isolation*

Despite the advantages offered by AD, creating a completely automated approach for extracting the relevant narration or annotation from the audio track and refining the alignment of the annotation with the video still poses some challenges. In the following, we discuss our automatic solution for AD narrations segmentation. We use two audio tracks included in DVDs: 1) the standard movie audio signal and 2) the standard movie audio mixed with AD narrations signal.

Vocal isolation techniques boost vocals, including dialogues and AD narrations

[11] http://www.acb.org/adp/movies.html
[12] http://main.wgbh.org/wgbh/pages/mag/dvsondvd.html
[13] http://www.dvdfab.cn/

while suppressing background movie sound in stereo signals. This technique is used widely in karaoke machines for stereo signals to remove the vocal track by reversing the phase of one channel to cancel out any signal perceived to come from the center while leaving the signals that are perceived as coming from the left or the right. The main reason for using vocal isolation for AD segmentation is based on the fact that AD narration is mixed in natural pauses in the dialogue. Hence, AD narration can only be present when there is no dialogue. In vocal isolated signals, whenever the narrator speaks, the movie signal is almost a flat line relative to the AD signal, allowing us to cleanly separate the narration by comparing the two signals. Figure 5.4 illustrates an example from the movie "Life of Pi", where in the original movie soundtrack there are sounds of ocean waves in the background.

Our approach has three main steps. First we isolate vocals, including dialogues and AD narrations. Second, we separate the AD narrations from dialogues. Finally, we apply a simple thresholding method to extract AD segment audio tracks.

We isolate vocals using Adobe Audition's center channel extractor[14] implementation to boost AD narrations and movie dialogues while suppressing movie background sounds on both AD and movie audio signals. We align the movie and AD audio signals by taking an FFT of the two audio signals, compute the cross-correlation, measure similarity for different offsets and select the offset which corresponds to peak cross-correlation. After alignment, we apply Least Mean Square (LMS) noise cancellation and subtract the AD mono squared signal from the movie mono squared signal in order to suppress dialogue in the AD signal. For the majority of movies on the market (among the 104 movies that we purchased, 12 movies have been mixed to the center of the audio signal, therefore we were not able to automatically align them), applying LMS results in cleaned AD narrations for the AD audio signal. Even in cases where the shapes of the standard movie audio signal and standard movie audio mixed with AD signal are very different - due to the AD mixing process - our procedure is sufficient for the automatic segmentation of AD narration.

Finally, we extract the AD audio tracks by detecting the beginning and end of AD narration segments in the AD audio signal (i.e. where the narrator starts and stops speaking) using a simple thresholding method that we applied to all DVDs without changing the threshold value. This is in contrast to the semi-automatic approach presented in Section 5.3.1.1, which requires individual adjustment of a threshold for each movie.

5.3.2.3 *Movie/AD alignment and professional transcription*

AD audio narration segments are time-stamped based on our automatic AD narration segmentation. In order to compensate for the potential 1-2 seconds misalignment between the AD narrator speaking and the corresponding scene in the movie, we automatically add two seconds to the end of each video clip. Also we discard all the transcriptions related to movie introduction/ending which are located at the beginning and the end of movies.

[14] creative.adobe.com/products/audition

	Unique Movies	Words	Sentences	Clips	Average length, sec.	Total length, h.
MPII-MD (AD)	55	330,086	37,272	37,266	4.2 (4.1)	44.0 (42.5)
MPII-MD (Movie script)	50	317,728	31,103	31,071	3.9 (3.6)	33.8 (31.1)
MPII-MD (Total)	94	647,814	68,375	68,337	4.1 (3.9)	77.8 (73.6)
M-VAD (AD)	92	502,926	55,904	46,589	6.2 -	84.6 -
LSMDC 15 Training	153	914.327	91,941	91,908	4.9 (4.8)	124.9 (121.4)
LSMDC 15 Validation	12	63,789	6,542	6,542	5.3 (5.2)	9.6 (9.4)
LSMDC 15&16 Public Test	17	87,150	10,053	10,053	4.2 (4.1)	11.7 (11.3)
LSMDC 15&16 Blind Test	20	83,766	9,578	9,578	4.5 (4.4)	12.0 (11.8)
LSMDC 15 (Total)	200	1,149,032	118,114	118,081	4.8 (4.7)	158.1 (153.9)
LSMDC 16 Training	153	922,918	101,079	101,046	4.1 (3.9)	114.9 (109.7)
LSMDC 16 Validation	12	63,321	7,408	7,408	4.1 (3.9)	8.4 (8.0)
LSMDC 15&16 Public Test	17	87,150	10,053	10,053	4.2 (4.1)	11.7 (11.3)
LSMDC 15&16 Blind Test	20	83,766	9,578	9,578	4.5 (4.4)	12.0 (11.8)
LSMDC 16 (Total)	200	1,157,155	128,118	128,085	4.1 (4.0)	147.0 (140.8)

Table 5.1: Movie description dataset statistics, see discussion in Section 5.3.4. For average/total length we report the "2-seconds-expanded" alignment, used in our work, and an actual manual alignment in brackets.

In order to obtain high quality text descriptions, the AD audio segments were transcribed with more than 98% transcription accuracy, using a professional transcription service[15]. These services use a combination of automatic speech recognition techniques and human transcription to produce a high quality transcription. Our audio narration isolation technique allows us to process the audio into small, well defined time segments and reduce the overall transcription effort and cost.

5.3.3 The Large Scale Movie Description Challenge (LSMDC)

To build our Large Scale Movie Description Challenge (LSMDC), we combine the M-VAD and MPII-MD datasets. We first identify the overlap between the two, so that the same movie does not appear in the training and test set of the joined dataset. We also exclude script-based movie alignments from the validation and test sets of MPII-MD. The datasets are then joined by combining the corresponding training, validation and test sets, see Table 5.1 for detailed statistics. The combined test set is used as a *public* test set of the challenge. We additionally acquired 20 more movies where we only release the video clips, but not the aligned sentences. They form the *blind* test set of the challenge and are only used for evaluation. We rely on the respective best aspects of M-VAD and MPII-MD for the public and blind test sets: we provide Blu-ray quality for them, use the automatic alignment/

[15]TranscribeMe professional transcription, http://transcribeme.com

Dataset	Vocab. size	Nouns	Verbs	Adjectives	Adverbs
MPII-MD	18,871	10,558	2,933	4,239	1,141
M-VAD	17,609	9,512	2,571	3,560	857
LSMDC 15	22,886	12,427	3,461	5,710	1,288
LSMDC 16	22,500	12,181	3,394	5,633	1,292

Table 5.2: Vocabulary and POS statistics (after word stemming) for our movie description datasets, see discussion in Section 5.3.4.

transcription described in Section 5.3.2 and clean them using a manual alignment as in Section 5.3.1.3. For the second edition of our challenge, LSMDC 2016, we also manually align the M-VAD validation and training sets and release them with Blu-ray quality. The manual alignment results in many multi-sentences descriptions to be split. Also the more precise alignment reduces the average clip length.

We set up the evaluation server[3] for the challenge using the Codalab[16] platform. The challenge data is available online[2]. We provide more information about the challenge setup and results in Section 5.6.

In addition to the description task, LSMDC 2016 includes three additional tracks, not discussed in this chapter. There is a movie annotation track which asks to select the correct sentence out of five in a multiple-choice test, a retrieval track which asks to retrieve the correct test clip for a given sentence, and a fill-in-the-blank track which requires to predict a missing word in a given description and the corresponding clip. Torabi *et al.* (2016) provide more details about the annotation and the retrieval tasks, while Maharaj *et al.* (2017) describe the movie fill-in-the-blank task.

5.3.4 Movie description dataset statistics

Table 5.1 presents statistics for the number of words, sentences and clips in our movie description corpora. We also report the average/total length of the annotated time intervals. We report both, the "2-seconds-expanded" clip alignment (see Section 5.3.1.3) and the actual clip alignment in brackets. In total MPII-MD contains 68,337 clips and 68,375 sentences (rarely multiple sentences might refer to the same video clip), while M-VAD includes 46,589 clips and 55,904 sentences.

Our combined LSMDC 2015 dataset contains over 118K sentence-clips pairs and 158 hours of video. The training/validation/public-/blind-test sets contain 91,908, 6,542, 10,053 and 9,578 video clips respectively. This split balances movie genres within each set, which is motivated by the fact that the vocabulary used to describe, say, an action movie could be very different from the vocabulary used in a comedy movie. After manual alignment of the training/validation sets, the new LSMDC 2016 contains 101,046 training clips, 7,408 validation clips and 128K clips in total.

[16] https://codalab.org/

Dataset	Multi-sent.	Domain	Sentence source	Videos	Clips	Sent.	Length, h
YouCook (Das *et al.*, 2013)	x	cook.	crowd	88	-	2,668	2.3
TACoS (Regneri *et al.*, 2013)	x	cook.	crowd	127	7,206	18,227	10.1
TACoS Multi-Level (ours)	x	cook.	crowd	185	24,764	74,828	15.8
MSVD (Chen and Dolan, 2011)		open	crowd	-	1,970	70,028	5.3
TGIF (Li *et al.*, 2016)		open	crowd	-	100,000	125,781	≈86.1
MSR-VTT (Xu *et al.*, 2016)		open	crowd	7,180	10,000	200,000	41.2
VTW (Zeng *et al.*, 2016)	x	open	crowd/prof.	18,100	-	44,613	213.2
M-VAD (ours)	x	open	professional	92	46,589	55,904	84.6
MPII-MD (ours)	x	open	professional	94	68,337	68,375	77.8
LSMDC 15 (ours)	x	open	professional	200	118,081	118,114	158.1
LSMDC 16 (ours)	x	open	professional	200	128,085	128,118	147.0

Table 5.3: Comparison of video description datasets. Discussion see Section 5.3.5.

Table 5.2 illustrates the vocabulary size, number of nouns, verbs, adjectives, and adverbs in each respective dataset. To compute the part of speech statistics for our corpora we tag and stem all words in the datasets with the Standford Part-Of-Speech (POS) tagger and stemmer toolbox (Toutanova *et al.*, 2003), then we compute the frequency of stemmed words in the corpora. It is important to notice that in our computation each word and its variations in corpora is counted once since we applied stemmer. Interesting observation on statistics is that e.g. the number of adjectives is larger than the number of verbs, which shows that the AD is describing the characteristics of visual elements in the movie in high detail.

5.3.5 Comparison to other video description datasets

We compare our corpus to other existing parallel video corpora in Table 5.3. We look at the following properties: availability of multi-sentence descriptions (long videos described continuously with multiple sentences), data domain, source of descriptions and dataset size. The main limitations of prior datasets include the coverage of a single domain (Das *et al.*, 2013; Regneri *et al.*, 2013) and having a limited number of video clips (Chen and Dolan, 2011). Recently, a few video description datasets have been proposed, namely MSR-VTT (Xu *et al.*, 2016), TGIF (Li *et al.*, 2016) and VTW (Zeng *et al.*, 2016). Similar to MSVD dataset (Chen and Dolan, 2011), MSR-VTT is based on YouTube clips. While it has a large number of sentence descriptions (200K) it is still rather small in terms of the number of video clips (10K). TGIF is a large dataset of 100k image sequences (GIFs) with associated descriptions. VTW is a dataset which focuses on longer YouTube videos (1.5 minutes on average) and aims to generate concise video titles from user provided descriptions as well as editor provided titles. All these datasets are similar in that they contain web-videos, while our proposed dataset focuses on movies. Similar to e.g. VTW, our dataset has a

"multi-sentence" property, making it possible to study multi-sentence description or understanding stories and plots.

5.4 APPROACHES FOR MOVIE DESCRIPTION

Given a training corpus of aligned videos and sentences we want to describe a new unseen test video. In this section we discuss two approaches to the video description task that we benchmark on our proposed datasets. Our first approach in Section 5.4.1 is based on the statistical machine translation (SMT) approach of (Rohrbach *et al.*, 2013b). Our second approach (Section 5.4.2) learns to generate descriptions using Long Short-Term Memory network (LSTM). For the first step both approaches rely on visual classifiers learned on annotations (labels) extracted from natural language descriptions using our semantic parser (Section 5.4.1.1). While the first approach does not differentiate which features to use for different labels, our second approach defines different semantic groups of labels and uses most relevant visual features for each group. For this reason we refer to this approach as *Visual-Labels*. Next, the first approach uses the classifier scores as input to a CRF to predict a semantic representation (SR) (SUBJECT, VERB, OBJECT, LOCATION), and then translates it into a sentence with SMT. On the other hand, our second approach directly provides the classifier scores as input to an LSTM which generates a sentence based on them. Figure 5.5 shows an overview of the two discussed approaches.

5.4.1 Semantic parsing + Statistical Machine Translation (SMT)

As our first approach we adapt the two-step translation approach of (Rohrbach *et al.*, 2013b). As a first step it trains the visual classifiers based on manually annotated tuples e.g. $\langle cut, knife, tomato \rangle$ provided with the video. Then it trains a CRF which aims to predict such tuple, or semantic representation (SR), from a video clip. At a second step, the Statistical Machine Translation (SMT) (Koehn *et al.*, 2007) is used to translate the obtained SR into a natural language sentence, e.g. *"The person cuts a tomato with a knife"*, see Figure 5.5(a). While we cannot rely on a manually annotated SR as in (Rohrbach *et al.*, 2013b), we automatically mine the SR from sentences using semantic parsing which we introduce in this section.

5.4.1.1 *Semantic parsing*

Learning from a parallel corpus of videos and natural language sentences is challenging when no annotated intermediate representation is available. In this section we introduce our approach to exploit the sentences using semantic parsing. The proposed method automatically extracts intermediate semantic representations (SRs) from the natural sentences.

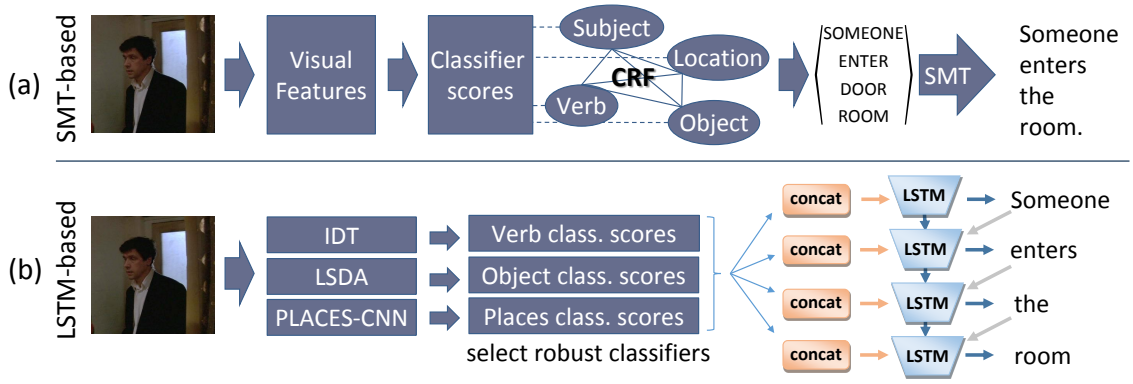


Figure 5.5: Overview of our movie description approaches: (a) SMT-based approach, adapted from (Rohrbach *et al.*, 2013b); (b) our proposed LSTM-based approach.

Phrase	WordNet Mapping	VerbNet Mapping	Desired Frame
the man	man_n^1	Agent.animate	Agent: man_n^1
begin to shoot	shoot_v^4	shoot_v^4	Action: shoot_v^4
a video	video_n^2	Patient.inanimate	Patient: video_n^2
in	in	PP.in	
the moving bus	bus_n^1	NP.Location. solid	Location: moving bus_n^1

Table 5.4: Semantic parse for "*He began to shoot a video in the moving bus*". For discussion, see Section 5.4.1.1.

Approach. We lift the words in a sentence to a semantic space of roles and WordNet (Fellbaum, 1998) senses by performing SRL (Semantic Role Labeling) and WSD (Word Sense Disambiguation). For an example, refer to Table 5.4 where the desired outcome of SRL and WSD on the input sentence "*He shot a video in the moving bus*" is "Agent: man_n^1, Action: shoot_v^4, Patient: video_n^2, Location: bus_n^1". Here, e.g. shoot_v^4 refers to the fourth verb sense of shoot in WordNet[17]. This is similar to the semantic representation of Rohrbach *et al.* (2013b), except that those semantic frames were constructed manually while we construct them automatically and our role fillers are additionally sense disambiguated. As verbs are known to have high ambiguity, the

[17]The WordNet senses for *shoot* and *video* are:

- shoot_v^1: hit with missile ... video_n^1: picture in TV
- shoot_v^2: kill by missile ... video_n^2: a recording ...
- ... ...
- shoot_v^4: make a film ... video_n^4: broadcasting ...

where, shoot_v^1 refers to the first verb (v) sense of shoot.

disambiguation step will provide clearer representations (corresponding WordNet sense) of a large set of verbs present in movie descriptions.

We start by decomposing the typically long sentences present in movie descriptions into smaller clauses using the ClausIE tool (Del Corro and Gemulla, 2013). For example, "*he shot and modified the video*" is split into two clauses "*he shot the video*" and "*he modified the video*"). We then use the OpenNLP tool suite[18] to chunk every clause into phrases. These chunks are disambiguated to their WordNet senses[17] by enabling a state-of-the-art WSD system called IMS (Zhong and Ng, 2010), to additionally disambiguate phrases that are not present in WordNet and thus, out of reach for IMS. We identify and disambiguate the head word of an out of WordNet phrase, e.g. the moving bus to the proper WordNet sense bus_n^1 via IMS. In this way we make an extension to IMS so it works for phrases and not just words. We link verb phrases to the proper sense of its head word in WordNet (e.g. begin to shoot to shoot_v^4). The phrasal verbs such as e.g. *"pick up"* or *"turn off"* are preserved as long as they exist in WordNet.

Having estimated WordNet senses for the words and phrases, we need to assign semantic role labels to them. Typical SRL systems require large amounts of training data, which we do not possess for the movie domain. Therefore, we propose leveraging VerbNet (Kipper *et al.*, 2006; Schuler *et al.*, 2009), a manually curated high-quality linguistic resource for English verbs that supplements WordNet verb senses with syntactic frames and semantic roles, as a distant signal to assign role labels. Every VerbNet verb sense comes with a syntactic frame e.g. for shoot_v^4, the syntactic frame is NP V NP. VerbNet also provides a role restriction on the arguments of the roles e.g. for shoot_v^3 (sense killing), the role restriction is *Agent.animate V Patient.**animate** PP Instrument.solid*. For another sense, shoot_v^4 (sense film), the semantic restriction is *Agent.animate V Patient.**inanimate***. We ensure that the selected WordNet verb sense adheres to both the syntactic frame and the semantic role restriction provided by VerbNet. For example, in Table 5.4, because video_n^2 is a type of inanimate object (inferred through WordNet noun taxonomy), this sense correctly adheres to the VerbNet role restriction. We can now simply apply the VerbNet suggested role Patient to video_n^2.

Semantic representation. Although VerbNet is helpful as a distant signal to disambiguate and perform semantic role labeling, VerbNet contains over 20 roles and not all of them are general or can be recognized reliably. Therefore, for simplicity, we generalize and group them to get the SUBJECT, VERB, OBJECT, LOCATION roles. For example, the roles patient, recepient, and, benefeciary are generalized to OBJECT. We explore two approaches to obtain the labels based on the output of the semantic parser. First is to use the extracted text chunks directly as labels. Second is to use the corresponding senses as labels (and therefore group multiple text labels). In the following we refer to these as *text-* and *sense-labels*. Thus from each sentence we extract a semantic representation in a form of (SUBJECT, VERB, OBJECT, LOCATION).

[18]OpenNLP tool suite: http://opennlp.sourceforge.net/

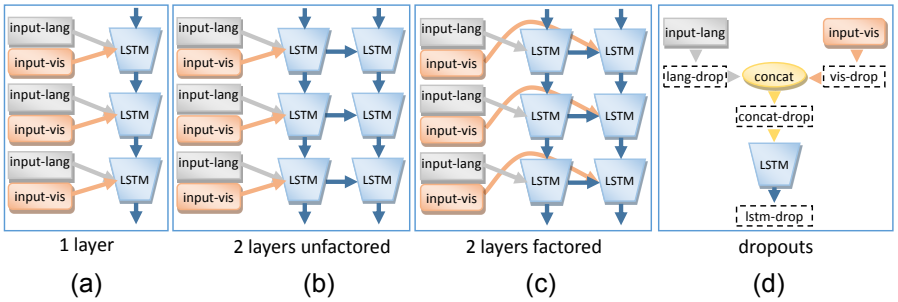


Figure 5.6: (a-c) LSTM architectures. (d) Variants of placing the dropout layer.

5.4.1.2 *Statistical Machine Translation (SMT)*

For the sentence generation we build on the two-step translation approach of (Rohrbach *et al.*, 2013b). As the first step it learns a mapping from the visual input to the semantic representation (SR), modeling pairwise dependencies in a CRF using visual classifiers as unaries. The unaries are trained using an SVM on dense trajectories (Wang and Schmid, 2013). In the second step it translates the SR to a sentence using Statistical Machine Translation (SMT) (Koehn *et al.*, 2007). For this the approach uses a concatenated SR as input language, e.g. *cut knife tomato*, and natural sentence as output language, e.g. *The person slices the tomato.* We obtain the SR automatically from the semantic parser, as described above, Section 5.4.1.1. In addition to dense trajectories we use the features described in Section 5.3.1.4.

5.4.2 Visual labels + LSTM

Next we present our two-step LSTM-based approach. The first step performs visual recognition using the visual classifiers which we train according to labels' semantics and "visuality". The second step generates textual descriptions using an LSTM network (see Figure 5.5(b)). We explore various design choices for building and training the LSTM.

5.4.2.1 *Robust visual classifiers*

For training we rely on a parallel corpus of videos and weak sentence annotations. As before (see Section 5.4.1) we parse the sentences to obtain a set of labels (single words or short phrases, e.g. *look up*) to train visual classifiers. However, this time we aim to select the most *visual* labels which can be robustly recognized. In order to do that we take three steps.

Avoiding parser failure. Not all sentences can be parsed successfully, as e.g. some sentences are incomplete or grammatically incorrect. To avoid loosing the potential labels in these sentences, we match our set of initial labels to the sentences which

the parser failed to process. Specifically, we do a simple word matching, i.e. if the label is found in the sentence, we consider this sentence as a positive for the label.

Semantic groups. Our labels correspond to different semantic groups. In this work we consider three most important groups: verbs, objects and places. We propose to treat each label group independently. First, we rely on a different representation for each semantic group, which is targeted to the specific group. Namely we use the activity recognition features Improved Dense Trajectories (DT) for verbs, LSDA scores for objects and PLACES-CNN scores for places. Second, we train one-vs-all SVM classifiers for each group separately. The intuition behind this is to avoid "wrong negatives" (e.g. using *object* "bed" as negative for *place* "bedroom").

Visual labels. Now, how do we select *visual* labels for our semantic groups? In order to find the verbs among the labels we rely on our semantic parser (Section 5.4.1.1). Next, we look up the list of "places" used in (Zhou *et al.*, 2014) and search for corresponding words among our labels. We look up the object classes used in (Hoffman *et al.*, 2014) and search for these "objects", as well as their base forms (e.g. "domestic cat" and "cat"). We discard all the labels that do not belong to any of our three groups of interest as we assume that they are likely not visual and thus are difficult to recognize. Finally, we discard labels which the classifiers could not learn reliably, as these are likely noisy or not visual. For this we require the classifiers to have certain minimum area under the ROC-curve (Receiver Operating Characteristic). We estimate a threshold for the ROC values on a validation set. We empirically evaluate this as well as all other design choices of our approach in Section 5.5.4.2.

5.4.2.2 *LSTM for sentence generation*

We rely on the basic LSTM architecture proposed in (Donahue *et al.*, 2015) for video description. At each time step an LSTM generates a word and receives the visual classifiers (*input-vis*) as well as as the previous generated word (*input-lang*) as input (see Figure 5.6(a)). We encode each word with a one-hot-vector according to its index in a dictionary and project it in a lower dimensional embedding. The embedding is jointly learned during training of the LSTM. We feed in the classifier scores as input to the LSTM which is equivalent to the best variant proposed in (Donahue *et al.*, 2015). We analyze the following aspects for this architecture:

Layer structure. We compare a 1-layer architecture with a 2-layer architecture. In the 2-layer architecture, the output of the first layer is used as input for the second layer (Figure 5.6b) and was used by (Donahue *et al.*, 2015) for video description. Additionally we also compare to a 2-layer factored architecture of (Donahue *et al.*, 2015), where the first layer only gets the language as input and the second layer gets the output of the first as well as the visual input.

Dropout placement. To learn a more robust network which is less likely to overfit we rely on a dropout (Hinton *et al.*, 2012), i.e. a ratio r of randomly selected units is set to 0 during training (while all others are multiplied with $1/r$). We explore different ways to place dropout in the network, i.e. either for language input (*lang-drop*) or visual (*vis-drop*) input only, for both inputs (*concat-drop*) or for the LSTM output (*lstm-drop*), see Figure 5.6(d).

5.5 EVALUATION ON MPII-MD AND M-VAD

In this section we evaluate and provide more insights about our movie description datasets MPII-MD and M-VAD. We compare ADs to movie scripts (Section 5.5.1), present a short evaluation of our semantic parser (Section 5.5.2), present the automatic and human evaluation metrics for description (Section 5.5.3) and then benchmark the approaches to video description introduced in Section 5.4 as well as other related work. We conclude this section with an analysis of the different approaches (Section 5.5.5).

In Section 5.6 we will extend this discussion to the results of the Large Scale Movie Description Challenge.

5.5.1 Comparison of AD vs. script data

We compare the AD and script data using 11 movies from the MPII-MD dataset where both are available (see Section 5.3.1.2). For these movies we select the overlapping time intervals with an intersection over union overlap of at least 75%, which results in 279 sentence pairs, we remove 2 pairs which have idendical sentences. We ask humans via Amazon Mechanical Turk (AMT) to compare the sentences with respect to their correctness and relevance to the video, using both video intervals as a reference (one at a time). Each task was completed by 5 different human subjects, covering 2,770 tasks done in total. Table 5.5 presents the results of this evaluation. AD is ranked as more correct and relevant in about $2/3$ of the cases (i.e. there is margin of about 33%). Looking at the more strict evaluation where at least 4 out of 5 judges agree (in brackets in Table 5.5) there is still a significant margin of 24.5% between ADs and movie scripts for Correctness, and 28.1% for Relevance. One can assume that in the cases of lower agreement the descriptions are probably of similar quality. This evaluation supports our intuition that scrips contain mistakes and irrelevant content even after being cleaned up and manually aligned.

5.5.2 Semantic parser evaluation

We empirically evaluate the various components of the semantic parsing pipeline, namely, clause splitting (Clause), POS tagging and chunking (NLP), semantic role labeling (Roles), and, word sense disambiguation (WSD). We randomly sample 101 sentences from the MPII-MD dataset over which we perform semantic parsing and

	Correctness	Relevance
Movie scripts	33.9 (11.2)	33.4 (16.8)
ADs	66.1 (35.7)	66.6 (44.9)

Table 5.5: Human evaluation of movie scripts and ADs: which sentence is more correct/relevant with respect to the video (forced choice). Majority vote of 5 judges in %. In brackets: at least 4 out of 5 judges agree. See also Section 5.5.1.

Corpus	Clause	NLP	Roles	WSD
MPII-MD	0.89	0.62	0.86	0.7

Table 5.6: Semantic parser accuracy on MPII-MD. Discussion in Section 5.5.2.

log the outputs at various stages of the pipeline (similar to Table 5.4). We let three human judges evaluate the results for every token in the clause (similar to evaluating every row in Table 5.4) with a correct/ incorrect label. From this data, we consider the majority vote for every token in the sentence (i.e. at least 2 out of 3 judges must agree). For a given clause, we assign a score of 1 to a component if the component made no mistake for the entire clause. For example, "Roles" gets a score of 1 if, according to majority vote from the judges, we correctly estimate all semantic roles in the clause. Table 5.6 reports the average accuracy of the components over 130 clauses (generated from 101 sentences).

It is evident that the poorest performing parts are the NLP and the WSD components. Some of the NLP mistakes arise due to incorrect POS tagging. WSD is considered a hard problem and when the dataset contains rare words, the performance is severely affected.

5.5.3 Evaluation metrics for description

In this section we describe how we evaluate the generated descriptions using automatic and human evaluation.

5.5.3.1 *Automatic metrics*

For automatic evaluation we rely on the MS COCO Caption Evaluation API[19]. The automatic evaluation measures include BLEU-1,-2,-3,-4 (Papineni *et al.*, 2002), METEOR (Lavie, 2014), ROUGE-L (Lin, 2004), and CIDEr (Vedantam *et al.*, 2015). We also use the recently proposed evaluation measure SPICE (Anderson *et al.*, 2016), which aims to compare the semantic content of two descriptions, by matching the information contained in dependency parse trees for both descriptions. While we report all measures for the final evaluation in the LSMDC (Section 5.6), we focus our

[19]https://github.com/tylin/coco-caption

discussion on METEOR and CIDEr scores in the preliminary evaluations in this section. According to (Elliott and Keller, 2013; Vedantam *et al.*, 2015), METEOR/CIDEr supersede previously used measures in terms of agreement with human judgments.

5.5.3.2 *Human evaluation*

For the human evaluation we rely on a ranking approach, i.e. human judges are given multiple descriptions from different systems, and are asked to rank them with respect to the following criteria: correctness, relevance, and grammar, motivated by prior work Rohrbach *et al.* (2013b) and on the other hand we asked human judges to rank sentences for "how helpful they would be for a blind person to understand what is happening in the movie". The AMT workers are given randomized sentences, and, in addition to some general instruction, the following definitions:

Grammar. "Rank grammatical correctness of sentences: Judge the fluency and readability of the sentence (independently of the correctness with respect to the video)."

Correctness. "Rank correctness of sentences: For which sentence is the content more correct with respect to the video (independent if it is complete, i.e. describes everything), independent of the grammatical correctness."

Relevance. "Rank relevance of sentences: Which sentence contains the more salient (i.e. relevant, important) events/objects of the video?"

Helpful for the blind. In the LSMDC evaluation we introduce a new measure, which should capture how useful a description would be for blind people: "Rank the sentences according to how useful they would be for a blind person which would like to understand/follow the movie without seeing it."

5.5.4 Movie description evaluation

As the collected text data comes from the movie context, it contains a lot of information specific to the plot, such as names of the characters. We pre-process each sentence in the corpus, transforming the names to "Someone" or "people" (in case of plural).

We first analyze the performance of the proposed approaches on the MPII-MD dataset, and then evaluate the best version on the M-VAD dataset. For MPII-MD we split the 11 movies with associated scripts and ADs (in total 22 alignments, see Section 5.3.1.2) into validation set (8) and test set (14). The other 83 movies are used for training. On M-VAD we use 10 movies for testing, 10 for validation and 72 for training.

	METEOR
SMT with our sense-labels	
IDT 30	4.93
IDT 100	5.12
Combi 100	5.19
SMT with our text-labels	
IDT 30	5.59
IDT 100	5.51
Combi 100	5.42

Table 5.7: Video description performance of different SMT versions on MPII-MD. Discussion in Section 5.5.4.1.

5.5.4.1 *Semantic parsing + SMT*

Table 5.7 summarizes results of multiple variants of the SMT approach when using the SR from our semantic parser. "Combi" refers to combining IDT, HYBRID, and PLACES as unaries in the CRF. We did not add LSDA as we found that it reduces the performance of the CRF. After extracting the labels we select the ones which appear at least 30 or 100 times as our visual attributes. Overall, we observe similar performance in all cases, with slightly better results for text-labels than sense-labels. This can be attributed to sense disambiguation errors of the semantic parser. In the following we use the "IDT 30" model, which achieves the highest score of 5.59, and denote it as "SMT-Best".

5.5.4.2 *Visual labels + LSTM*

We start with exploring different design choices of our approach. We build on the labels discovered by the semantic parser. To learn classifiers we select the labels that appear at least 30 times, resulting in 1,263 labels. The parser additionally tells us whether the label is a verb. The LSTM output/hidden unit as well as memory cell have each 500 dimensions.

Robust visual classifiers. We first analyze our proposal to consider groups of labels to learn different classifiers and also to use different visual representations for these groups (see Section 5.4.2). In Table 5.8 we evaluate our generated sentences using different input features to the LSTM on the validation set of MPII-MD. In our baseline, in the top part of Table 5.8, we use the same visual descriptors for all labels. The PLACES feature is best with 7.10 METEOR. Combination by stacking all features (IDT + LSDA + PLACES) improves further to 7.24 METEOR. The second part of the table demonstrates the effect of introducing different semantic label groups. We first split the labels into "Verbs" and all others. Given that some labels appear in both roles, the total number of labels increases to 1328 (line 5). We compare two settings of training the classifiers: "Retrieved" (we retrieve the classifier scores from

Approach	Labels	Classifiers (METEOR in %) Retrieved	Trained
Baseline: all labels treated the same way			
(1) IDT	1263	-	6.73
(2) LSDA	1263	-	7.07
(3) PLACES	1263	-	7.10
(4) IDT+LSDA+PLACES	1263	-	7.24
Visual labels			
(5) Verbs(IDT), Others(LSDA)	1328	7.08	7.27
(6) Verbs(IDT), Places(PLACES), Others(LSDA)	1328	7.09	7.39
(7) Verbs(IDT), Places(PLACES), Objects(LSDA)	913	7.10	7.48
(8) + restriction to labels with $ROC \geq 0.7$	263	7.41	**7.54**
Baseline: all labels treated the same way, labels from (8)			
(9) IDT+LSDA+PLACES	263	7.16	7.20

Table 5.8: Comparison of different choices of labels and visual classifiers. All results reported on the validation set of MPII-MD. For discussion see Section 5.5.4.2. Bold indicates the best performing variant in the table.

the classifiers trained in the previous step), "Trained" (we train the SVMs specifically for each label type, e.g. "Verbs"). Next, we further divide the non-"Verb" labels into "Places" and "Others"(line 6), and finally into "Places" and "Objects"(line 7). We discard the unused labels and end up with 913 labels. Out of these labels, we select the labels where the classifier obtains a ROC higher or equal to 0.7 (threshold selected experimentally). After this we obtain 263 labels and the best performance in the "Trained" setting (line 8). To support our intuition about the importance of the label discrimination (i.e. using different features for different semantic groups of labels), we propose another baseline (line 9). Here we use the same set of 263 labels but provide the same feature for all of them, namely the best performing combination IDT + LSDA + PLACES. As we see, this results in an inferior performance.

We make several observations from Table 5.8 which lead to robust visual classifiers from the weak sentence annotations. a) It is beneficial to select features based on the label semantics. b) Training one-vs-all SVMs for specific label groups consistently improves the performance as it avoids "wrong" negatives. c) Focusing on more "visual" labels helps: we reduce the LSTM input dimensionality to 263 while improving the performance.

LSTM design choices. Now, as described in Section 5.4.2.2, we look at different LSTM architectures and training configurations. In the following we use the best performing "Visual Labels" approach, Table 5.8, line (8).

We start with examining the architecture, where we explore different configurations of LSTM and dropout layers. Table 5.9(a) shows the performance of three different networks: "1 layer", "2 layers unfactored" and "2 layers factored" intro-

Architecture	*METEOR*
1 layer	**7.54**
2 layers unfact.	**7.54**
2 layers fact.	7.41

(a) LSTM architectures (fixed parameters: LSTM-drop, dropout 0.5)

Dropout	*METEOR*
no dropout	7.19
lang-drop	7.13
vis-drop	7.34
concat-drop	7.29
lstm-drop	**7.54**

(b) Dropout strategies (fixed parameters: 1-layer, dropout 0.5)

Dropout ratio	*METEOR*
r=0.1	7.22
r=0.25	7.42
r=0.5	**7.54**
r=0.75	7.46

(c) Dropout ratios (fixed parameters: 1-layer, LSTM-drop)

Table 5.9: LSTM architectures, dropout strategies and dropout ratios, MPII-MD val set. Labels, classifiers as Table 5.8, line (8). For discussion see Section 5.5.4.2. Bold indicates the best performing variant in the table.

Approach	*METEOR*
lr=0.005, step=2000	7.30
lr=0.01, step=2000	7.54
lr=0.02, step=2000	7.51
lr=0.005, step=4000	7.49
lr=0.01, step=4000	**7.59**
lr=0.02, step=4000	7.28

(a) Base learning rates

Approach	*METEOR*
step=2000, iter=25,000	7.54
step=4000, iter=25,000	**7.59**
step=6000, iter=25,000	7.40
step=8000, iter=25,000	7.32
poly, pow=0.5, maxiter=25,000	7.36
poly, pow=0.5, maxiter=10,000	7.45
poly, pow=0.7, maxiter=25,000	7.43
poly, pow=0.7, maxiter=10,000	7.43

(b) Learning strategies with lr=0.01

Table 5.10: (a) Comparison of different base learning rates, network trained for 25,000 iterations. (b) Comparison of different learning strategies with lr=0.01. Labels and classifiers from Table 5.8 (8). All results reported on the MPII-MD val set.

duced in Section 5.4.2.2. As we see, the "1 layer" and "2 layers unfactored" perform equally well, while "2 layers factored" is inferior to them. In the following experiments we use the simpler "1 layer" network. We then compare different dropout placements as illustrated in (Table 5.9(b)). We obtain the best result when applying dropout after the LSTM layer ("lstm-drop"), while having no dropout or applying it only to language leads to stronger over-fitting to the visual features. Putting dropout after the LSTM (and prior to a final prediction layer) makes the entire system more robust. As for the best dropout ratio, we find that 0.5 works best with lstm-dropout (Table 5.9(c)).

Next we compare different learning rates (Table 5.10 (a)) and learning strategies (Table 5.10 (b)). We find that the best learning rate in the step-based learning is 0.01, while step size 4000 slightly improves over step size 2000 (which we used in Table 5.8). We explore an alternative learning strategy, namely decreasing learning rate according to a polynomial decay. We experiment with different exponents (0.5 and 0.7) and numbers of iterations (25K and 10K), using the base-learning rate 0.01. Our results show that the step-based learning is superior to the polynomial learning.

Approach	*METEOR*
1 net: lr 0.01, step 2000, iter=25,000	7.54
ensemble of 3 nets	7.52
1 net: lr 0.01, step 4000, iter=25,000	7.59
ensemble of 3 nets	7.68
1 net: lr 0.01, step 4000, iter=15,000	7.55
ensemble of 3 nets	**7.72**

Table 5.11: Ensembles of networks with different random initializations. All results reported on the validation set of MPII-MD.

In most of experiments we trained our networks for 25,000 iterations. After looking at the METEOR performance for intermediate iterations we found that for the step size 4000 at iteration 15,000 we achieve best performance overall. Additionally we train multiple LSTMs with different random orderings of the training data. In our experiments we combine three in an ensemble, averaging the resulting word predictions. In most cases the ensemble improves over the single networks in terms of METEOR score (see Table 5.11).

To summarize, the most important aspects that decrease over-fitting and lead to better sentence generation are: (a) a correct learning rate and step size, (b) dropout after the LSTM layer, (c) choosing the training iteration based on METEOR score as opposed to only looking at the LSTM accuracy/loss which can be misleading, and (d) building ensembles of multiple networks with different random initializations. In the following section we compare our best ensemble (selected on the validation set) to related work on the test sets of MPII-MD and M-VAD.

5.5.4.3 *Comparison to related work*

Experimental setup. In this section we perform the evaluation on the test set of the MPII-MD dataset (6,578 clips) and M-VAD dataset (4,951 clips). We use METEOR and CIDEr for automatic evaluation and we perform a human evaluation on a random subset of 1,300 video clips, see Section 5.5.3 for details. For M-VAD experiments we train our method on M-VAD and use the same LSTM architecture and parameters as for MPII-MD, but select the number of iterations on the M-VAD validation set.

Results on MPII-MD. Table 5.12 summarizes the results on the test set of MPII-MD. Here we additionally include the results from a nearest neighbor baseline, i.e. we retrieve the closest sentence from the training corpus using L1-normalized visual features and the intersection distance. Our SMT-Best approach clearly improves over the nearest neighbor baselines. With our Visual-Labels approach we significantly improve the performance, specifically by 1.44 METEOR points and 1.84 CIDEr points. Moreover, we improve over the recent approach of (Venugopalan *et al.*, 2015b), which also uses an LSTM to generate video descriptions. Exploring different strategies to

Approach	METEOR in %	CIDEr in %	Human evaluation: rank Correct.	Grammar	Relev.
NN baselines					
IDT	4.87	2.77	-	-	-
LSDA	4.45	2.84	-	-	-
PLACES	4.28	2.73	-	-	-
HYBRID	4.34	3.29	-	-	-
SMT-Best (ours)	5.59	8.14	2.11	2.39	2.08
S2VT (Venugopalan *et al.*, 2015b)	6.27	9.00	2.02	**1.67**	2.06
Visual-Labels (ours)	**7.03**	**9.98**	**1.87**	1.94	**1.86**
NN METEOR upperbound	19.43	-	-	-	-

Table 5.12: Comparison of our proposed methods to prior work on MPII-MD test set. Human eval ranked 1 to 3, lower is better. For discussion see Section 5.5.4.3. Bold values indicate the best performing variant per measure/column.

Approach	METEOR in %	CIDEr in %
Temporal attention (Yao *et al.*, 2015)	4.33	5.55
S2VT (Venugopalan *et al.*, 2015b)	5.62	7.22
Visual-Labels (ours)	**6.36**	**7.48**

Table 5.13: Comparison of our proposed methods to prior work on M-VAD test set. Human eval ranked 1 to 3, lower is better. For discussion see Section 5.5.4.3. Bold values indicate the best performing variant per measure/column.

label selection and classifier training, as well as various LSTM configurations allows to obtain better result than prior work on the MPII-MD dataset. Human evaluation mainly agrees with the automatic measure. Visual-Labels outperforms both other methods in terms of Correctness and Relevance, however it loses to S2VT in terms of Grammar. This is due to the fact that S2VT produces overall shorter (7.4 versus 8.7 words per sentence) and simpler sentences, while our system generates longer sentences and therefore has higher chances to make mistakes. We also propose a retrieval upperbound. For every test sentence we retrieve the closest training sentence according to the METEOR score. The rather low METEOR score of 19.43 reflects the difficulty of the dataset. We show some qualitative results in Figure 5.7.

Results on M-VAD. Table 5.13 shows the results on the test set of M-VAD dataset. Our Visual-Labels method outperforms S2VT (Venugopalan *et al.*, 2015b) and Temporal attention (Yao *et al.*, 2015) in METEOR and CIDEr score. As we see, the results agree with Table 5.12, but are consistently lower, suggesting that M-VAD is more challenging than MPII-MD. We attribute this to a more precise manual alignment of the MPII-MD dataset.

	Approach	Sentence
	SMT-Best (ours)	Someone is a man, someone is a man.
	S2VT	Someone looks at him, someone turns to someone.
	Visual-Labels (ours)	Someone is standing in the crowd, a little man with a little smile.
	Reference	Someone, back in elf guise, is trying to calm the kids.
	SMT-Best (ours)	The car is a water of the water.
	S2VT	On the door, opens the door opens.
	Visual-Labels (ours)	The fellowship are in the courtyard.
	Reference	They cross the quadrangle below and run along the cloister.
	SMT-Best (ours)	Someone is down the door, someone is a back of the door, and someone is a door.
	S2VT	Someone shakes his head and looks at someone.
	Visual-Labels (ours)	Someone takes a drink and pours it into the water.
	Reference	Someone grabs a vodka bottle standing open on the counter and liberally pours some on the hand.

Figure 5.7: Qualitative comparison of our proposed methods to prior work: S2VT (Venugopalan *et al.*, 2015b). Examples from the test set of MPII-MD. Visual-Labels identifies activities, objects, and places better than the other two methods. See Section 5.5.4.3.

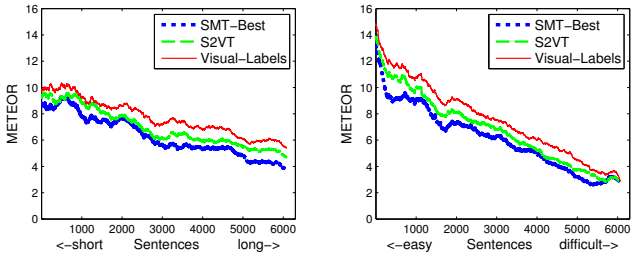
5.5.5 Movie description analysis

Despite the recent advances in the video description task, the performance on the movie description datasets (MPII-MD and M-VAD) remains rather low. In this section we want to look closer at three methods, SMT-Best, S2VT and Visual-Labels, in order to understand where these methods succeed and where they fail. In the following we evaluate all three methods on the MPII-MD test set.

5.5.5.1 *Difficulty versus performance*

As the first study we suggest to sort the test reference sentences by difficulty, where difficulty is defined in multiple ways.

Some of the intuitive sentence difficulty measures are its length and average frequency of its words. When sorting the data by difficulty (increasing sentence length or decreasing average word frequency), we find that all three methods have the same tendency to obtain lower METEOR score as the difficulty increases. Figure 5.8(a) shows the performance of compared methods w.r.t. the sentence length. For the word frequency the correlation is even stronger, see Figure 5.8(b). Visual-Labels consistently outperforms the other two methods, most notable as the difficulty increases.



(a) Sentence length (b) Word frequency

Figure 5.8: Y-axis: METEOR score per sentence. X-axis: MPII-MD test sentences 1 to 6,578 sorted by (a) length (increasing); (b) word frequency (decreasing). Shown values are smoothed with a mean filter of size 500. For discussion see Section 5.5.5.1.

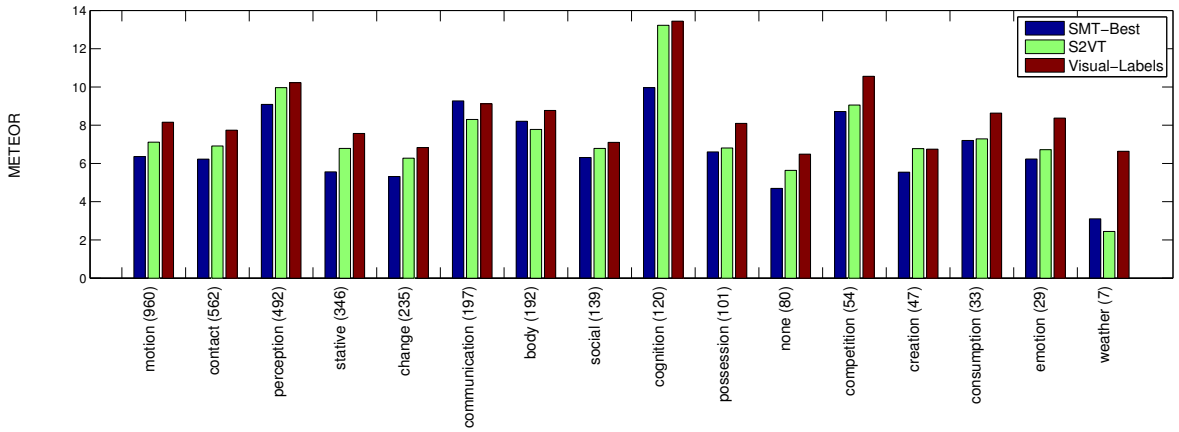


Figure 5.9: Average METEOR score for WordNet verb Topics. Selected sentences with single verb, number of sentences in brackets. For discussion see Section 5.5.5.2.

5.5.5.2 *Semantic analysis*

WordNet Verb Topics. Next we analyze the test reference sentences w.r.t. verb semantics. We rely on WordNet Topics (high level entries in the WordNet ontology), e.g. “motion”, “perception”, defined for most synsets in WordNet (Fellbaum, 1998). Sense information comes from our automatic semantic parser, thus it might be noisy. We showcase the 3 most frequent verbs for each Topic in Table 5.14. We select sentences with a single verb, group them according to the verb Topic and compute an average METEOR score for each Topic, see Figure 5.9. We find that Visual-Labels is best for all Topics except “communication", where SMT-Best wins. The most frequent verbs there are “look up” and “nod”, which are also frequent in the dataset and in the sentences produced by SMT-Best. The best performing Topic, “cognition”, is highly biased to “look at” verb. The most frequent Topics, “motion” and “contact”, which are also visual (e.g. “turn”, “walk”, “sit”), are nevertheless quite challenging,

Topic	Entropy	Top-1	Top-2	Top-3
motion	7.05	turn	walk	shake
contact	7.10	open	sit	stand
perception	4.83	look	stare	see
stative	4.84	be	follow	stop
change	6.92	reveal	start	emerge
communication	6.73	look up	nod	face
body	5.04	smile	wear	dress
social	6.11	watch	join	do
cognition	5.21	look at	see	read
possession	5.29	give	take	have
none	5.04	throw	hold	fly
creation	5.69	hit	make	do
competition	5.19	drive	walk over	point
consumption	4.52	use	drink	eat
emotion	6.19	draw	startle	feel
weather	3.93	shine	blaze	light up

Table 5.14: Entropy and top 3 frequent verbs of each WordNet topic. For discussion see Section 5.5.5.2.

which we attribute to their high diversity (see their entropy w.r.t. different verbs and their frequencies in Table 5.14). Topics with more abstract verbs (e.g. “be”, “have”, “start”) get lower scores.

Top 100 best and worst sentences. We look at 100 test reference sentences, where Visual-Labels obtains highest and lowest METEOR scores. Out of 100 best sentences 44 contain the verb “look” (including phrases such as “look at”). The other frequent verbs are “walk”, “turn", “smile”, “nod”, “shake”, i.e. mainly visual verbs. Overall the sentences are simple. Among the worst 100 sentences we observe more diversity: 12 contain no verb, 10 mention unusual words (specific to the movie), 24 have no subject, 29 have a non-human subject. This leads to a lower performance, in particular, as most training sentences contain “Someone” as subject and generated sentences are biased towards it.

Summary. a) The test reference sentences that mention verbs like “look” get higher scores due to their high frequency in the dataset. b) The sentences with more “visual” verbs tend to get higher scores. c) The sentences without verbs (e.g. describing a scene), without subjects or with non-human subjects get lower scores, which can be explained by dataset biases.

5.6 THE LARGE SCALE MOVIE DESCRIPTION CHALLENGE

The Large Scale Movie Description Challenge (LSMDC) was held twice, first in conjunction with ICCV 2015 (LSMDC 15) and then at ECCV 2016 (LSMDC 16). For the automatic evaluation we set up an evaluation server[3]. During the first phase of the challenge the participants could evaluate the outputs of their system on the public test set. In the second phase of the challenge the participants were provided

Approach	BLEU 1	BLEU 2	BLEU 3	BLEU 4	METEOR	ROUGE	CIDEr	SPICE
Submissions to LSMDC 15								
Visual-Labels (ours)	16.1	5.2	**2.1**	**0.9**	**7.1**	**16.4**	**11.2**	13.2
S2VT (Venugopalan *et al.*, 2015a)	**17.4**	**5.3**	1.8	0.7	7.0	16.1	9.1	11.4
Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015)	11.0	3.4	1.3	0.6	6.1	15.6	9.0	13.4
Temporal Attention (Yao *et al.*, 2015)	5.6	1.5	0.6	0.3	5.2	13.4	6.2	**14.3**
Submissions to LSMDC 16								
Temporal Tessellation (Kaufman *et al.*, 2016)	14.5	4.1	1.4	0.6	5.8	13.4	10.1	7.7
Aalto University (Shetty and Laaksonen, 2016)	6.9	1.6	0.5	0.2	3.4	7.0	3.5	2.6
Seoul NU	9.2	2.9	1.0	0.4	4.0	9.6	7.6	4.8
SNUVL (Yu *et al.*, 2017b)	15.6	4.4	1.4	0.4	7.1	14.7	7.0	11.5
IIT Kanpur	11.8	3.6	1.3	0.5	7.4	14.2	4.7	7.2
VD-ivt (BUPT CIST AI lab)	15.9	4.3	1.0	0.3	**8.0**	15.0	4.8	10.6

Table 5.15: Automatic evaluation on the blind test set of the LSMDC, in %. For discussion see Section 5.6.2. Bold indicates the best performing approach per measure/column for LSMDC 15, and LSMDC 16, if it improved over LSMDC 15.

with the videos from the blind test set (without textual descriptions). These were used for the final evaluation. To measure performance of the competing approaches we performed both automatic and human evaluation. The submission format was similar to the MS COCO Challenge (Chen *et al.*, 2015) and we also used the identical automatic evaluation protocol. The challenge winner was determined based on the human evaluation. In the following we review the participants and their results for both LSMDC 15 and LSMDC 16. As they share the same public and blind test sets, as described in Section 5.3.3, we can also compare the submissions to both challenges with each other.

5.6.1 LSMDC participants

We received 4 submissions to LSMDC 15, including our Visual-Labels approach. The other submissions are S2VT (Venugopalan *et al.*, 2015a), Temporal Attention (Yao *et al.*, 2015) and Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015). For LSMDC 16 we received 6 new submissions. As the blind test set is not changed between LSMDC 2015 to LSMDC 2016, we look at all the submitted results jointly. In the following we summarize the submissions based on the (sometimes very limited) information provided by the authors.

5.6.1.1 *LSMDC 15 submissions*

S2VT (Venugopalan *et al.*, 2015a). Venugopalan *et al.* (2015a) propose S2VT, an encoder-decoder framework, where a single LSTM encodes the input video, frame by frame, and decodes it into a sentence. We note that the results to LSMDC were obtained with a different set of hyper-parameters then the results discussed in the previous section. Specifically, S2VT was optimized w.r.t. METEOR on the validation set, which resulted in significantly longer but also nosier sentences.

Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015). Shetty and Laaksonen (2015) evaluate diverse visual features as input for an LSTM generation framework. Specifically they use dense trajectory features (Wang *et al.*, 2013a) extracted for the entire clip and VGG (Simonyan and Zisserman, 2015) and GoogleNet (Szegedy *et al.*, 2015) CNN features extracted at the center frame of each clip. They find that training 80 concept classifiers on MS COCO with the CNN features, combined with dense trajectories provides the best input for the LSTM.

Temporal Attention (Yao *et al.*, 2015). Yao *et al.* (2015) propose a soft-attention model based on (Xu *et al.*, 2015a) which selects the most relevant temporal segments in a video, incorporates 3-D CNN and generates a sentence using an LSTM.

5.6.1.2 *LSMDC 16 submissions*

Temporal Tessellation (Kaufman *et al.*, 2016). This submission retrieves a nearest neighbor from the training set, learning a unified space using Canonical Correlation Analysis (CCA) over textual and visual features. For the textual representation it relies on the Word2Vec representation using a Fisher Vector encoding with a Hybrid Gaussian-Laplacian Mixture Model (Klein *et al.*, 2015) and for the visual representation it uses RNN Fisher Vector (Lev *et al.*, 2015), encoding video frames with the 19-layer VGG.

Aalto University (Shetty and Laaksonen, 2016). Shetty and Laaksonen (2016) rely on an ensemble of four models which were trained on the MSR-VTT dataset (Xu *et al.*, 2016) without additional training on the LSMDC dataset. The four models were trained with different combinations of key-frame based GoogleLeNet features and segment based dense trajectory and C3D features. A separately trained evaluator network was used to predict the result of the ensemble.

Seoul NU. This work relies on temporal and attribute attention.

SNUVL (Yu *et al.*, 2017b). Yu *et al.* (2017b) first learn a set of semantic attribute classifiers. To generate a description for a video clip, they rely on attention over semantic attributes.

Approach	Avg. sent. length	Vocabulary size	% Unique sentences	% Novel sentences
Submissions to LSMDC 15				
Visual-Labels (ours)	7.47	525	45.11	66.76
S2VT (Venugopalan *et al.*, 2015a)	8.77	663	30.17	72.10
Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015)	5.16	401	9.09	30.81
Temporal Attention (Yao *et al.*, 2015)	3.63	117	1.39	6.48
Submissions to LSMDC 16				
Temporal Tessellation (Kaufman *et al.*, 2016)	9.34	5,530	58.35	0.00
Aalto University (Shetty and Laaksonen, 2016)	6.83	651	24.39	94.09
Seoul NU	6.16	459	24.26	52.78
SNUVL (Yu *et al.*, 2017b)	8.53	756	41.54	76.03
IIT Kanpur	16.2	1,172	39.37	100.00
VD-ivt (BUPT CIST AI lab)	8.00	7	0.01	100.00
Reference	8.75	6,820	97.19	92.63

Table 5.16: Description statistics for different methods and reference sentences on the blind test set of the LSMDC. For discussion see Section 5.6.2.

IIT Kanpur. This submission uses an encoder-decoder framework with 2 LSTMs, one LSTM used to encode the frame sequence of the video and another to decode it into a sentence.

VD-ivt (BUPT CIST AI lab). According to the authors, their VD-ivt model consists of three parallel channels: a basic video description channel, a sentence to sentence channel for language learning, and a channel to fuse visual and textual information.

5.6.2 LSMDC quantitative results

We first discuss the submissions w.r.t. to automatic measures and then discuss the human evaluations, which determined the winner for the challenges.

5.6.2.1 *Automatic evaluation*

We first look at the results of the automatic evaluation on the blind test set of LSMDC in Table 5.15. In the first edition of the challenge, LSMDC 15, our Visual-Labels approach obtains highest scores in all evaluation measures except BLEU-1,-2, where S2VT wins. One reason for lower scores for Frame-Video-Concept Fusion and Temporal Attention appears to be the generated sentence length, which is much smaller compared to the reference sentences, as we discuss below (see also

Approach	Correctness	Grammar	Relevance	Helpful for blind
Visual-Labels (ours)	3.32	3.37	3.32	**3.26**
S2VT (Venugopalan *et al.*, 2015a)	3.55	3.09	3.53	3.42
Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015)	**3.10**	**2.70**	**3.29**	3.29
Temporal Attention (Yao *et al.*, 2015)	3.14	2.71	3.31	3.36
Reference	1.88	3.13	1.56	1.57

Table 5.17: Human evaluation on the blind test set of the LSMDC 2015. Human eval ranked 1 to 5, lower is better. For discussion see Section 5.6.2. Bold indicates the best performing approach per measure / column.

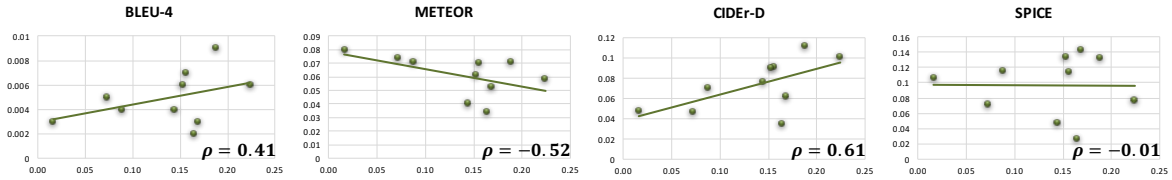


Figure 5.10: LSDMC 16: We plot the correlation between human evaluation score (x-axis) and 4 automatic measures (y-axis).

Table 5.16). When extended to LSMDC 16 submissions, we observe that most approaches perform below S2VT / Visual-Labels, except for VD-ivt, which achieves METEOR 8.0. Surprisingly, but confirmed with the authors, VD-ivt predicts only a single sentence "*Someone is in the front of the room.*", which seems to be optimized w.r.t. the METEOR score, while e.g. CIDEr score shows that this sentence is not good for most video clips. While most approaches are generating novel descriptions, Temporal Tessellation is the only retrieval-based approached among the submissions. It takes a second place w.r.t. the CIDEr score, while not achieving particularly high scores in other measures.

We closer analyze the outputs of the compared approaches in Table 5.16, providing detailed statistics over the generated descriptions. Among the LSMDC 15 submissions, with respect to the sentence length, Visual-Labels and S2VT demonstrate similar properties to the reference descriptions, while the approaches Frame-Video-Concept Fusion and Temporal Attention generate much shorter sentences (5.16 and 3.63 words on average vs. 8.74 of the references). In terms of vocabulary size all approaches fall far below the reference descriptions. This large gap indicates a problem in that all the compared approaches focus on a rather small set of visual and language concepts, ignoring a long tail in the distribution. The number of unique sentences confirms the previous finding, showing slightly higher numbers for Visual-Labels and S2VT, while the other two tend to frequently generate the same

Approach	better or equal than reference
Submissions to LSMDC 15	
Visual-Labels (ours)	18.8
S2VT (Venugopalan *et al.*, 2015a)	15.6
Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015)	15.2
Temporal Attention (Yao *et al.*, 2015)	16.8
Submissions to LSMDC 16	
Temporal Tessellation (Kaufman *et al.*, 2016)	**22.4**
Aalto University (Shetty and Laaksonen, 2016)	16.4
Seoul NU	14.4
SNUVL (Yu *et al.*, 2017b)	8.8
IIT Kanpur	7.2
VD-ivt (BUPT CIST AI lab)	1.6

Table 5.18: LSMDC 16. Human evaluation. Ratio of sentences which are judged better or equal compared to the reference description, with at least two out of three judges agreeing (in %). For discussion see Section 5.6.2. Bold indicates the best performing approach in the table.

description for different clips. Finally, the percentage of novel sentences (not present among the training descriptions) highlights another aspect, namely the amount of novel vs. retrieved descriptions. As we see, all the methods "retrieve" some amount of descriptions from training data, while the approach Temporal Attention produces only 7.36% novel sentences. Looking at the LSMDC 16 submissions, we, not surprisingly, see that Temporal Tessellation retrieval approach achieves highest diversity among all approaches. Most other submissions have similar statistics to LSMDC 15 submissions. Interestingly, Shetty and Laaksonen (2016) generate many novel sentences, as they are not trained on LSMDC, but on the MSR-VTT dataset. Two outliers are IIT Kanpur, which generates very long and noisy descriptions, and VD-ivt, which, as mentioned above, generates the same sentence for all video clips.

5.6.2.2 *Human evaluation*

We performed separate human evaluations for LSMDC 15 and LSMDC 16.

LSMDC 15. The results of the human evaluation are shown in Table 5.17. The human evaluation was performed over 1,200 randomly selected clips from the blind test set of LSMDC. We follow the evaluation protocol defined in Section 5.5.3.2. As known from literature (Chen *et al.*, 2015; Elliott and Keller, 2013; Vedantam *et al.*, 2015), automatic evaluation measures do not always agree with the human evaluation. Here we see that human judges prefer the descriptions from Frame-Video-Concept Fusion approach in terms of correctness, grammar and relevance. In

	Approach	Sentence
	Visual-Labels (ours)	Someone lies on the bed.
	S2VT	Someone lies asleep on his bed.
	Frame-Video-Concept Fusion	Someone lies on the bed.
	Temporal Attention	Someone lies in bed.
	Reference	Someone lies on her side facing her new friend.
	Visual-Labels (ours)	Someone sits down.
	S2VT	Someone sits on the couch and looks at the tv.
	Frame-Video-Concept Fusion	Someone sits at the table.
	Temporal Attention	Someone looks at someone.
	Reference	Someone takes a seat and someone moves to the stove.
	Visual-Labels (ours)	Someone walks to the front of the house.
	S2VT	Someone looks at the house.
	Frame-Video-Concept Fusion	Someone walks up to the house.
	Temporal Attention	Someone looks at someone.
	Reference	Someone sets down his young daughter then moves to a small wooden table.
	Visual-Labels (ours)	Someone turns to someone.
	S2VT	Someone looks at someone.
	Frame-Video-Concept Fusion	Someone turns to someone.
	Temporal Attention	Someone stands alone.
	Reference	Someone dashes for the staircase.
	Visual-Labels (ours)	Someone takes a deep breath and takes a deep breath.
	S2VT	Someone looks at someone and looks at him.
	Frame-Video-Concept Fusion	Someone looks up at the ceiling.
	Temporal Attention	Someone stares at someone.
	Reference	Someone digs out her phone again, eyes the display, and answers the call.

Figure 5.11: Qualitative comparison of our approach Visual-Labels, S2VT (Venugopalan *et al.*, 2015a), Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015) and Temporal Attention (Yao *et al.*, 2015) on the blind test set of the LSMDC. Discussion see Section 5.6.3.

our alternative evaluation, in terms of being helpful for the blind, Visual-Labels wins. Possible explanation for it is that in this evaluation criteria human judges penalized less the errors in the descriptions but rather looked at their overall informativeness. In general, the gap between different approaches is not large. Based on the human evaluation the winner of the LSMDC 15 challenge is Frame-Video-Concept Fusion approach of Shetty and Laaksonen (2015).

LSMDC 16. For the LSMDC 16 the evaluation protocol is different from the one above. As we have to compare more approaches the ranking becomes unfeasible. Additionally we would like to capture the human agreement in this evaluation. This leads us to the following evaluation protocol which is inspired by the human evaluation metric “M1" in the MS COCO Challenge (Chen *et al.*, 2015). The humans are provided with randomized pairs (reference, generated sentence) from each system and asked to decide in terms of being helpful for the blind person a) if sentence 1 is better b) both are similar c) sentence 2 is better. Each pair is judged

	Approach	Sentence
	Visual-Labels (ours)	Someone takes a seat on the table and takes a seat on his desk.
	S2VT	Someone looks at someone and smiles.
	Frame-Video-Concept Fusion	Someone looks at someone.
	Temporal Attention	Someone gets up.
	Temporal Tessellation	Farther along, the mustached stranger sits on a bench.
	Reference	Later, someone sits with someone and someone.
	Visual-Labels (ours)	Someone gets out of the car and walks off.
	S2VT	Someone walks up to the front of the house.
	Frame-Video-Concept Fusion	Someone walks up to the front door.
	Temporal Attention	Someone gets out of the car.
	Temporal Tessellation	He sees a seated man on the TV gesturing.
	Reference	Now someone steps out of the carriage with his new employers.
	Visual-Labels (ours)	Someone walks up to the street, and someone is walking to the other side of.
	S2VT	Someone walks over to the table and looks at the other side of the house.
	Frame-Video-Concept Fusion	Someone walks away.
	Temporal Attention	Someone gets out of the car.
	Temporal Tessellation	Later smiling, the two walk hand in hand down a busy sidewalk noticing every hat-wearing man they pass.
	Reference	The trio starts across a bustling courtyard.
	Visual-Labels (ours)	Someone sips his drink.
	S2VT	Someone sits at the table and looks at someone.
	Frame-Video-Concept Fusion	Someone sits up.
	Temporal Attention	Someone looks at someone.
	Temporal Tessellation	Someone sits at a table sipping a drink.
	Reference	As the men drink red wine, someone and someone watch someone take a sip.
	Visual-Labels (ours)	Someone takes a bite.
	S2VT	Someone sits at the table.
	Frame-Video-Concept Fusion	Someone looks at someone.
	Temporal Attention	Someone looks at someone.
	Temporal Tessellation	Later at the dinner table.
	Reference	Someone tops off someone's glass.

Figure 5.12: Qualitative comparison of our approach Visual-Labels, S2VT (Venugopalan *et al.*, 2015a), Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015), Temporal Attention (Yao *et al.*, 2015), and Temporal Tessellation (Kaufman *et al.*, 2016) on 5 consecutive clips from the blind test set of the LSMDC. Discussion see Section 5.6.3.

by 3 humans. For an approach to get a point at least 2 out of 3 humans should agree that a generated sentence is better or equal to a reference. The results of the human evaluation on 250 randomly selected sentence pairs are presented in Table 5.18. Temporal Tessellation (Kaufman *et al.*, 2016) is ranked best by the human judges and thus it wins the LSMDC 16 challenge. Visual-Labels gets the second place, next are Temporal Attention and Aalto University. The VD-ivt submission with identical descriptions is ranked worst. Additionally we measure the correlation between the automatic and human evaluation in Figure 5.10. We compare BLEU@4, METEOR, CIDEr and SPICE and find that CIDEr score provides the highest and reasonable (0.61) correlation with human judgments. SPICE shows no correlation, METEOR demonstrates negative correlation. We attribute this to the fact that the approaches generate very different types of descriptions (long/short, simple/retrieved from the training data, etc.) as discussed above and that we only have a single reference to compute these metrics. While we believe that these metrics can still provide reasonable scores for similar models, comparing very diverse methods and results, requires human evaluation. However, also for human evaluation, further studies are needed in the future, to determine what are the best evaluation protocols.

5.6.3 LSMDC qualitative results

Figure 5.11 shows qualitative results from the competing approaches submitted to LSMDC 15. The first two examples are success cases, where most of the approaches are able to describe the video correctly. The third example is an interesting case where visually relevant descriptions, provided by most approaches, do not match the reference description, which focuses on an action happening in the background of the scene ("Someone sets down his young daughter then moves to a small wooden table."). The last two rows contain partial and complete failures. In one all approaches fail to recognize the person running away, only capturing the "turning" action which indeed happened before running. In the other one, all approaches fail to recognize that the woman interacts with the small object (phone).

Figure 5.12 compares all LSMDC 15 approaches with the LSMDC 16 winner, Temporal Tessellation (Kaufman *et al.*, 2016), on a sequence of 5 consecutive clips. We can make the following observations from these examples. Although, Temporal Tessellation is a retrieval-based approach, it does very well in many cases, providing an added benefit of fluent and grammatically correct descriptions. One side-effect of retrieval is that when it fails, it produces a completely irrelevant description, e.g. the second example. Temporal Tessellation and Visual-Labels are able to capture important details, such as sipping a drink, which the other methods fail to recognize. Descriptions generated by Visual-Labels and S2VT tend to be longer and noisier than the ones by Frame-Video-Concept Fusion and Temporal Attention, while Temporal Attention tends to produce generally applicable sentences, e.g. "Someone looks at someone".

5.7 CONCLUSION

In this chapter we present the Large Scale Movie Description Challenge (LSMDC), a novel dataset of movies with aligned descriptions sourced from movie scripts and ADs (audio descriptions for the blind, also referred to as DVS). Altogether the dataset is based on 200 movies and has 128,118 sentences with aligned clips. We compare AD with previously used script data and find that AD tends to be more correct and relevant to the movie than script sentences.

Our approach, *Visual-Labels*, to automatic movie description trains visual classifiers and uses their scores as input to an LSTM. To handle the weak sentence annotations we rely on three ingredients. (1) We distinguish three semantic groups of labels (verbs, objects, and places). (2) We train them separately, removing the noisy negatives. (3) We select only the most reliable classifiers. For sentence generation we show the benefits of exploring different LSTM architectures and learning configurations.

To evaluate different approaches for movie description, we organized a challenge at ICCV 2015 (LSMDC 15) where we evaluated submissions using automatic and human evaluation criteria. We found that the approaches S2VT and our Visual-Labels generate longer and more diverse descriptions than the other submissions but are also more susceptible to content or grammatical errors. This consequently leads to worse human rankings with respect to correctness and grammar. In contrast, Frame-Video-Concept Fusion (Shetty and Laaksonen, 2015) wins the challenge by predicting medium length sentences with intermediate diversity, which gets rated best in human evaluation for correctness, grammar, and relevance. When ranking sentences with respect to the criteria "helpful for the blind", our Visual-Labels is well received by human judges, likely because it includes important aspects provided by the strong visual labels. Overall all approaches have problems with the challenging long-tail distributions of our data. Additional training data cannot fully ameliorate this problem because a new movie might always contain novel parts. We expect new techniques, including relying on different modalities, see e.g. (Hendricks *et al.*, 2016b), to overcome this challenge.

The second edition of our challenge (LSMDC 16) was held at ECCV 2016. This time we introduced a new human evaluation protocol to allow comparison of a large number of approaches. We found that the best approach in the new evaluation with the "helpful for the blind" criteria is a retrieval-based approach Temporal Tessellation (Kaufman *et al.*, 2016). Likely, human judges prefer the rich while also grammatically correct descriptions provided by this method. In the future work the movie description approaches should aim to achieve rich yet correct and fluent descriptions. Our evaluation server will continue to be available for automatic evaluation.

Our dataset has already been used beyond description, e.g. for learning video-sentence embeddings or for movie question answering. Beyond our current challenge on single sentences, the dataset opens new possibilities to understand stories and plots across multiple sentences in an open domain scenario on a large scale.

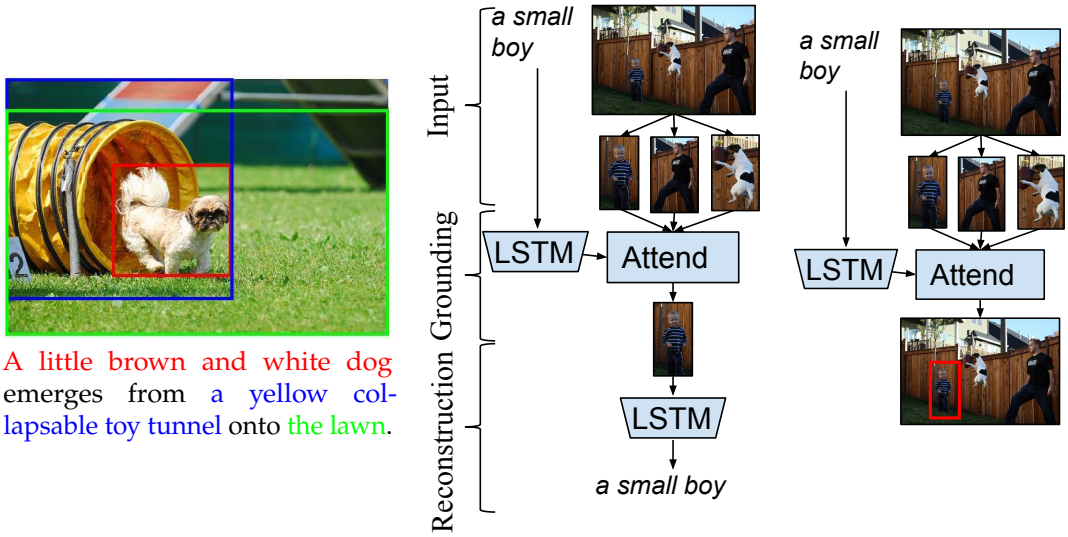
6 GROUNDING OF TEXTUAL PHRASES IN IMAGES BY RECONSTRUCTION

The previous chapters were concerned with the task of automatic video description. In this chapter we look at the task of visual grounding. Grounding (i.e. localizing) arbitrary, free-form textual phrases in visual content is a challenging problem with many applications for human-computer interaction and image-text reference resolution. Few datasets provide the ground truth spatial localization of phrases, thus it is desirable to learn from data with no or little supervision for grounding. We propose a novel approach which learns grounding by reconstructing a given phrase using an attention mechanism, which can be either latent or optimized directly. During training our approach encodes the phrase using a recurrent network language model and then learns to attend to the relevant image region in order to reconstruct the input phrase. At test time, the correct attention, i.e. the grounding, is evaluated. If grounding supervision is available it can be directly applied via a loss over the attention mechanism. We demonstrate the effectiveness of our approach on the Flickr 30k Entities (Plummer *et al.*, 2015) and ReferItGame (Kazemzadeh *et al.*, 2014) datasets with different levels of supervision, ranging from no supervision over partial supervision to full supervision. Our supervised variant improves by a large margin over the state-of-the-art on both datasets.

In Chapter 7 we propose a modification of this approach, as we explore different ways of combining visual and language representations.

6.1 INTRODUCTION

Language grounding in visual data is an interesting problem studied both in the computer vision (Karpathy and Fei-Fei, 2015; Karpathy *et al.*, 2014a; Kong *et al.*, 2014; Plummer *et al.*, 2015; Hu *et al.*, 2016b) and natural language processing (Krishnamurthy and Kollar, 2013; Matuszek *et al.*, 2012) communities. Such grounding can be done on different levels of granularity: from coarse, e.g. associating a paragraph of text to a scene in a movie (Tapaswi *et al.*, 2015; Zhu *et al.*, 2015b), to fine, e.g. localizing a word or phrase in a given image (Plummer *et al.*, 2015; Hu *et al.*, 2016b). In this chapter we focus on the latter scenario. Many prior efforts in this area have focused on rather constrained settings with a small number of nouns to ground (Lin *et al.*, 2014a; Kong *et al.*, 2014). On the contrary, we want to tackle the problem of grounding arbitrary natural language phrases in images. Most parallel corpora of sentence/visual data do not provide localization annotations (e.g. bounding boxes) and the annotation process is costly. We propose an approach which can learn to localize phrases relying only on phrases associated with images without bounding



(a) Predicted grounding. (b) Training time. (c) Test time.

Figure 6.1: (a) Without bounding box annotations at training time our approach GroundeR can ground free-form natural language phrases in images. (b) During training our latent attention approach reconstructs phrases by learning to attend to the correct box. (c) At test time, the attention model infers the grounding for each phrase. For semi-supervised and fully supervised variants see Figure 6.2.

box annotations but which is also able to incorporate phrases with bounding box supervision when available (see Figure 6.1).

The main idea of our approach is shown in Figure 6.1(b,c). Let us first consider the scenario where no localization supervision is available. Given images paired with natural language phrases we want to localize these phrases with a bounding box in the image (Figure 6.1c). To do this we propose a model (Figure 6.1b) which learns to attend to a bounding box proposal and, based on the selected bounding box, reconstructs the phrase. As the second part of the model (Figure 6.1b, bottom) is able to predict the correct phrase only if the first part of the model attended correctly (Figure 6.1b, top), this can be learned without additional bounding box supervision. Our method is based on *Ground*ing with a *R*econstruction loss and hence named *GroundeR*. Additional supervision is integrated in our model by adding a loss function which directly penalizes incorrect attention before the reconstruction step. At test time we evaluate whether the model attends to the correct bounding box.

We propose a novel approach to grounding of textual phrases in images which can operate in all supervision modes: with no, a few, or all grounding annotations available. We evaluate our GroundeR approach on the Flickr 30k Entities (Plummer *et al.*, 2015) and ReferItGame (Kazemzadeh *et al.*, 2014) datasets and show that our unsupervised variant is better than prior work and our supervised approach significantly outperforms state-of-the-art on both datasets. Interestingly, our semi-supervised approach can effectively exploit small amounts of labeled data and surpasses the supervised variant by exploiting multiple losses.

6.2 RELATED WORK

Related work on language grounding in image and video data has been presented in Section 2.3 of the thesis. In the following we compare our problem setting and approach to relevant work on object co-localization (or co-detection). We then review how the attention mechanism is used in the recent work and finally, discuss other approaches that perform bi-directional mapping between two domains.

Object co-localization focuses on discovering and detecting an object in images or videos without any bounding box annotation, but only from image/video level labels (Blaschko *et al.*, 2010; Cinbis *et al.*, 2014; Joulin *et al.*, 2014; Kwak *et al.*, 2015; Song *et al.*, 2014; Tang *et al.*, 2014; Yu and Siskind, 2015b). These works are similar to ours with respect to the amount of supervision, but they focus on a few discrete classes, while our approach can handle arbitrary phrases and allows for localization of novel phrases. There are also works that propose to train detectors for a wide range of concepts using image-level annotated data from web image search (Chen and Gupta, 2015; Divvala *et al.*, 2014). These approaches are complementary to ours in the sense of obtaining large scale concept detectors with little supervision, however they do not tackle complex phrases e.g. "a blond boy on the left" which is the focus of our work.

Attention in vision tasks. Recently, different attention mechanisms have been applied to a range of computer vision tasks. The general idea is that given a visual input, e.g. set of features, at any given moment we might want to focus only on part of it, e.g. attend to a specific subset of features (Bahdanau *et al.*, 2015). Xu *et al.* (2015a) integrate spatial attention into their image captioning pipeline. They consider two variants: "soft" and "hard" attention, meaning that in the latter case the model is only allowed to pick a single location, while in the first one the attention "weights" can be distributed over multiple locations. Jin *et al.* (2015) adapt the soft-attention mechanism and attends to bounding box proposals, one word at a time, while generating an image captioning. Yao *et al.* (2015) rely on a similar mechanism to perform temporal attention for selecting frames in video description task. Yeung *et al.* (2015) use attention mechanism to densely label actions in a video sequence. Our approach relies on soft-attention mechanism, similar to the one of Xu *et al.* (2015a). We apply it to the language grounding task where attention helps us to select a bounding box proposal for a given phrase.

Bi-directional mapping. In our model, a phrase is first mapped to a image region through attention, and then the image region is mapped back to phrase during reconstruction. There is conceptual similarity between previous work and ours on the idea of bi-directional mapping from one domain to another. In autoencoders (Vincent *et al.*, 2008), input data is first mapped to a compressed vector during encoding, and then reconstructed during decoding. Chen and Zitnick (2015) use a bi-directional mapping from visual features to words and from words to visual features in a recurrent neural network model. The idea is to generate descriptions from visual features and then to reconstruct visual features given a description. Similar to Chen and Zitnick (2015), our model can also learn to associate input text with

visual features, but through attending to an image region rather than reconstructing directly from words. In the linguistic community, Ammar *et al.* (2014) proposed a CRF Autoencoder, which generates latent structures for the given language input and then reconstructs the input from these latent structures, with the application to e.g. part-of-speech tagging.

6.3 GROUNDER: GROUNDING BY RECONSTRUCTION

The goal of our approach is to ground natural language phrases in images. More specifically, to ground a phrase p in an image I means to find a region r_j in the image which corresponds to this phrase. r_j can be any subset of I, e.g. a segment or a bounding box. The core insight of our method is that there is a bi-directional correspondence between an image region and the phrase describing it. As a correct grounding of a textual phrase should result in an image region which a human would describe using this phrase, i.e. it is possible to reconstruct the phrase based on the grounded image region. Thus, the key idea of our approach is to learn to ground a phrase by reconstructing this phrase from an automatically localized region. Figure 6.1 gives an overview of our approach.

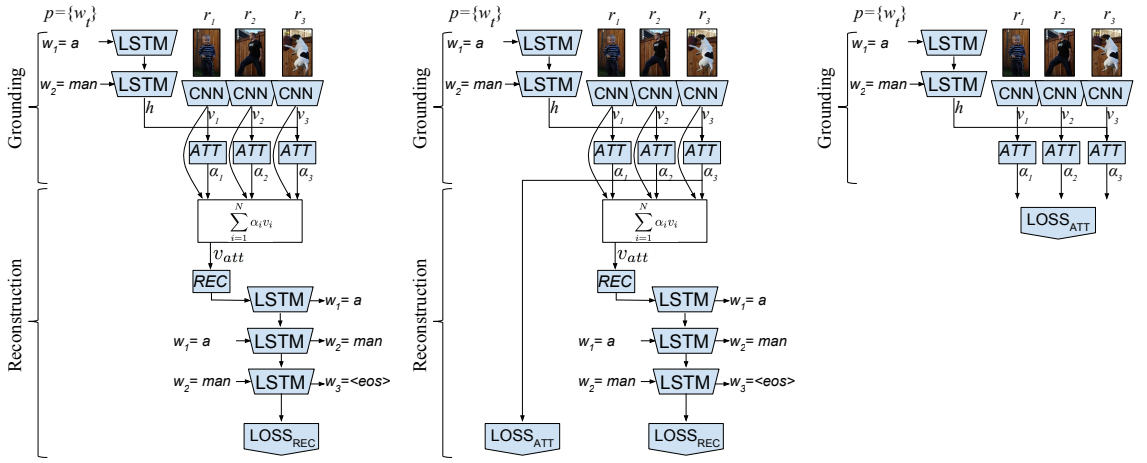
In this work, we utilize a set of automatically generated bounding box proposals $\{r_i\}_{i \in N}$ for the image I. Given a phrase p, during training our model works in two parts: the first part aims to attend to the most relevant region r_j (or potentially also multiple regions) based on the phrase p, and then the second part tries to reconstruct the same phrase p from region(s) r_j it attended to in the first phase. Therefore, by training to reconstruct the text phrase, the model learns to first ground the phrase in the image, and then generate the phrase from that region. Figure 6.2a visualizes the network structure. At test time, we remove the phrase reconstruction part, and use the first part for phrase grounding. The described pipeline can be extended to accommodate partial supervision, i.e. ground-truth phrase localization. For that we integrate an additional loss into the model, which directly optimizes for correct attention prediction, see Figure 6.2b. Finally, we can adapt our model to the fully supervised scenario by removing the reconstruction phase, see Figure 6.2c.

In the following we present the details of the two parts in our approach: learning to attend to the correct region for a given phrase and learning to reconstruct the phrase from the attended region. For simplicity, but without loss of generality, we will refer to r_j as a single bounding box.

6.3.1 Learning to ground

We frame the problem of grounding a phrase p in image I as selecting a bounding box r_j from a set of image region proposals $\{r_i\}_{i=1,\cdots,N}$. To select the correct bounding box, we define an attention function f_{ATT} and select the box j which receives the maximum attention:

$$j = \arg\max_i f_{ATT}(p, r_i) \tag{6.1}$$



(a) Unsupervised (b) Semi-supervised (c) Fully supervised

Figure 6.2: Our model learns grounding of textual phrases in images with (a) no, (b) little (c) or full supervision of localization, through a grounding part and a reconstruction part. During training, the model distributes its attention to a single or several boxes, and learns to reconstruct the input phrase based on the boxes it attends to. At test time, only the grounding part is used.

In the following we describe the details of how we model the attention in f_{ATT}. The attention mechanism used in our model is inspired by and similar to the soft attention formulations of Jin *et al.* (2015) and Xu *et al.* (2015a). However, our inputs to the attention predictor are not single words but rather multi-word phrases, and consequently we also do not have a "doubly stochastic attention" which is used in Xu *et al.* (2015a) to normalize the attention across words.

The phrases that we are dealing with might be very complex thus we require a good language model to represent them. We choose a Long Short-Term Memory network (LSTM) (Hochreiter and Schmidhuber, 1997) as our phrase encoder, as it has been shown effective in various language modeling tasks, e.g. translation (Sutskever *et al.*, 2014). We encode our query phrase word by word with an LSTM and obtain a representation of the phrase using the hidden state h at the final time step as:

$$h = f_{LSTM}(p) \tag{6.2}$$

Each word w_t in the phrase p is first encoded with a one-hot-vector. Then it is embedded in the lower dimensional space and given to LSTM.

Next, each bounding box r_i is encoded using a convolutional neural network (CNN) to compute the visual feature vector v_i:

$$v_i = f_{CNN}(r_i) \tag{6.3}$$

Based on the encoded phrase and feature representation of each proposal, we use a two layer perceptron to compute the attention on the proposal r_i:

$$\bar{\alpha}_i = f_{ATT}(p, r_i) = W_2\phi(W_h h + W_v v_i + b_1) + b_2 \tag{6.4}$$

where ϕ is the rectified linear unit (ReLU): $\phi(x) = max(0, x)$. We found that this architecture performs better than e.g. a single layer perceptron with a hyperbolic tangent nonlinearity used in Bahdanau *et al.* (2015).

We get normalized attention weights α_i by using softmax, which can be interpreted as probability of region r_i being the correct region $r_{\hat{j}}$:

$$\alpha_i = P(i = \hat{j}|\bar{\alpha}) = \frac{\exp(\bar{\alpha}_i)}{\sum_{k=1}^{N} \exp(\bar{\alpha}_k)} \tag{6.5}$$

If at training time we have ground truth information, i.e. that $r_{\hat{j}}$ is the correct proposal box, then we can compute the loss L_{att} based on our prediction as:

$$L_{att} = -\frac{1}{B} \sum_{b=1}^{B} \log(P(\hat{j}|\bar{\alpha})), \tag{6.6}$$

where B is the number of phrases per batch. This loss activates only if the training sample has the ground-truth attention value, otherwise, it is zero. If we do not have ground truth annotations then we have to define a loss function to learn the parameters of f_{ATT} in a weakly supervised manner. In the next section we describe how we define this loss by aiming to reconstruct the phrase based on the boxes that are attended to. At test time, we calculate the IOU (intersection over union) value between the selected box r_j and the ground truth box $r_{\hat{j}}$.

6.3.2 Learning to reconstruct

The key idea of our phrase reconstruction model is to learn to reconstruct the phrase only from the attended boxes. Given an attention distribution over the boxes, we compute a weighted sum over the visual features and the attention weights α_i:

$$v_{att} = \sum_{i=1}^{N} \alpha_i v_i, \tag{6.7}$$

which aggregates the visual features from the attended boxes. Then, the visual features v_{att} are further encoded into v'_{att} using a non-linear encoding layer:

$$v'_{att} = f_{REC}(v_{att}) = \phi(W_a v_{att} + b_a) \tag{6.8}$$

We reconstruct the input phrase based on this encoded visual feature v'_{att} over attended regions. During reconstruction, we use an image description LSTM that takes v'_{att} as input to generate a distribution over phrases p:

$$P(p|v'_{att}) = f_{LSTM}(v'_{att}) \tag{6.9}$$

where $P(p|v'_{att})$ is a distribution over the phrases conditioned on the input visual feature. Our approach for phrase generation is inspired by Donahue *et al.* (2015);

Vinyals *et al.* (2015) who have effectively used LSTM for generating image descriptions based on visual features. Given a visual feature, it learns to predict a word sequence $\{w_t\}$. At each time step t, the model predicts a distribution over the next word w_{t+1} conditioned on the input visual feature v'_{att} and all the previous words. We use a single LSTM layer and we feed the visual input only at the first time step. We use LSTM as our phrase encoder as well as decoder. Although one could potentially use other approaches to map phrases into a lower dimensional semantic space, it is not clear how one would do the reconstruction without the recurrent network, given that we have to train encoding and decoding end-to-end.

Importantly, the entire grounding+reconstruction model is trained as a single deep network through back-propagation by maximizing the likelihood of the ground truth phrase $\hat{p}$ generated during reconstruction, where we define the training loss for batch size B:

$$L_{rec} = -\frac{1}{B}\sum_{b=1}^{B}\log(P(\hat{p}|v'_{att})) \tag{6.10}$$

Finally, in the semi-supervised model we have both losses L_{att} and L_{rec}, which are combined as follows:

$$L = \lambda L_{att} + L_{rec} \tag{6.11}$$

where parameter λ regulates the importance of the attention loss.

6.4 EXPERIMENTS

We first discuss the experimental setup and design choices of our implementation and then present quantitative results on the test sets of Flickr 30k Entities (Tables 6.1,6.2) and ReferItGame (Table 6.3) datasets. We find our best results to outperform state-of-the-art on both datasets by a significant margin. Figures 6.3 and 6.4 show qualitatively how well we can ground phrases in images.

6.4.1 Experimental setup

We evaluate GroundeR on the datasets Flickr 30k Entities (Plummer *et al.*, 2015) and ReferItGame (Kazemzadeh *et al.*, 2014). Flickr 30k Entities contains over 275K bounding boxes from 31K images associated with natural language phrases. Some phrases in the dataset correspond to multiple boxes, e.g. "two men". For consistency with Plummer *et al.* (2015), in such cases we consider the union of the boxes as ground truth. We use 1,000 images for validation, 1,000 for testing and 29,783 for training. The ReferItGame dataset contains over 99K regions from 20K images. Regions are associated with natural language expressions, constructed to disambiguate the described objects. We use the bounding boxes provided by Hu *et al.* (2016b) and the same test split, namely 10K images for testing; the rest we split in 9K training and 1K validation images.

We obtain 100 bounding box proposals for each image using Selective Search (Uijlings *et al.*, 2013) for Flickr 30k Entities and Edge Boxes (Zitnick and Dollár, 2014) for ReferItGame dataset. For our semi-supervised and fully supervised models we obtain the ground-truth attention by selecting the proposal box which overlaps most with the ground-truth box, while the overlap IOU (intersection over union) is above 0.5. Thus, our fully supervised model is not trained with all available training phrase-box pairs, but only with those where such proposal boxes exist.

On the Flickr 30k Entities for the visual representation we rely on the VGG16 network (Simonyan and Zisserman, 2015) trained on ImageNet (Deng *et al.*, 2009). For each box we extract a 4,096 dimensional feature from the fully connected fc7 layer. We also consider a VGG16 network fine-tuned for object detection on PASCAL (Everingham *et al.*, 2010), trained using Fast R-CNN (Girshick, 2015a). In the following we refer to both features as VGG-CLS and VGG-DET, respectively. We do not fine-tune the VGG representation for our task to reduce computational and memory load, however, our model trivially allows back-propagation into the image representation which likely would lead to further improvements. For the ReferItGame dataset we use the VGG-CLS features and additional spatial features provided by Hu *et al.* (2016b). We concatenate both and refer to the obtained feature as VGG+SPAT. For the language encoding and decoding we rely on the LSTM variant implemented in Caffe (Jia *et al.*, 2014) which we initialize randomly and jointly train with the grounding task.

At test time we compute the accuracy as the ratio of phrases for which the attended box overlaps with the ground-truth box by more than 0.5 IOU.

6.4.2 Design choices and findings

In all experiments we use the Adam solver (Kingma and Ba, 2014), which adaptively changes the learning rate during training. We train our models for about 20/50 epochs for the Flickr 30k Entities/ReferItGame dataset, and pick the iteration on the validation set. Next, we report our results for optimizing hyperparmeters on the validation set of Flickr 30k Entities while using the VGG-CLS features.

Regularization. Applying L2 regularization to parameters (weight decay) is important for the best performance of our unsupervised model. By introducing the weight decay of 0.0005 we improve the accuracy from 20.33% to 22.96%. In contrast, when supervision is available, we introduce batch normalization (Ioffe and Szegedy, 2015) for the phrase encoding LSTM and visual feature, which leads to a performance improvement, in particular from 37.42% to 40.93% in the supervised scenario.

Layer initialization. We experiment with different ways to initialize the layer parameters. The configuration which works best for us is using uniform initialization for LSTM, MSRA (He *et al.*, 2015) for convolutional layers, and Xavier (Glorot and Bengio, 2010) for all other layers. Switching from Xavier to MSRA initialization for the convolutional layers improves the accuracy of the unsupervised model from 21.04% to 22.96%.

Approach	Accuracy		
	Other	VGG-CLS	VGG-DET
Unsupervised training			
Deep Fragments [6]	21.78	-	-
GroundeR	-	24.66	28.94
Supervised training			
CCA (Plummer *et al.*, 2015)	-	27.42	-
SCRC (Hu *et al.*, 2016b)	-	27.80	-
DSPE (Wang *et al.*, 2016a)	-	-	43.89
GroundeR	-	41.56	47.81
Semi-supervised training			
GroundeR 3.12% annot.	-	33.02	42.32
GroundeR 6.25% annot.	-	37.10	44.02
GroundeR 12.5% annot.	-	38.67	44.96
GroundeR 25.0% annot.	-	39.31	45.32
GroundeR 50.0% annot.	-	40.72	46.65
GroundeR 100.0% annot.	-	42.43	48.38
Proposal upperbound	77.90	77.90	77.90

Table 6.1: Phrase localization performance on Flickr 30k Entities with different levels of bounding box supervision, accuracy in %.

6.4.3 Experiments on the Flickr 30k Entities dataset

We report the performance of our approach with multiple levels of supervision in Table 6.1. In the last line of the table we report the proposal upper-bound accuracy, namely the presence of the correct box among the proposals (which overlaps with the ground-truth box with $IOU > 0.5$).

Unsupervised training. We start with the unsupervised scenario, i.e. no phrase localization ground-truth is used at training time. Our approach, which relies on VGG-CLS features, is able to achieve 24.66% accuracy. Note that the VGG network trained on ImageNet has not seen any bounding box annotations at training time. VGG-DET, which was fine-tuned for detection, performs better and achieves 28.94% accuracy. We can further improve this by taking a sentence constraint into account. Namely, it is unlikely that two different phrases from one sentence are grounded to the same box. Thus we post-process the attended boxes: we jointly process the phrases from one sentence and greedily select the highest scoring box for each phrase, while the same box cannot be selected twice. This allows us to reach the accuracy of 25.01% for VGG-CLS and 29.02% for VGG-DET. While we currently only use a sentence constraint as a simple post processing step at test time, it would be interesting to include a sentence level constraint during training as part of future work. We compare to the unsupervised Deep Fragments approach of Karpathy *et al.* (2014a). Note, that Karpathy *et al.* (2014a) do not report the grounding

performance and does not allow for direct comparison with our work. With our best case evaluation[1] of Deep Fragments (Karpathy *et al.*, 2014a), which also relies on detection boxes and features, we achieve an accuracy of 21.78%. Overall, the ranking objective in Karpathy *et al.* (2014a) can be seen complimentary to our reconstruction objective. It might be possible, as part of future work, to combine both objectives to learn even better models without grounding supervision.

Supervised training. Next we look at the fully supervised scenario. The accuracy achieved by Plummer *et al.* (2015) is 27.42%[2] and by SCRC (Hu *et al.*, 2016b) is 27.80%. Recent approach of Wang *et al.* (2016a) achieves 43.89% with VGG-DET features. Our approach, when using VGG-CLS features achieves an accuracy of 41.56%, significantly improving over prior works that use VGG-CLS. We further improve our result to impressive 47.81% when using VGG-DET features.

Semi-supervised training. Finally, we move to the semi-supervised scenario. The notation "x% annot." means that x% of the annotated data (where ground-truth attention is available) is used. As described in Section 6.3.2 we have a parameter λ which controls the weight of the attention loss L_{att} vs. the reconstruction loss L_{rec}. We estimate the value of λ on validation set and fix it for all iterations. We found that we need higher weight on L_{att} when little supervision is available. E.g. for 3.12% of supervision $\lambda = 200$ and for 12.5% supervision $\lambda = 50$. This is due to the fact that in these cases only 3.12% / 12.5% of labeled instances contribute to L_{att}, while all instances contribute to L_{rec}.

When integrating 3.12% of the available annotated data into the model we significantly improve the accuracy from 24.66% to 33.02% (VGG-CLS) and from 28.94% to 42.32% (VGG-DET). The accuracy further increases when providing more annotations, reaching 42.43% for VGG-CLS and 48.38% for VGG-DET when using all annotations. As ablation of our semi-supervised model we evaluated the supervised model while only using the respective x% of annotated data. We observed consistent improvement of our semi-supervised model over the supervised model. Intrestingly, when using all available supervision, L_{rec} still helps to improve performance over the supervised model (42.43% vs. 41.56%, 48.38% vs. 47.81%). Our intuition for this is that L_{att} only has a single correct bounding box (which overlaps most with the ground truth), while L_{rec} can also learn from overlapping boxes with high but not best overlap.

Results per phrase type. Flickr 30k Entities dataset provides a "type of phrase" annotation for each phrase, which we analyze in Table 6.2. Our unsupervised

[1] We train the Deep Fragments model (Karpathy *et al.*, 2014a) on the the Flickr 30k dataset and evaluate with the Flickr 30k Entities ground truth phrases and boxes. Our trained Deep Fragments model achieves 11.2%/16.5% recall@1 for image annotation/search compared to 10.3%/16.4% reported in Karpathy *et al.* (2014a). As there is a large number of dependency tree fragments per sentence (on average 9.5) which are matched to proposal boxes, rather than on average 3.0 noun phrases per sentence in Flickr 30k Entities, we make a best case study in favor of Karpathy *et al.* (2014a). For each ground-truth phrase we take the maximum overlapping dependency tree fragments (w.r.t. word overlap), compute the IOU between their matched boxes and the ground truth, and take the highest IOU.

[2] The number was provided by the authors of Plummer *et al.* (2015), while in Plummer *et al.* (2015) they report 25.30% for phrases automatically extracted with a parser.

Phrase type	people	clothing	body parts	animals	vehicles	instruments	scene	other	novel
Number of instances	5,656	2,306	523	518	400	162	1,619	3,374	2,214
Unsupervised training									
GroundeR (VGG-DET)	44.32	9.02	0.96	46.91	46.00	19.14	28.23	16.98	25.43
Supervised training									
CCA embedding Plummer *et al.* (2015)	29.58	24.20	10.52	33.40	34.75	35.80	20.20	20.75	n/a
GroundeR (VGG-CLS)	53.80	34.04	7.27	49.23	58.75	22.84	52.07	24.13	34.28
GroundeR (VGG-DET)	61.00	38.12	10.33	62.55	68.75	36.42	58.18	29.08	40.83
Semi-supervised training									
GroundeR (VGG-DET) 3.12% annot.	56.51	29.84	9.18	57.34	59.75	28.40	50.71	24.48	34.28
GroundeR (VGG-DET) 100.0% annot.	60.24	39.16	14.34	64.48	67.50	38.27	59.17	30.56	42.37
Proposal upperbound	85.93	66.70	41.30	84.94	89.00	70.99	91.17	69.29	79.90

Table 6.2: Detailed phrase localization, Flickr 30k Entities, accuracy in %.

approach does well on phrases like "people", "animals", "vehicles" and worse on "clothing" and "body parts". This could be due to confusion between people and their clothing or body parts. To address this, one could jointly model the phrases and add spatial relations between them in the model. Body parts are also the most challenging type to detect, with the proposal upper-bound of only 41.3%. The supervised model with VGG-CLS features outperforms Plummer *et al.* (2015) in all types except "body parts" and "instruments", while with VGG-DET it is better or similar in all types. Semi-supervised model brings further significant performance improvements, in particular for "body parts". In the last column we report the accuracy for novel phrases, i.e. the ones which did not appear in the training data. On these phrases our approach maintains high performance, although it is lower than the overall accuracy. This shows that learned language representation is effective and allows transfer to unseen phrases.

Summary Flickr 30k Entities. Our unsupervised approach performs similar (VGG-CLS) or better (VGG-DET) than the fully supervised methods of Plummer *et al.* (2015) and Hu *et al.* (2016b) (Table 6.1). Incorporating a small amount of supervision (e.g. 3.12% of annotated data) allows us to outperform Plummer *et al.* (2015) and Hu *et al.* (2016b) also when VGG-CLS features are used. Our best supervised model achieves 47.81%, surpassing all the previously reported results, including Wang *et al.* (2016a). Our semi-supervised model efficiently exploits the reconstruction loss L_{rec} which allows it to outperform the supervised model.

6.4.4 Experiments on the ReferItGame dataset

Table 6.3 summarizes results on the ReferItGame dataset. We compare our approach to the previously introduced fully supervised method SCRC (Hu *et al.*, 2016b), as well as provide reference numbers for two other baselines: LRCN Donahue *et al.* (2015) and CAFFE-7K (Guadarrama *et al.*, 2014) reported in Hu *et al.* (2016b). The LRCN baseline of Hu *et al.* (2016b) is using the image captioning model LRCN

Approach	Accuracy		
	Other	VGG	VGG+SPAT
Unsupervised training			
LRCN (Donahue *et al.*, 2015) (reported in Hu *et al.* (2016b))	8.59	-	-
CAFFE-7K (Guadarrama *et al.*, 2014) (reported in Hu *et al.* (2016b))	10.38	-	-
GroundeR	-	10.69	10.70
Supervised training			
SCRC (Hu *et al.*, 2016b)	-	-	17.93
GroundeR	-	23.44	26.93
Semi-supervised training			
GroundeR 3.12% annot.	-	13.70	15.03
GroundeR 6.25% annot.	-	16.19	19.53
GroundeR 12.5% annot.	-	19.02	21.65
GroundeR 25.0% annot.	-	21.43	24.55
GroundeR 50.0% annot.	-	22.67	25.51
GroundeR 100.0% annot.	-	24.18	28.51
Proposal upperbound	59.38	59.38	59.38

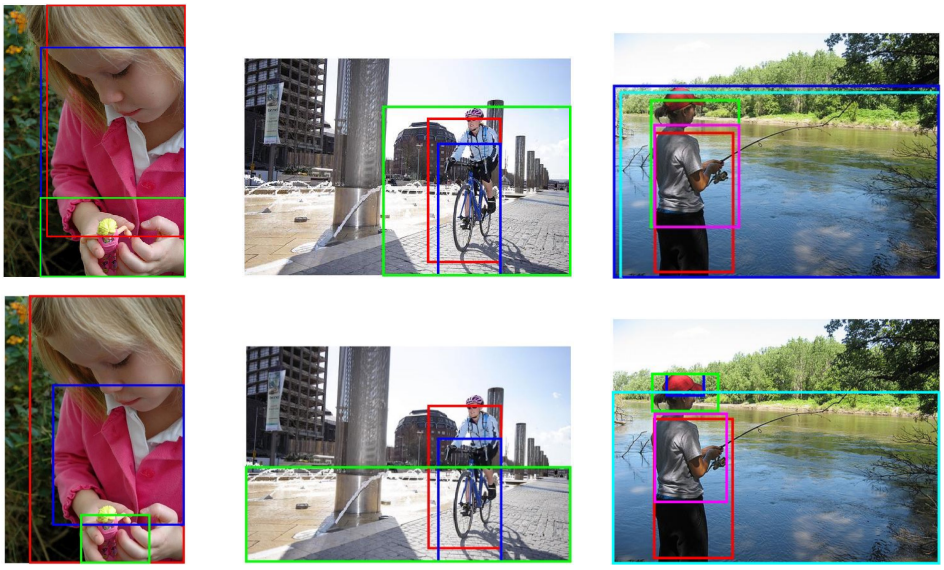
Table 6.3: Phrase localization performance on ReferItGame with different levels of bounding box supervision, accuracy in %.

(Donahue *et al.*, 2015) trained on MSCOCO (Lin *et al.*, 2014b) to score how likely the query phrase is to be generated for the proposal box. CAFFE-7K is a large scale object classifier trained on ImageNet (Deng *et al.*, 2009) to distinguish 7K classes. Guadarrama *et al.* (2014) predict a class for each proposal box and constructs a word bag with all the synonyms of the class-name based on WordNet (Fellbaum, 1998). The obtained word bag is then compared to the query phrase after both are projected to a joint vector space. Both approaches are unsupervised w.r.t. the phrase bounding box annotations. Table 6.3 reports the results of our approach with VGG, as well as VGG+SPAT features of Hu *et al.* (2016b).

Unsupervised training. In the unsupervised scenario our GroundeR performs competitive with the LRCN and CAFFE-7K baselines, achieving 10.7% accuracy. We note that in this case VGG and VGG+SPAT perform similarly.

Supervised training. In the supervised scenario we compare to the best prior work on this dataset, SCRC (Hu *et al.*, 2016b), which reaches 17.93% accuracy. Our supervised approach, which uses identical visual features, significantly improves this performance to 26.93%.

Semi-supervised training. Moving to the semi-supervised scenario again demonstrates performance improvements, similar to the ones observed on Flickr 30k Entities datset. Even the small amount of supervision (3.12%) significantly improves performance to 15.03% (VGG+SPAT), while with 100% of annotations we achieve 28.51%,



A little girl in a pink shirt is looking at a toy doll.

A woman is riding a bicycle on the pavement.

A girl with a red cap, hair tied up and a gray shirt is fishing in a calm lake.

Figure 6.3: Qualitative results on the test set of Flickr 30k Entities. Top : GroundeR (VGG-DET) unsupervised, bottom: GroundeR (VGG-DET) supervised.

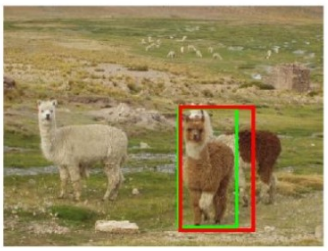
outperforming the supervised model.

Summary ReferItGame dataset. While the unsupervised model only slightly improves over prior work, the semi-supervised version can effectively learn from few labeled training instances, and with all supervision it achieves 28.51%, improving over Hu *et al.* (2016b) by a large margin of 10.6%. Overall the performance on ReferItGame dataset is significantly lower than on Flickr 30k Entities. We attribute this to two facts. First, the training set of ReferItGame is rather small compared to Flickr 30k (9k vs. 29k images). Second, the proposal upperbound on ReferItGame is significantly lower than on Flickr 30k Entities (59.38% vs 77.90%) due to the complex nature of the described objects and "stuff" image regions.

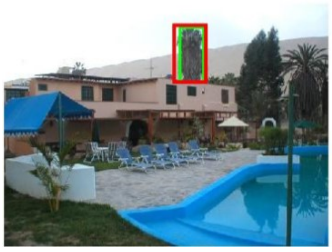
Qualitative results. We provide qualitative results on Flickr 30K Entities dataset in Figure 6.3. We compare our unsupervised and supervised approaches, both with VGG-DET features. The supervised approach visibly improves the localization quality over the unsupervised approach, which nevertheless is able to localize many phrases correctly. Figure 6.4 presents qualitative results on ReferItGame dataset. We show the predictions of our supervised approach, as well as the ground-truth boxes. One can see the difficulty of the task from the presented examples, including two failures in the bottom row. One requires good language understanding in order to correctly ground such complex phrases. In order to ground expressions like "hut to the nearest left of the person on the right" we would need to additionally model relations between objects, an interesting direction for future work.

two people on right

picture of a bird flying above sand



dat alpaca up in front, total coffeelate swag



palm tree coming out of the top of the building

guy with blue shirt and yellow shorts

hut to the nearest left of the person on the right

Figure 6.4: Qualitative results on the test set of ReferItGame: GroundeR (VGG+SPAT) supervised. Green: ground-truth box, red: predicted box.

6.5 CONCLUSION

In this chapter we address the challenging task of grounding unconstrained natural phrases in images. We consider different scenarios of available bounding box supervision at training time, namely none, little, and full supervision. We propose a novel approach, GroundeR, which learns to localize phrases in images by attending to the correct box proposal and reconstructing the phrase and is able to operate in all of these supervision scenarios. In the unsupervised scenario we are competitive or better than related work. Our semi-supervised approach works well with a small portion of available annotated data and takes advantage of the unsupervised data to outperform purely supervised training using the same amount of labeled data. It outperforms state-of-the-art, both on Flickr 30k Entities and ReferItGame dataset, by 4.5% and 10.6%, respectively.

Our approach is rather general and it could be applied to other regions such as segmentation proposals instead of bounding box proposals. In Chapter 8 it is applied to associate people head tracks with their names. Possible extensions for our approach are to include constraints within sentences at training time, jointly reason about multiple phrases, and to take into account spatial relations between them.

7 MULTIMODAL COMPACT BILINEAR POOLING FOR VISUAL QUESTION ANSWERING AND VISUAL GROUNDING

Modeling textual or visual information with vector representations trained from large language or visual datasets has been successfully explored in recent years. However, tasks such as visual question answering and visual grounding, discussed in the previous chapter, require combining these vector representations with each other. Approaches to multimodal pooling include element-wise product or sum, as well as concatenation of the visual and textual representations. We hypothesize that these methods are not as expressive as an outer product of the visual and textual vectors. As the outer product is typically infeasible due to its high dimensionality, we instead propose utilizing Multimodal Compact Bilinear pooling (MCB) to efficiently and expressively combine multimodal features. We extensively evaluate MCB on the visual question answering and grounding tasks. We consistently show the benefit of MCB over ablations without MCB. For visual question answering, we present an architecture which uses MCB twice, once for predicting attention over spatial features and again to combine the attended representation with the question representation. This model outperforms the state-of-the-art on the Visual7W dataset and the VQA challenge. For the visual grounding we replace the multi-modal combination by concatenation with MCB in our approach presented in the previous chapter, and improve over the state-of-the-art on two datasets.

7.1 INTRODUCTION

Representation learning for text and images has been extensively studied in recent years. Recurrent neural networks (RNNs) are often used to represent sentences or phrases (Sutskever *et al.*, 2014; Kiros *et al.*, 2015b), and convolutional neural networks (CNNs) have shown to work best to represent images (Donahue *et al.*, 2013; He *et al.*, 2016). For tasks such as visual question answering (VQA) and visual grounding, most approaches require joining the representation of both modalities. For combining the two vector representations (multimodal pooling), current approaches in VQA or grounding rely on concatenating vectors or applying element-wise sum or product. While this generates a joint representation, it might not be expressive enough to fully capture the complex associations between the two different modalities.

In this chapter, we propose to rely on Multimodal Compact Bilinear pooling (MCB) to get a joint representation. Bilinear pooling computes the outer product between two vectors, which allows, in contrast to element-wise product, a multiplicative interaction between all elements of both vectors. Bilinear pooling models

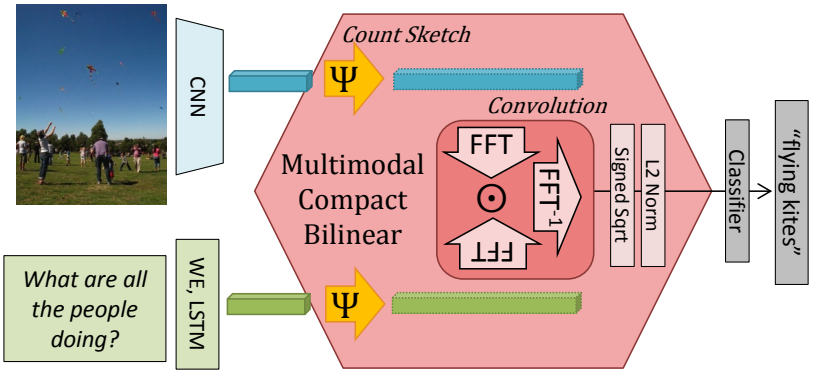


Figure 7.1: Multimodal Compact Bilinear Pooling for visual question answering.

(Tenenbaum and Freeman, 2000) have been shown to be beneficial for fine-grained classification for vision only tasks Lin *et al.* (2015b). However, given their high dimensionality (n^2), bilinear pooling has so far not been widely used. In this chapter, we adopt the idea from Gao *et al.* (2016b) which shows how to efficiently compress bilinear pooling for a single modality. In this chapter, we discuss and extensively evaluate the extension to the multimodal case for text and visual modalities. As shown in Figure 7.1, Multimodal Compact Bilinear pooling (MCB) is approximated by randomly projecting the image and text representations to a higher dimensional space (using Count Sketch (Charikar *et al.*, 2002)) and then convolving both vectors efficiently by using element-wise product in Fast Fourier Transform (FFT) space. We use MCB to predict answers for the VQA task and locations for the visual grounding task. For open-ended question answering, we present an architecture for VQA which uses MCB twice, once to predict spatial attention and the second time to predict the answer. For multiple-choice question answering we introduce a third MCB to relate the encoded answer to the question-image space. Additionally, we discuss the benefit of attention maps and additional training data for the VQA task. To summarize, MCB is evaluated on two tasks, four datasets, and with a diverse set of ablations and comparisons to the state-of-the-art.

7.2 RELATED WORK

In the following we discuss which multimodal pooling techniques have been employed in various recent VQA and visual grounding approaches. We then review the works which like us adopt bilinear pooling for computer vision tasks. Finally, we discuss the relation of our work to multimodal embedding learning.

Multimodal pooling. Current approaches to multimodal pooling involve element-wise operations or vector concatenation. In the visual question answering domain, a number of models have been proposed. Simpler models such as iBOWIMG baseline (Zhou *et al.*, 2015a) use concatenation and fully connected layers to combine the image and question modalities. Stacked Attention Networks (Yang *et al.*, 2016a) and Spatial Memory Networks (Xu *et al.*, 2015a) use LSTMs or extract soft-attention on

the image features, but ultimately use element-wise product or element-wise sum to merge modalities. D-NMN (Andreas *et al.*, 2016a) introduced REINFORCE to dynamically create a network and use element-wise product to join attentions and element-wise sum predict answers. Dynamic Memory Networks (DMN) (Xiong *et al.*, 2016) pool the image and question with element-wise product and sum, attending to part of the image and question with an Episodic Memory Module (Kumar *et al.*, 2016). DPPnet (Noh *et al.*, 2015) creates a Parameter Prediction Network which learns to predict the parameters of the second to last visual recognition layer dynamically from the question. Similar to this work, DPPnet allows multiplicative interactions between the visual and question encodings. Lu *et al.* (2016) recently proposed a model that extracts multiple co-attentions on the image and question and combines the co-attentions in a hierarchical manner using element-wise sum, concatenation, and fully connected layers. For the visual grounding task, in the previous chapter we presented our approach GroundeR, where the language phrase embedding is concatenated with the visual features in order to predict the attention weights over multiple bounding box proposals. Similarly, Hu *et al.* (2016a) concatenate phrase embeddings with visual features at different spatial locations to obtain a segmentation.

Bilinear pooling. Bilinear pooling has been applied to the fine-grained visual recognition task. Lin *et al.* (2015b) use two CNNs to extract features from an image and combine the resulting vectors using an outer product, which is fully connected to an output layer. Gao *et al.* (2016b) address the space and time complexity of bilinear features by viewing the bilinear transformation as a polynomial kernel. Pham and Pagh (2013) describe a method to approximate the polynomial kernel using Count Sketches and convolutions.

Joint multimodal embeddings. In order to model similarities between two modalities, many prior works have learned joint multimodal spaces, or embeddings. Some of such embeddings are based on Canonical Correlation Analysis (Hardoon *et al.*, 2004) e.g. (Gong *et al.*, 2014; Klein *et al.*, 2015; Plummer *et al.*, 2015), linear models with ranking loss (Frome *et al.*, 2013; Karpathy and Fei-Fei, 2015; Socher *et al.*; Weston *et al.*, 2011) or non-linear deep learning models (Kiros *et al.*, 2014; Mao *et al.*, 2015; Ngiam *et al.*, 2011). Our MCB pooling can be seen as a complementary operation that allows us to capture different interactions between two modalities more expressively than e.g. concatenation. Consequently, many embedding learning approaches could benefit from incorporating such interactions.

7.3 MULTIMODAL COMPACT BILINEAR POOLING FOR VISUAL AND TEXTUAL EMBEDDINGS

For the task of visual question answering (VQA) or visual grounding, we have to predict the most likely answer or location $\hat{a}$ for a given image $\mathbf{x}$ and question or

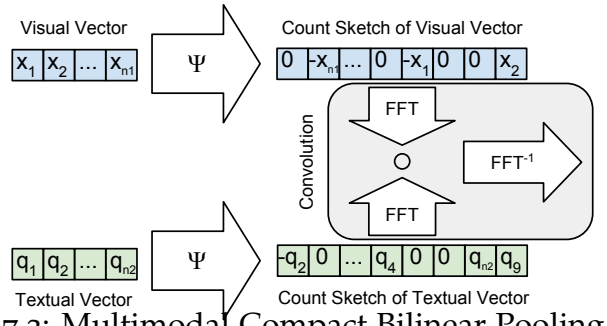


Figure 7.2: Multimodal Compact Bilinear Pooling (MCB)

phrase $\mathbf{q}$. This can be formulated as

$$\hat{a} = \underset{a \in A}{\operatorname{argmax}}\, p(a|\mathbf{x}, \mathbf{q}; \theta) \tag{7.1}$$

with parameters θ and the set of answers or locations A. For an image embedding $x = \Xi(\mathbf{x})$ (i.e. a CNN) and question embedding $q = \Omega(\mathbf{q})$ (i.e. an LSTM), we are interested in getting a good joint representation by pooling both representations. With a multimodal pooling $\Phi(x, q)$ that encodes the relationship between x and q well, it becomes easier to learn a classifier for Equation (7.1).

In this section, we first discuss our multimodal pooling Φ for combining representations from different modalities into a single representation (Section 7.3.1) and then detail our architectures for VQA (Section 7.3.2) and visual grounding (Section 7.3.3), further explaining how we predict $\hat{a}$ with the given image representation Ξ and text representation Ω.

7.3.1 Multimodal Compact Bilinear Pooling (MCB)

Bilinear models (Tenenbaum and Freeman, 2000) take the outer product of two vectors $x \in \mathbb{R}^{n_1}$ and $q \in \mathbb{R}^{n_2}$ and learn a model W (here linear), i.e. $z = W[x \otimes q]$, where $\otimes$ denotes the outer product (xq^T) and $[\,]$ denotes linearizing the matrix in a vector. As discussed in the introduction, bilinear pooling is interesting because it allows all elements of both vectors to interact with each other in a multiplicative way. However, the high dimensional representation (i.e. when n_1 and n_2 are large) leads to an infeasible number of parameters to learn in W. For example, we use $n_1 = n_2 = 2048$ and $z \in \mathbb{R}^{3000}$ for VQA. W thus would have 12.5 billion parameters, which leads to very high memory consumption and high computation times.

We thus need a method that projects the outer product to a lower dimensional space and also avoids computing the outer product directly. As suggested by Gao *et al.* (2016b) for a single modality, we rely on the Count Sketch projection function Ψ (Charikar *et al.*, 2002), which projects a vector $v \in \mathbb{R}^n$ to $y \in \mathbb{R}^d$. We initialize two vectors $s \in \{-1, 1\}^n$ and $h \in \{1, ..., d\}^n$: s contains either 1 or -1 for each index, and h maps each index i in the input v to an index j in the output y. Both s and h are initialized randomly from a uniform distribution and remain constant for future invocations of count sketch. y is initialized as a zero vector. For every element $v[i]$

Algorithm 1 Multimodal Compact Bilinear

1: input: $v_1 \in \mathbb{R}^{n_1}, v_2 \in \mathbb{R}^{n_2}$
2: output: $\Phi(v_1, v_2) \in \mathbb{R}^d$
3: **procedure** MCB(v_1, v_2, n_1, n_2, d)
4: **for** $k \leftarrow 1 \dots 2$ **do**
5: **if** h_k, s_k not initialized **then**
6: **for** $i \leftarrow 1 \dots n_k$ **do**
7: sample $h_k[i]$ from $\{1, \dots, d\}$
8: sample $s_k[i]$ from $\{-1, 1\}$
9: $v'_k = \Psi(v_k, h_k, s_k, n_k)$
10: $\Phi = \text{FFT}^{-1}(\text{FFT}(v'_1) \odot \text{FFT}(v'_2))$
11: **return** Φ
12: **procedure** $\Psi(v, h, s, n)$
13: $y = [0, \dots, 0]$
14: **for** $i \leftarrow 1 \dots n$ **do**
15: $y[h[i]] = y[h[i]] + s[i] \cdot v[i]$
16: **return** y

its destination index $j = h[i]$ is looked up using h, and $s[i] \cdot v[i]$ is added to $y[j]$. See lines 1-9 and 12-16 in Algorithm 1.

This allows us to project the outer product to a lower dimensional space, which reduces the number of parameters in W. To avoid computing the outer product explicitly, Pham and Pagh (2013) showed that the count sketch of the outer product of two vectors can be expressed as convolution of both count sketches: $\Psi(x \otimes q, h, s) = \Psi(x, h, s) * \Psi(q, h, s)$, where $*$ is the convolution operator. Additionally, the convolution theorem states that convolution in the time domain is equivalent to element-wise product in the frequency domain. The convolution $x' * q'$ can be rewritten as $\text{FFT}^{-1}(\text{FFT}(x') \odot \text{FFT}(q'))$, where $\odot$ refers to element-wise product. These ideas are summarized in Figure 7.2 and formalized in Algorithm 1, which is based on the Tensor Sketch algorithm of Pham and Pagh (2013). We invoke the algorithm with $v_1 = x$ and $v_2 = q$. We note that this easily extends and remains efficient for more than two multi-modal inputs as the combination happens as element-wise product.

7.3.2 Architectures for VQA

In VQA, the input to the model is an image and a question, and the goal is to answer the question. Our model extracts representations for the image and the question, pools the vectors using MCB, and arrives at the answer by treating the problem as a multi-class classification problem with 3,000 possible classes.

We extract image features using a 152-layer Residual Network (He *et al.*, 2016) that is pretrained on ImageNet data (Deng *et al.*, 2009). Images are resized to 448×448, and we use the output of the layer ("pool5") before the 1000-way classifier. We then perform L_2 normalization on the 2048-D vector.

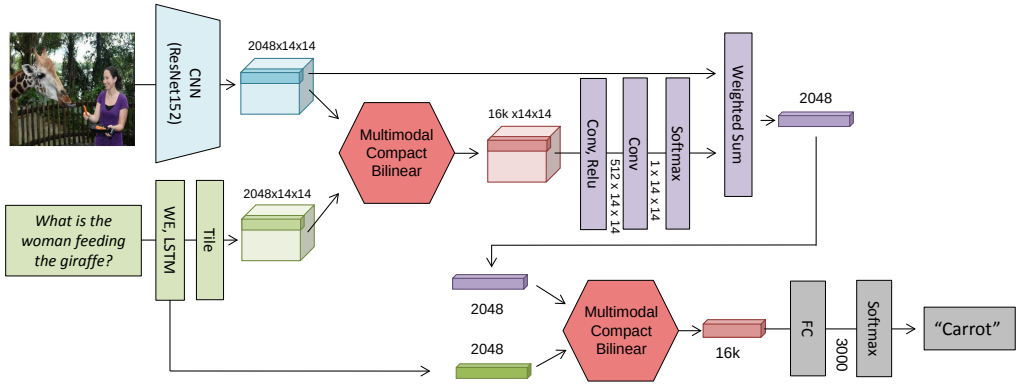


Figure 7.3: Our architecture for VQA: Multimodal Compact Bilinear (MCB) with Attention. Conv implies convolutional layers and FC implies fully connected layers. For details see Section 7.3.2.

Input questions are first tokenized into words, the words are one-hot encoded and passed through a learned embedding layer. The *tanh* nonlinearity is used after the embedding, followed by a 2-layer LSTM with 1024 units in each layer. The outputs of each LSTM layer are concatenated to form a 2048-D vector.

The two vectors are then passed through MCB. The MCB is followed by an element-wise signed square-root and L_2 normalization. After MCB pooling, a fully connected layer connects the resulting 16,000-D multimodal representation to the 3,000 top answers.

Attention. To incorporate spatial information, we use soft attention on our MCB pooling method. Explored by Xu *et al.* (2015a) for image captioning and by Xu and Saenko (2016) and Yang *et al.* (2016a) for VQA, the soft attention mechanism can be easily integrated in our model.

For each spatial grid location in the visual representation (i.e. last convolutional layer of ResNet [res5c], last convolutional layer of VGG [conv5]), we use MCB pooling to merge the slice of the visual feature with the language representation. As depicted in Figure 7.3, after the pooling we use two convolutional layers to predict the attention weight for each grid location. We apply softmax to produce a normalized soft attention map. We then take a weighted sum of the spatial vectors using the attention map to create the attended visual representation. We also experiment with generating multiple attention maps to allow the model to make multiple "glimpses" which are concatenated before being merged with the language representation through another MCB pooling for prediction. Predicting attention maps with MCB pooling allows the model to effectively learn how to attend to salient locations based on both the visual and language representations.

Answer encoding. For VQA with multiple choices, we can additionally embed the answers. We base our approach on the proposed MCB with attention. As can be seen from Figure 7.4, to deal with multiple variable-length answer choices, each choice is encoded using a word embedding and LSTM layers whose weights are

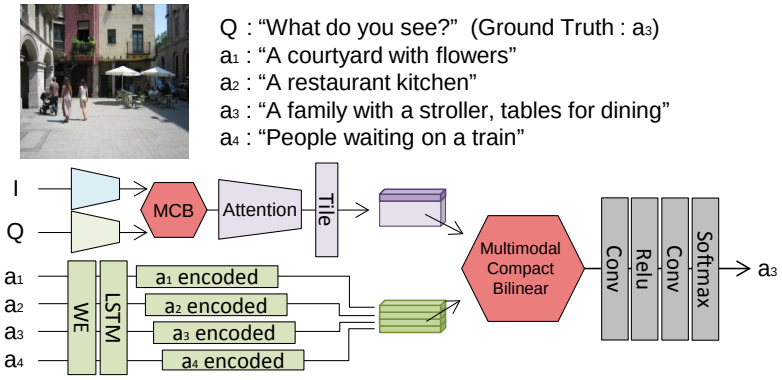


Figure 7.4: Our architecture for VQA: MCB with Attention and Answer Encoding

shared across the candidates. In addition to using MCB with attention, we use an additional MCB pooling to merge the encoded answer choices with the multimodal representation of the original pipeline. The resulting embedding is projected to a classification vector with a dimension equal to the number of answers.

7.3.3 Architecture for visual grounding

We base our grounding approach on the fully-supervised version of GroundeR (Chapter 6). The overview of our model is shown in Figure 7.5. The input to the model is a query natural language phrase and an image along with multiple proposal bounding boxes. The goal is to predict a bounding box which corresponds to the query phrase. We replace the concatenation of the visual representation and the encoded phrase in GroundeR with MCB to combine both modalities. In contrast to the previous chapter, we include a linear embedding of the visual representation and L_2 normalization of both input modalities, instead of batch normalization (Ioffe and Szegedy, 2015), which we found to be beneficial when using MCB for the grounding task.

7.4 EVALUATION ON VISUAL QUESTION ANSWERING

We evaluate the benefit of MCB with a diverse set of ablations on two visual question answering datasets.

7.4.1 Datasets

The **Visual Question Answering (VQA)** real-image dataset (Antol *et al.*, 2015) consists of approximately 200,000 MSCOCO images (Lin *et al.*, 2014b), with 3 questions per image and 10 answers per question. There are 3 data splits: train (80K images), validation (40K images), and test (80K images). Additionally, there is a 25% subset of test named test-dev. Accuracies for ablation experiments are reported on the test-dev

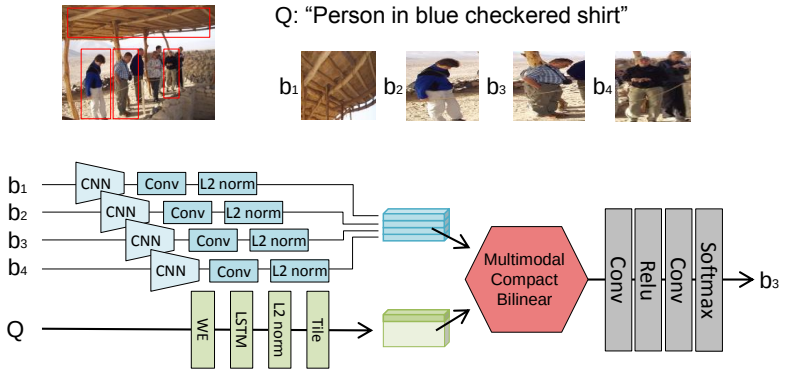


Figure 7.5: Our Architecture for Grounding with MCB (Sec. 7.3.3)

data split. We use the VQA tool provided by Antol *et al.* (2015) for evaluation. We conducted most of our experiments on the open-ended real-image task. In Table 7.4, we also report our multiple-choice real-image scores.

The **Visual Genome** dataset (Krishna *et al.*, 2016) uses 108,249 images from the intersection of YFCC100M (Thomee *et al.*, 2015) and MSCOCO. For each image, an average of 17 question-answer pairs are collected. There are 1.7 million QA pairs of the 6W question types (*what, where, when, who, why*, and *how*). Compared to the VQA dataset, Visual Genome represents a more balanced distribution of the 6W questions. The average question and answer lengths for Visual Genome are larger than for the VQA dataset. To leverage the Visual Genome dataset as additional training data, we remove all the unnecessary words such as ”a”, ”the”, and ”it is” from the answers to decrease the length of the answers and extract QA pairs whose answers are single-worded. The extracted data is filtered based on the answer vocabulary space created from the VQA dataset, leaving us with additional 1M image-QA triplets.

The **Visual7W** dataset (Zhu *et al.*, 2016) is a part of the Visual Genome. It adds a 7th *which* question category to accommodate visual answers, but we only evaluate the models on the Telling task which involves 6W questions. The natural language answers in Visual7W are in a multiple-choice format and each question comes with four answer candidates, with only one being the correct answer. Visual7W is composed of 47,300 images from MSCOCO and there are a total of 139,868 QA pairs.

7.4.2 Experimental setup

We use the Adam solver (Kingma and Ba, 2014) with $\epsilon = 0.0007$, $\beta_1 = 0.9$, $\beta_2 = 0.999$. We use dropout after the LSTM layers and in fully connected layers. For the experiments in Tables 7.1 and 7.2, we train on the VQA train split, validate on the VQA validation split, and report results on the VQA test-dev split. We use early stopping: if the validation score does not improve for 50,000 iterations, we stop training and evaluate the best iteration on test-dev. For the Visual7W, we use the same hyperparameters and training settings as in the VQA experiments. We use the splits from Zhu *et al.* (2016) to train, validate, and test our models. We also compute

Method	Accuracy
Element-wise Sum	56.50
Concatenation	57.49
Concatenation + FC	58.40
Concatenation + FC + FC	57.10
Element-wise Product	58.57
Element-wise Product + FC	56.44
Element-wise Product + FC + FC	57.88
MCB ($2048 \times 2048 \rightarrow 16K$)	**59.83**
Full Bilinear ($128 \times 128 \rightarrow 16K$)	58.46
MCB ($128 \times 128 \rightarrow 4K$)	58.69
Element-wise Product with VGG-19	55.97
MCB ($d = 16K$) with VGG-19	**57.05**
Concatenation + FC with Attention	58.36
MCB ($d = 16K$) with Attention	**62.50**

Table 7.1: Comparison of multimodal pooling methods. Models are trained on the VQA train split and tested on test-dev.

accuracies on this data using their evaluation code. For VQA multiple choice, we train the open-ended models and take the argmax over the multiple choice answers at test time. For Visual7W, we use the answer encoding as described in Section 7.3.2.

7.4.3 Ablation results

We compare the performance of non-bilinear and bilinear pooling methods in Table 7.1. We see that MCB pooling outperforms all non-bilinear pooling methods, such as eltwise sum, concatenation, and eltwise product.

One could argue that the compact bilinear method simply has more parameters than the non-bilinear pooling methods, which contributes to its performance. We compensated for this by stacking fully connected layers (with 4096 units per layer, ReLU activation, and dropout) after the non-bilinear pooling methods to increase their number of parameters. However, even with similar parameter budgets, non-bilinear methods could not achieve the same accuracy as the MCB method. For example, the "Concatenation + FC + FC" pooling method has approximately $4096^2 + 4096^2 + 4096 \times 3000 \approx 46$ million parameters, which matches the 48 million parameters available in MCB with $d = 16000$. However, the performance of the "Concatenation + FC + FC" method is only 57.10% compared to MCB's 59.83%.

Section 2 in Table 7.1 also shows that compact bilinear pooling has no impact on accuracy compared to full bilinear pooling. Section 3 in Table 7.1 demonstrates that the MCB brings improvements regardless of the image CNN used. We primarily use ResNet-152 in this work, but MCB also improves performance if VGG-19 is used.

Compact Bilinear d	**Accuracy**
1024	58.38
2048	58.80
4096	59.42
8192	59.69
16000	**59.83**
32000	59.71

Table 7.2: Accuracies for different values of d, the dimension of the compact bilinear feature. Models are trained on the VQA train split and tested on test-dev. Details in Section 7.4.3.

Method	What	Where	When	Who	Why	How	Avg
Zhu et al.	51.5	57.0	75.0	59.5	55.5	49.8	54.3
Concat+Att.	47.8	56.9	74.1	62.3	52.7	**51.2**	52.8
MCB+Att.	**60.3**	**70.4**	**79.5**	**69.2**	**58.2**	51.1	**62.2**

Table 7.3: Multiple-choice QA tasks accuracy (%) on Visual7W test set.

Section 4 in Table 7.1 shows that our soft attention model works best with MCB pooling. In fact, attending to the Concatenation + FC layer has the same performance as not using attention at all, while attending to the MCB layer improves performance by 2.67 points.

Table 7.2 compares different values of d, the output dimensionality of the multimodal compact bilinear feature. Approximating the bilinear feature with a 16,000-D vector yields the highest accuracy.

We also evaluated models with multiple attention maps or channels. One attenion map achieves 64.67%, two 65.08% and four 64.24% accuracy (trained on train+val). Visual inspection of the generated attention maps reveals that an ensembling or smoothing effect occurs when using multiple maps.

Table 7.3 presents results for the Visual7W multiple-choice QA task. The MCB with attention model outperforms the previous state-of-the-art by 7.9 points overall and performs better in almost every category.

7.4.4 Comparison to state-of-the-art

Table 7.4 compares our approach with the state-of-the-art on VQA test set. Our best single model uses MCB pooling with two attention maps. Additionally, we augment our training data with images and QA pairs from the Visual Genome dataset. We also concatenate the learned word embedding with pretrained GloVe vectors (Pennington *et al.*, 2014).

Each model in our ensemble of 7 models uses MCB with attention. Some of the models were trained with data from Visual Genome, and some were trained

	Test-dev					Test-standard				
	Open Ended				MC	Open Ended				MC
	Y/N	No.	Other	All	All	Y/N	No.	Other	All	All
MCB	81.2	35.1	49.3	60.8	65.4	-	-	-	-	-
MCB+Genome	81.7	36.6	51.5	62.3	66.4	-	-	-	-	-
MCB+Att.	82.2	37.7	54.8	64.2	68.6	-	-	-	-	-
MCB+Att.+GloVe	82.5	37.6	55.6	64.7	69.1	-	-	-	-	-
MCB+Att.+Genome	81.7	38.2	57.0	65.1	69.5	-	-	-	-	-
MCB+Att.+GloVe+Genome	82.3	37.2	57.4	65.4	69.9	-	-	-	-	-
Ensemble of 7 Att. models	**83.4**	**39.8**	**58.5**	**66.7**	**70.2**	**83.2**	**39.5**	**58.0**	**66.5**	**70.1**
Naver Labs (challenge 2nd)	83.5	39.8	54.8	64.9	69.4	83.3	38.7	54.6	64.8	69.3
HieCoAtt (Lu *et al.*, 2016)	79.7	38.7	51.7	61.8	65.8	-	-	-	62.1	66.1
DMN+ (Xiong *et al.*, 2016)	80.5	36.8	48.3	60.3	-	-	-	-	60.4	-
FDA (Ilievski *et al.*, 2016)	81.1	36.2	45.8	59.2	-	-	-	-	59.5	-
D-NMN (Andreas *et al.*, 2016a)	81.1	38.6	45.5	59.4	-	-	-	-	59.4	-
AMA (Wu *et al.*, 2016)	81.0	38.4	45.2	59.2	-	81.1	37.1	45.8	59.4	-
SAN (Yang *et al.*, 2016a)	79.3	36.6	46.1	58.7	-	-	-	-	58.9	-
NMN (Andreas *et al.*, 2016b)	81.2	38.0	44.0	58.6	-	81.2	37.7	44.0	58.7	-
AYN (Malinowski *et al.*, 2016)	78.4	36.4	46.3	58.4	-	78.2	36.3	46.3	58.4	-
SMem (Xu and Saenko, 2016)	80.9	37.3	43.1	58.0	-	80.9	37.5	43.5	58.2	-
VQA team (Antol *et al.*, 2015)	80.5	36.8	43.1	57.8	62.7	80.6	36.5	43.7	58.2	63.1
DPPnet (Noh *et al.*, 2015)	80.7	37.2	41.7	57.2	-	80.3	36.9	42.2	57.4	-
iBOWIMG (Zhou *et al.*, 2015a)	76.5	35.0	42.6	55.7	-	76.8	35.0	42.6	55.9	62.0

Table 7.4: Open-ended and multiple-choice (MC) results on VQA test set (trained on train+val set) compared with state-of-the-art: accuracy in %. See Section 7.4.4.

with two attention maps. This ensemble is 1.8 points above the next best approach on the VQA open-ended task and 0.8 points above the next best approach on the multiple-choice task (on Test-dev). Even without ensembles, our "MCB + Genome + Att. + GloVe" model still outperforms the next best result by 0.5 points, with an accuracy of 65.4% versus 64.9% on the open-ended task (on Test-dev).

7.5 EVALUATION ON VISUAL GROUNDING

7.5.1 Datasets

We evaluate our visual grounding approach on two datasets. The first is Flickr30k Entities (Plummer *et al.*, 2015) which consists of 31K images from Flickr30k dataset (Hodosh *et al.*, 2014) with 244K phrases localized with bounding boxes. We follow the experimental setup from the previous chapter, e.g. we use the same Selective

[1]Plummer *et al.* (2016) achieve higher accuracy of 50.89% when taking into account box size and color. We believe our approach would also benefit from such additional features.

Method	Accuracy, %
Plummer *et al.* (2015)	27.42
Hu *et al.* (2016b)	27.80
Plummer *et al.* (2016)[1]	43.84
Wang *et al.* (2016a)	43.89
GroundeR (Chapter 6)	47.81
Concatenation	46.50
Element-wise Product	47.41
Element-wise Product + Conv	47.86
MCB	**48.69**

Table 7.5: Grounding accuracy on Flickr30k Entities dataset.

Method	Accuracy, %
Hu *et al.* (2016b)	17.93
GroundeR (Chapter 6)	26.93
Concatenation	25.48
Element-wise Product	27.80
Element-wise Product + Conv	27.98
MCB	**28.91**

Table 7.6: Grounding accuracy on ReferItGame dataset.

Search (Uijlings *et al.*, 2013) object proposals and the Fast R-CNN (Girshick, 2015a) fine-tuned VGG16 features (Simonyan and Zisserman, 2015). The second dataset is ReferItGame (Kazemzadeh *et al.*, 2014), which contains 20K images from IAPR TC-12 dataset (Grubinger *et al.*, 2006) with segmented regions from SAIAPR-12 dataset (Escalante *et al.*, 2010) and 120K associated natural language referring expressions. For ReferItGame we follow the experimental setup of Hu *et al.* (2016b) and rely on their ground-truth bounding boxes extracted around the segmentation masks. We use the Edge Box (Zitnick and Dollár, 2014) object proposals and visual features (VGG16 combined with the spatial features, which encode bounding box relative position) from Hu *et al.* (2016b).

7.5.2 Experimental setup

In all experiments we use the Adam solver with $\epsilon = 0.0001$. The embedding size is 500 both for visual and language embeddings. We use $d = 2048$ in the MCB pooling, which we found to work best for the visual grounding task. The accuracy is measured as percentage of query phrases which have been localized correctly. The phrase is localized correctly if the predicted bounding box overlaps with the ground-truth bounding box by more than 50% intersection over union (IOU).

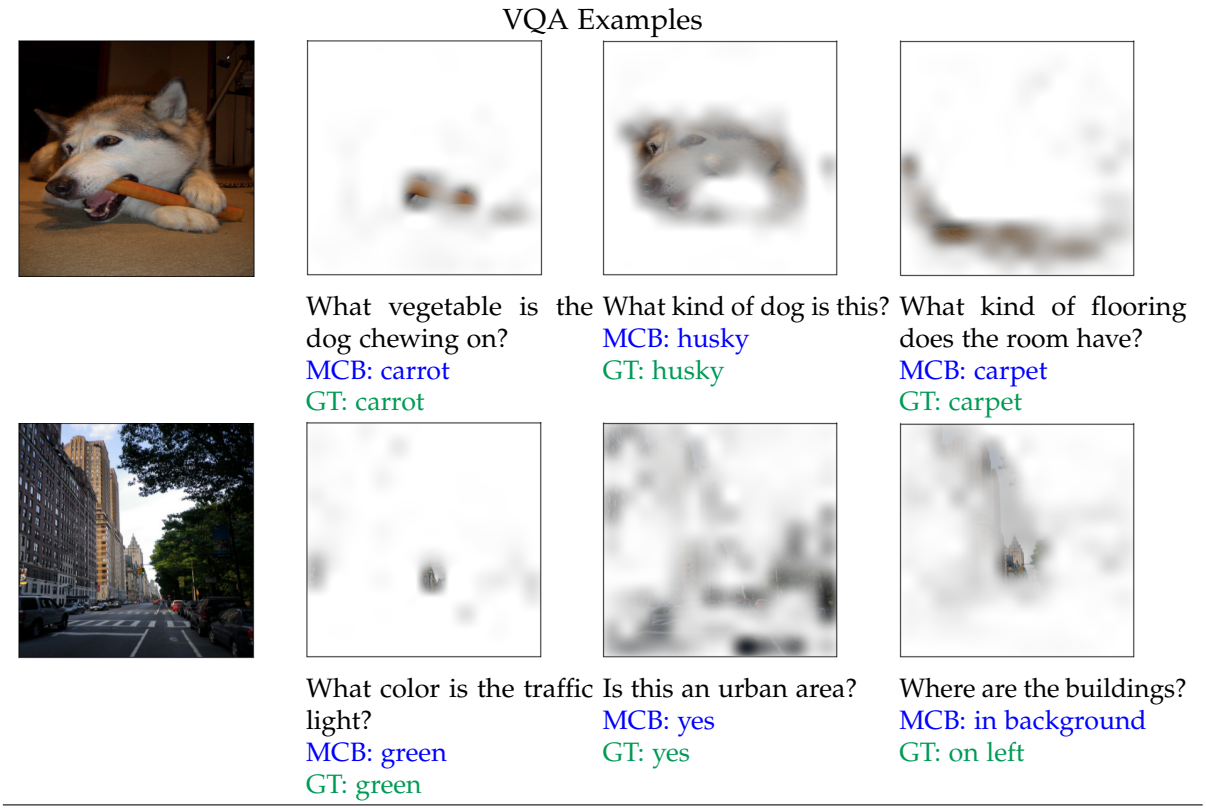


Figure 7.6: Top: predicted answers and attention maps from MCB model on VQA images. Bottom: predicted grounding from MCB model (left) and Eltwise Product + Conv model (right) on Flickr30k Entities images.

7.5.3 Results

Tables 7.5 and 7.6 summarize our results in the visual grounding task. We present multiple ablations of our proposed architecture. First, we replace the MCB with simple concatenation of the embedded visual feature and the embedded phrase, resulting in 46.5% on the Flickr30k Entities and 25.48% on the ReferItGame datasets. The results can be improved by replacing the concatenation with the element-wise product of both embedded features (47.41% and 27.80%). We can further slightly increase the performance by introducing additional 2048-D convolution after the element-wise product (47.86% and 27.98%). However, even with fewer parameters, our MCB pooling significantly improves over this baseline on both datasets, reaching state-of-the-art accuracy of 48.69% on Flickr30k Entities and 28.91% on ReferItGame dataset. Figure 7.6 (bottom) shows examples of improved phrase localization.

7.6 CONCLUSION

In this chapter we propose Multimodal Compact Bilinear Pooling (MCB) to combine visual and text representations. For visual question answering, our architecture with attention and multiple MCBs gives significant improvements on two VQA datasets compared to state-of-the-art. In the visual grounding task, introducing MCB pooling into our grounding approach (Chapter 6) leads to improved phrase localization accuracy, indicating better interaction between query phrase representations and visual representations of proposal bounding boxes.

8 GENERATING DESCRIPTIONS WITH GROUNDED AND CO-REFERENCED PEOPLE

Learning how to generate descriptions of images or videos received major interest both in the computer vision and natural language processing communities. In Chapters 4 and 5 we have presented our own contributions to this end. While a few works have proposed to learn a grounding during the generation process in an unsupervised way (via an attention mechanism), it remains unclear how good the quality of the grounding is and whether it benefits the description quality. In this chapter we propose a movie description model which learns to generate description and jointly ground (localize) the mentioned characters as well as do visual co-reference resolution between pairs of consecutive sentences/clips. We also propose to use weak localization supervision through character mentions provided in movie descriptions to learn the character grounding. At training time, we first learn how to localize characters by relating their visual appearance to mentions in the descriptions via our semi-supervised approach from Chapter 6. We then provide this (noisy) supervision into our description model which greatly improves its performance. Our proposed description model improves over prior work w.r.t. generated description quality and additionally provides grounding and local co-reference resolution. We evaluate it on the MPII Movie Description dataset (Chapter 5) using automatic and human evaluation measures and using our newly collected grounding and co-reference data for characters.

8.1 INTRODUCTION

When humans talk about what they see, they not only use common objects and terms, but typically refer to reappearing entities, most commonly using names ("John") and referential words such as pronouns ("he", "it"). To correctly generate descriptions with reappearing entities, one needs to understand and link them across sentences and visual appearances (images/frames). Current image/video captioning datasets essentially ignore this aspect as they ask to independently describe each image/clip with a single sentence. At the same time, e.g. visual storytelling (Huang *et al.*, 2016) and movie description (Chapter 5) ultimately require solving this problem. However, the first approaches on visual storytelling (Huang *et al.*, 2016) so far have not taken it into account, and current movie description challenges and approaches (Chapter 5) abstract from it by looking at a single clip at a time and replacing all the character mentions with e.g. "Someone".

In this chapter we address grounded co-reference resolution, with application to movie description. The most prominent entities in movies are the people or

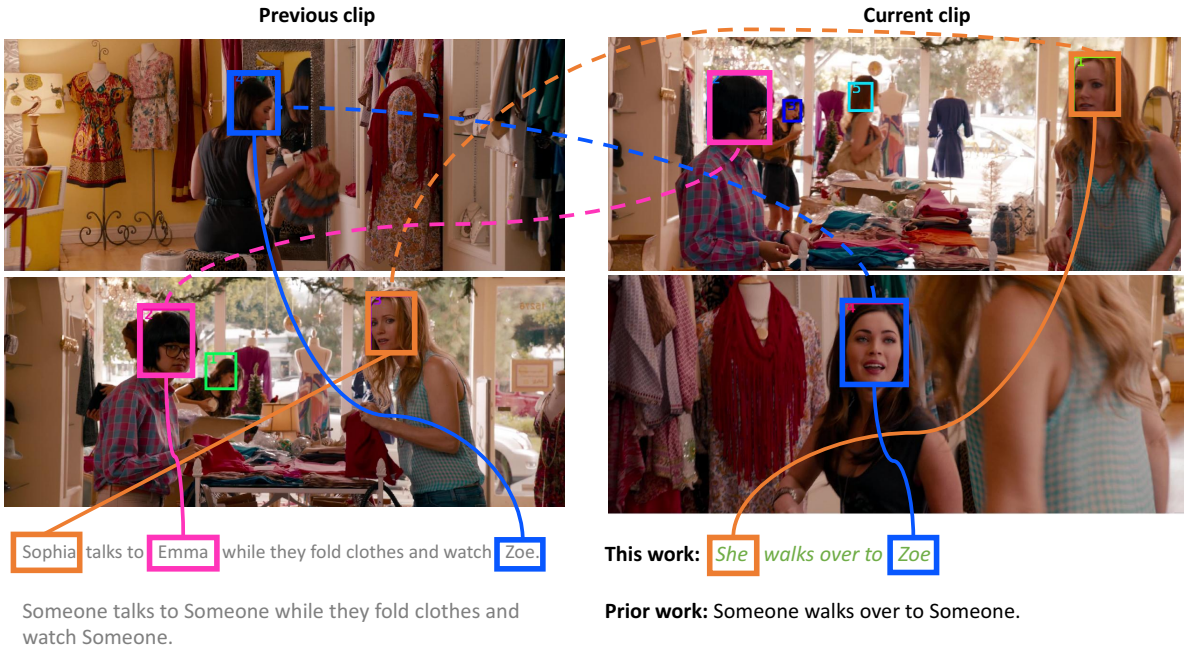


Figure 8.1: Bring in the color: our task is to generate grounded and co-referenced descriptions for the current clip using pronouns and new or reappearing character IDs, which are grounded, i.e. localized in the current clip (boxes and lines) and visually co-referenced to the previous clip (dashed lines). The visual grounding allows for co-reference to the previous clip/sentence which enables us using the pronoun "she" to refer to the first ID (Sophia).

characters. In fact, there is a long line of work which aims to link character mentions in movie or TV scripts with their visual tracks (Cour *et al.*, 2009; Everingham *et al.*, 2006; Sivic *et al.*, 2009; Tapaswi *et al.*, 2012; Parkhi *et al.*, 2015a; Bojanowski *et al.*, 2013; Ramanathan *et al.*, 2014). However, all these works are already given the description for all movies where they want to predict the linking. In contrast we want to generate a description, while jointly linking it with the currently and previously depicted character's visual presence. Specifically, the task we address in this work is to *generate descriptions* for movies and at the same time localize or *ground* the characters, recognize their gender and refer to them consistently, i.e. *co-reference* them across sentences, as visualized in Figure 8.1. Importantly, rather than trying to obtain consistent ids in the entire movie, we focus on robust *local* co-reference resolution on *two consecutive sentences/clips*. We argue that local co-reference resolution is an important problem on itself. On the one hand there are many characters without proper names and/or with only a few occurrences, which can and should be resolved locally, e.g. "*The priest* takes their vows. *He* declares them wife and husband". On the other hand, there are many hard decisions which have to be made locally, e.g. which character to describe and whether a character should be referenced by proper name or pronoun. To clarify, we do not generate the true proper names of the characters, but only identities with gender. We use a predefined set of names in our examples (e.g. Sophia). In future work we believe the true names could be extracted either from dialog, or from one/a few annotations per character.

Approaching the joint description and grounding task requires three main ingre-

dients: we need to *localize the characters*, we need to decide which character(s) to *pay attention to*, and we need to *co-reference* visual characters appearances in neighboring sentences/clips. In Section 8.4 we detail how we approach *character localization* using head detection and tracking via a two-stage clustering approach.While generating the sentence, we advocate to *jointly* decide which character to *pay attention to* and if and how to *co-reference* it to the previous grounded characters. In Section 8.5, we propose to adapt the attention mechanism (Bahdanau *et al.*, 2015; Xu *et al.*, 2015a) for this and extend it to attend *jointly* over both problems: grounding (i.e. track selection) and co-reference (i.e. track linking). A key insight is that this can not be learned purely from sentence supervision for generation. Instead, we supervise the joint-attention mechanism with automatically obtained linking of character mentions and tracks (Section 8.5.2). We note that at test time this supervision is not available and the system has learned, how to jointly ground, co-reference, and describe.

The contributions of this chapter include: a) a new task of movie description with grounded and co-referenced characters; to foster research in this direction we will share our newly collected co-reference annotations and grounding of character mentions in the MPII-MD dataset (Section 8.3); b) a novel approach which addresses this problem by jointly learning to ground the described characters and perform local co-reference resolution between the neighboring clips; c) a robust automatic way of obtaining linking between character mentions in text and visual tracks in video, which we use to supervise our description approach and which we show is essential for the co-reference resolution task.

8.2 RELATED WORK

Our work aims to do three tasks jointly: generating video descriptions, grounding, and co-reference resolution. The prior work on video description and visual grounding has been presented in Section 2.2 and Section 2.3 of the thesis, respectively. In the following we review related work on co-reference resolution. As we focus on people grounding and co-reference, we also discuss the related work on person re-identification and track naming.

Co-reference resolution. Co-reference resolution is task defined in linguistic community (Bergsma and Lin, 2006) where the goal is to establish correct links between named entities and references to then, e.g. pronouns. Ramanathan *et al.* (2014) address co-reference resolution in TV show descriptions with a bidirectional optimization using character visual appearance and linguistic co-reference resolution features.

Person re-identification. Person re-identification from face/head images is a well studied problem and recently many deep learning based approaches have been proposed to address it (Li *et al.*, 2014; Parkhi *et al.*, 2015b; Schroff *et al.*, 2015; Sun *et al.*, 2015; Taigman *et al.*, 2014; Zhou *et al.*, 2015b). Our work is related to this line of work as we aim to re-identify characters between two video clips while generating a

video description.

Linking tracks to names. Related works (Cour *et al.*, 2009; Everingham *et al.*, 2006; Ramanathan *et al.*, 2014; Sivic *et al.*, 2009; Tapaswi *et al.*, 2012) propose datasets for character identification targeting TV shows, which rely on alignment of video to movie/TV scripts. The goal is to track faces in the video and assign names to them. Typically the tracks include background characters. Bojanowski *et al.* (2013) attack the problem of learning a joint model of actors and actions in movies using weak supervision provided by scripts. Parkhi *et al.* (2015a) propose a multiple instance learning based approach which specifically focuses on recognizing background characters and show significant improvement over prior work. While there is a similarity between ours and this line of works, in fact we focus on different tasks. First, we aim to re-identify characters locally, without ever seeing them before. Second, when obtaining the matching between names and visual tracks, our goal is to predict the grounding for a given character, not to name all the tracks.

8.3 A DATASET FOR GROUNDED AND CO-REFERENCED CHARACTERS

One of our goals is to learn visual co-reference resolution. To address this task and evaluate it we collected annotations both on language and visual sides. On the language side we want to know when different mentions actually refer to the same person. On the visual side we require grounding of names to visual appearances. Towards these goals we collect new annotations for character co-reference resolution and grounding for the MPII Movie Description (MPII-MD) dataset (Chapter 5).

Co-reference annotations for character mentions. In the first step, we aim to label all the character mentions in the movie descriptions of the MPII-MD. The standard version of the descriptions consists of sentences with all character names replaced with "Someone" and multiple names (e.g. "Ann and Bob") with "people". Along with the transformed descriptions, the MPII-MD dataset provides the original descriptions with all the character names preserved. We rely on these and run the Stanford Named Entity Recognizer (NER) (Finkel *et al.*, 2005) and obtain our initial name list. We perform manual cleaning and filter out non-human related entities. We also manually check for names missed by NER and add them to our list. With the final name list we label the names in the entire dataset which include many instances missed by the original NER pass. E.g. there might be two different ways of referring to the same character ("Mary Jane" as "MJ"), so we link them together under one "alias". Additionally we annotate the gender of all the characters. In the second step, we annotate pronouns "he" and "she" in all descriptions. When possible we link them to one of the existing names (with some exceptions for rare characters which were not named). In total we label 45,325 name mentions and annotate 17,839 pronouns, see Table 8.1. With this information, we create our corpus **MPII-MD Co-ref+Gender** where we transform the original descriptions so

	Names	Pronouns	All Mentions	Boxes
Training	37,432	15,093	52,525	489
Validation	3,440	1,092	4,532	412
Test	4,453	1,654	6,107	1,748
Total	45,325	17,839	63,164	2,649

Table 8.1: Left: number of annotated mentions, right: number of named bounding boxes, on MPII-MD.

that every character mention, which appears in a previous sentence, is replaced with "MaleCoref"/"FemaleCoref", otherwise with "MaleName"/"FemaleName". We emphasize that this is the only difference to the standard "Someone" MPII-MD, and there are no other differences between the datasets, i.e. the video-clips and splits are identical.

Grounded character annotations. To evaluate the correctness of character grounding we annotate some characters with bounding boxes in video frames. For a subset of movies from MPII-MD Training, Validation and Test set we randomly select some sentences and annotate all the mentioned characters. More specifically, whenever the character is mentioned in the sentence and visible in the corresponding clip we annotate a few frames of the clip with his/her head bounding boxes. In the final evaluation we also want to check the co-reference correctness, i.e. the link between the character track in current and previous clips. Thus we include pairs of consecutive sentences/clips from the Test set in our annotations. In total we label 2,649 bounding boxes with names, see Table 8.1.

8.4 VISUAL REPRESENTATIONS FOR CHARACTERS AND THEIR CONTEXT

In this section our goal is to localize individual characters in video and extract visual representations informative of their appearance and context. Towards this goal we first detect, track, and extract localized representations for individual characters (Section 8.4.1), and then extract global representations which capture the scene and context not captured in localized character representations (Section 8.4.2).

8.4.1 Character tracks and representations

To localize the characters in movies we focus on localizing the heads as most of the time the head of a character is shown, but frequently not the full body. In contrast to prior work e.g. (Ramanathan *et al.*, 2014) we do not focus on frontal faces only but also allow for more challenging views, like back view. We first detect the heads (Section 8.4.1.1) and then track them with a two-step clustering approach which

is able to track across shot boundaries (Section 8.4.1.2). We extract several visual representations based on the tracks which allow us to estimate character's identity, activity, gender, and their importance to be described (Section 8.4.1.3).

8.4.1.1 *Head detection*

We first detect all person instances in our videos using a head detector. Unlike conventional face detectors, our head detector can reliably detect profile faces and even back view heads. This is desirable because movies contain a large variety of view angles on heads. Our detector is based on the Faster R-CNN (Girshick, 2015b), a state-of-the-art object detection framework. For training our head detector we collect head detection bounding box annotations over the PASCAL VOC 2010 trainval set. The dataset consists of 10,103 images of 7,372 head instances. 6,659 images do not have people, but we retain them as source of negatives. We make two modifications to the original Faster R-CNN configuration (tuned for 20 PASCAL object category detection) to make it more suitable for our head detection task. First, we account for small heads by adding smaller scale "anchor boxes". Anchor boxes refer to the default set of sliding window proposals from which Faster R-CNN regresses detection bounding boxes. Second, instead of doing hard negative mining by considering only proposals with ground truth overlap > 0 and ≤ 0.5 as negative samples, we include any proposal with overlap ≤ 0.5 as negatives. This greatly improves the quality of our head detector by increasing the diversity of negative head training samples. We run our detector on every frame of MPII-MD. We keep all the head detections with scores ≥ 0.5 and both dimensions ≥ 40 pixels.

8.4.1.2 *Head tracking*

After obtaining the head detections we aim to track them within the video clip. More specifically, we want to group all detections corresponding to the same person together. We need to take into account that the movies have shot boundaries (rapid changes in a camera viewpoint/angle). Thus the motion of a person can not be the only cue for tracking and we require an appearance cue to group together different views of the same character. This motivates our two-step approach, where we first group head detections within individual shots based on their motion and then further group the obtained tracks based on their appearance.

We obtain shot boundaries with a shot boundary classifier: To determine whether a there is a boundary between two frames we rely on two features. First, we obtain the color histograms on both frames and compute the Manhattan distance between the two. Second, we run a standard Matlab interest point tracking, namely, the Kanade-Lucas-Tomasi (KLT) point tracker (Lucas and Kanade, 1981; Tomasi and Kanade, 1991), initialized in the first frame with corner points from the minimum eigenvalue algorithm. We compute the ratio of points that are reliably tracked in the second frame. Based on these two characteristics we estimate the thresholds which allow us to detect shot boundaries and achieve high recall on a small set of manually annotated frame pairs w.r.t. to being a shot boundary. We select the parameters on a

set of annotated frames and get the F-score 0.98. We try to detect all boundaries if possible and not produce too many false positives (wrong boundaries). Our tracking approach can deal with some false positives by clustering different tracks together based on appearance.

Our tracking framework is based on work of Tang *et al.* (2015), a multicut (Chopra and Rao, 1993; Grötschel and Wakabayashi, 1989) tracker for pedestrians in street scene videos. The idea is to build a graph based on the person detections in the video, and then obtain the tracks by partitioning the graph into an optimal number of connected components, based on attractive and repulsive pairwise terms between pairs of detections. It is essentially a clustering based tracking formulation, which produces robust tracking result. In our work, we adapt the multicut tracker to generate tracks for person heads in video clips. We cast our tracking task as a two-level clustering problem. On the first level, we generate tracks from detections that are obtained on the consecutive frames within individual shots. To generate tracks from detections, we employ simple geometric features between detection bounding boxes. Specifically, given two detection bounding boxes b and b', each has spatial-temporal location (x, y, t) , scale h and a corresponding image region B. We define the following variables $\bar{h} = \frac{(h_b + h_{b'})}{2}$, $\Delta x = \frac{|x_b - x_{b'}|}{\bar{h}}$, $\Delta y = \frac{|y_b - y_{b'}|}{\bar{h}}$, $\Delta h = \frac{|h_b - h_{b'}|}{\bar{h}}$, $IOU = \frac{|B_d \cap B_{d'}|}{|B_d \cup B_{d'}|}$, where IOU is the intersection over union of the two detection bounding boxes. The pairwise feature is defined as $(\Delta x, \Delta y, \Delta h, IOU)$. Additionally, we add the quadratic terms of each feature to form a non linear mapping from feature space to the pairwise potentials.

On the second level, we cluster tracks that are obtained from the first level that are at least 5 frames long for computational efficiency. For the second level we rely on the visual appearance features. More specifically, for each track we mean pool the FaceVGG (Parkhi *et al.*, 2015b) fc7 features on the head crops. We then compute the *cosine* distance between each pair of tracks and use $1 - cosine$ as pairwise potentials in the second clustering step.

8.4.1.3 *Track representations*

As mentioned earlier we need the representations extracted from the tracks to allow us to (re-)identify the characters, predict their activity and gender, and estimate if they are worth describing.

For re-identification of characters we again rely on the FaceVGG (Parkhi *et al.*, 2015b) fc7 representation, referred to as v^{head} in the following. We mean pool the track representation over all head crops clustered in this track and refer to this as $v^{head}(t)$ of track t. We discuss in Section 8.5 how we estimate the similarity of two tracks for character re-identification in our pipeline. We include the person body context which could be useful to e.g. predict the person's activity. We extract the body region w.r.t. the head bounding box: 3 times wider and 6 times taller. We experiment with two visual features on the body region. First is a VGG (Simonyan and Zisserman, 2015) representation (fc7) fine-tuned on the 393 activities from the MPII human pose activity dataset (Pishchulin *et al.*, 2014) using the model provided

by Gkioxari *et al.* (2015). We only use the body crop ignoring the additional context features as they would be similar across tracks and thus likely not help too much to distinguish tracks, but would significantly increase computation. Another feature we compute is ResNet (He *et al.*, 2016) (pool5), trained on ImageNet (Deng *et al.*, 2009) for object classification. We mean pool both visual representations over all body crops in a track and refer to this as $v^{body}(t)$. In the experiments we specify if/which feature is being used. We find, as also noted by Parkhi *et al.* (2015a), that the described characters are frequently in the front, center, and large compared to characters not described (background characters). Rather than manually defining a good function we provide the following track statistics $v^{stat}(t)$ and allow our approach to learn from this data: track length, mean and standard deviation of head width/height/center/detection score.

We do not extract designated gender features, as we find that v^{head} and v^{body} carry strong information about this aspect. It is straightforward to include even more targeted representation as part of future work. All the computed representations are normalized element-wise by first mean centering and then dividing by the standard deviation to improve learning subsequent functions with deep learning.

8.4.2 Holistic video representations

In the previous section we discussed how and which localized features we extract for characters. To additionally capture context, objects, and scene information which are all important to describe movies, we additionally rely on global representations described in Chapter 5. We shortly review them in the following: 1) scores from 146 activity classifiers trained with Dense Trajectory features (Wang and Schmid, 2013); 2) scores from 99 object classifiers trained with LSDA (Hoffman *et al.*, 2014) responses; 3) scores from 18 scene classifiers trained with PLACES-CNN (Zhou *et al.*, 2014) responses. All the classifiers were trained using the words from descriptions as labels. The provided visual feature v^{global} is a 263 dimensional concatenation of all three groups of scores.

8.5 GENERATING GROUNDED AND CO-REFERENCED DESCRIPTIONS

As discussed in the introduction we focus on local character grounding and co-reference resolution, while generating the description. More specifically we aim to predict the character grounding and do co-reference resolution given the previous sentence grounding. At test time this allows to e.g. process the movie sequentially from start to end. In the following we rely on our transformed description corpus, **MPII-MD Co-ref+Gender**, as described in Section 8.3.

The key ideas of our approach are to predict grounding and co-reference resolution *jointly* while generating the sentence (Section 8.5.1) and to learn grounding and co-reference with noisy supervision at training time obtained automatically by linking character mentions and tracks (Section 8.5.2). Figure 8.2 provides an

overview of our model.

8.5.1 Predicting grounding and co-reference during sentence generation

For generating sentences we rely on the recurrent LSTM (Hochreiter and Schmidhuber, 1997) network as defined in (Zaremba and Sutskever, 2014). To predict the hidden state at step τ of the sentence, we provide it with the previous word $w_{\tau-1}$ and hidden state $h_{\tau-1}$, as well as the current visual representation v_τ: $h_\tau = f^{LSTM}([w_{\tau-1}v_\tau], h_{\tau-1})$ where [.] denotes concatenation. The f^{LSTM} has an additional hidden state or memory cell c_t which is not exposed. The word is then predicted as $w_\tau = f^{pred}(h_\tau) = Softmax(W^{pred}h_\tau + b^{pred})$ which can be supervised with the ground truth word $\hat{w}_\tau$. Note that our vocabulary $w \in V$ does not contain any character names, but only $V^{person} = \{MaleCoref, FemaleCoref, MaleName, FemaleName\} \subset V$.

In the following we discuss how we obtain a v_τ which allows to predict the correct word and at the same time solve the grounding and co-reference problem. We formulate the problem in terms of tracks which are the result of the head tracking in Section 8.4.1.2. We have tracks $t_c \in T^c$ in the **c**urrent clip ($C = |T^c|$), and tracks $t_p \in T^p$ in the **p**revious clip ($P = |T^p|$). We always assume the sentences in the previous clip are already grounded to tracks and only consider those tracks which correspond to mentions of characters in the sentence. Whenever we generate a word w_τ which refers to a person $w_\tau \in V^{person}$, the task is to also select which track $t_{\hat{c}}$ it corresponds to in the current clip and which track $t_{\hat{p}}$ in the previous clip. To account for the case when the person was not mentioned in the previous sentence we include t_0 in T^p which represents a null track, which has to be selected to indicate we describe a "new" name. As we are modeling only two consecutive clips at a time, this means if $t_{\hat{p}} = t_0$ we want to generate *MaleName* or *FemaleName* and *MaleCoref* or *FemaleCoref* otherwise.

Track re-identification for visual co-reference. To estimate similarity of two tracks t_p and t_c we learn a weighting after element-wise multiplication[1]:

$$v^{id}(t_p, t_c) = v^{head}(t_p) \odot v^{head}(t_c) \tag{8.1}$$

$$f^{id}(t_p, t_c) = W^{id}v^{id}(t_p, t_c) \tag{8.2}$$

For $p = 0$, which indicates no similar track exists, we set $v^{id}(t_0, t_c) = -1$. In preliminary experiments we found that this works better than 0, as values v^{id} are close to 0.

Learning grounding and co-reference jointly. The goal of our approach is to select a track $t_{\hat{c}}$ and the corresponding previous track $t_{\hat{p}}$ which matches the person

[1] A note to our notation: We use superscript for names of variables and functions and subscript for indexes. W is consistently used to represent learned multiplicative weights and b to represent additive bias weights.

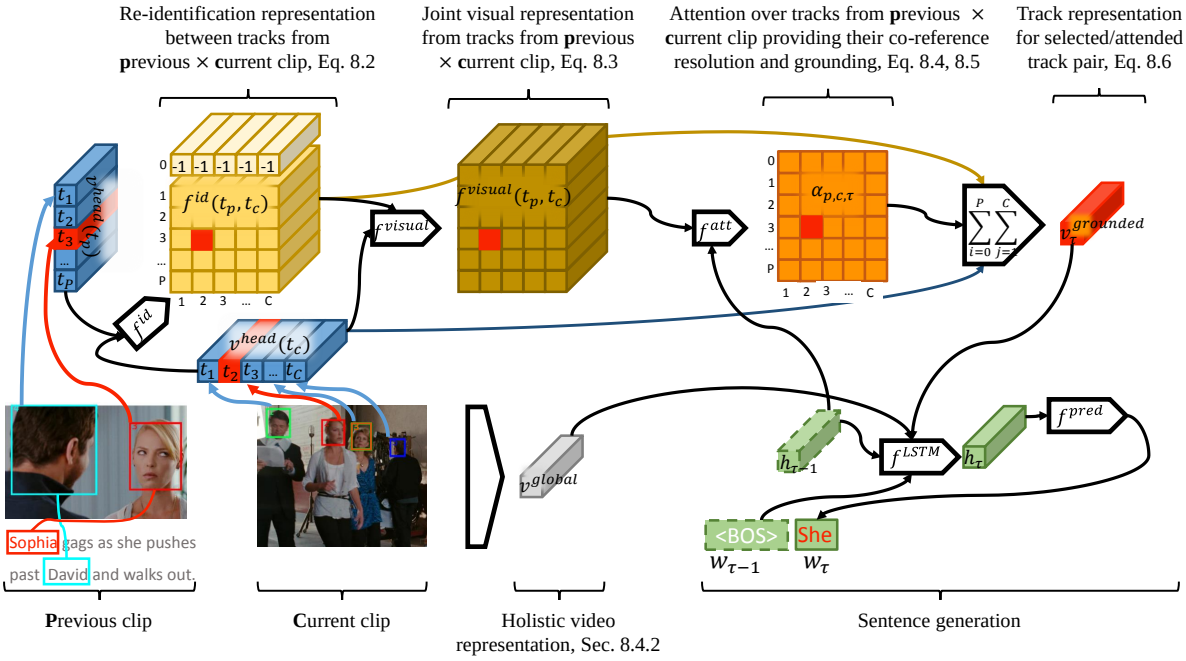


Figure 8.2: Overview of our model. Some components are omitted for clarity, e.g. we omit the body and statistic representations.

we are describing with the current word at time τ, in other words we ground this person in $t_{\hat{c}}$ and link it to $t_{\hat{p}}$. As noted above if $t_{\hat{p}} = t_0$ there is no previous track with the same identity as $t_{\hat{c}}$. We propose to jointly predict $\hat{c}$ and $\hat{p}$ using an attention mechanism which takes into account the re-identification and visual representations as well as the hidden state $h_{\tau-1}$ of the recurrent LSTM network generating the description.

The visual features are jointly embedded in the same space as the embedding learned for the hidden state:

$$f^{visual}(t_p, t_c) = W^{head} v^{head}(t_c) + W^{body} v^{body}(t_c) + W^{stat} v^{stat}(t_c) + f^{id}(t_p, t_c) + b^v \tag{8.3}$$

Afterwards visual and hidden state representation are element-wise multiplied and we learn a function to predict the attention α. This is inspired by Xu *et al.* (2015a), who combine convolutional visual features and the recurrent hidden state in the same way to predict spatial attention. Conceptually different, we predict two aspects jointly, the grounding t_p and linking t_c of tracks from different clips.

$$\bar{\alpha}_{p,c,\tau} = f^{att}(t_p, t_c, \tau) = W^{\alpha} \phi(W^h h_{\tau-1} + b^h) \odot \phi(f^{visual}(t_p, t_c)) + b^{\alpha} \tag{8.4}$$

with the htan non-linearity $\phi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$. The attention is normalized with softmax and then we use the predicted α in a weighted sum to get the new local visual

representation:

$$\alpha_{p,c,\tau} = \frac{\exp(\bar{\alpha}_{p,c,\tau})}{\sum_{i=0}^{P}\sum_{j=1}^{C}\exp(\bar{\alpha}_{i,k,\tau})} \tag{8.5}$$

$$v_{\tau}^{grounded} = \sum_{i=0}^{P}\sum_{j=1}^{C}\alpha_{p,c,\tau}[v^{head}(t_c), v^{body}(t_c), v^{stat}(t_c)v^{id}(t_p, t_c)], \tag{8.6}$$

where [.] denotes concatenation. We use this together with the global/holistic video representation v^{global} (see Section 8.4.2) and the previous word $w_{\tau-1}$ to predict the next hidden state of the recurrent LSTM network as discussed above: $h_{\tau} = f^{LSTM}([v^{grounded}, v^{global}, w_{\tau-1}], h_{\tau-1})$.

Supervising grounding and co-reference. While this system can be trained by only providing reference sentences as supervision, it is difficult to jointly correctly learn the grounding and co-reference resolution. We thus discuss in the next section how to obtain supervision for $\alpha_{p,c,\tau}$. Instead of annotating all characters mentions with tracks, we try to automatically predict the correct track t for each character mentions w_{τ} in the sentence. As we have ground truth co-reference on the text side for the entire training data (Section 8.3), we can construct the joint ground truth $\hat{\alpha}_{p,c,\tau}$ from the groundings per clip $\hat{\alpha}_{p,\tau}$, $\hat{\alpha}_{c,\tau}$. For all non-character words $w_{\tau} \notin V^{person}$, no supervision and thus no loss is provided. The losses from sentence supervision and grounding/co-reference supervision are weighted equally.

8.5.2 Obtaining automatic supervision: linking character mentions and tracks

In this section we discuss how to ground or link character mention with id m_{τ} in text at position τ to a corresponding visual track t_c in the video to provide ground truth $\hat{\alpha}_{c,\tau}$ used above. In contrast to sentence generation here we explicitly use the character mentions m (e.g. "Harry") which appear in the text. In other words we want to robustly choose the correct track for all character mentions. Note, that this is a different from e.g. Parkhi *et al.* (2015a), who aim to link all the visual tracks to correct names. To link the name mentions in text to tracks we adapt our approach GroundeR (Chapter 6). This approach was initially proposed for the task of localizing text phrases within an image without localization supervision, i.e. where the phrase is located. The main idea is to learn to *attend to* the right bounding box out of a set of proposals, by trying to reconstruct the phrase. We adapt this to our scenario by learning to localize a character $m_{\tau,k}$ in the set of tracks T_k from clip k, where character m is mentioned in the sentence k at position τ. We represent tracks with $v^{head}(t_{c,k})$ and encode character names m together with an identifier of the $gender(m) \in \{M, F\}$ as separate word in an LSTM. Adding the gender allows the model to exploit correlations with different visual appearance of male versus female people and thus simplifies selecting the right track. In the special case when the sentence k only contains a single name and the clip k contains a single track, i.e. $|T_k| = 1$ we assume that grounding is correct and this information is used as

additional supervision, thus enabling the semi-supervised setting of GroundeR. To train the model we use pairs $([gender(m_{\tau,k}), m_{\tau,k}], \{v^{head}(t_{c,k})\}_{c\in\{1..C\}})$ and predict the grounding as the track with maximum attention from the all tracks in the clip.

8.6 EVALUATION

We start with evaluating the quality of our person head detection and tracking. Then we look at the quality of automatic linking between character names and tracks, obtained in Section 8.5.2. Finally, we evaluate our complete pipeline for grounded movie description. We break down the evaluation in two parts: description quality and grounding quality.

8.6.1 Head detection and tracking

We evaluate our head detections and tracks on the collected bounding box annotations from Section 8.3. Given the annotated bounding boxes we compute detection recall by looking whether there is a head detection in a given frame that overlaps by IOU ≥ 0.5 (Intersection Over Union) with the annotated head box. The track recall is computed similarly, based on the presence of the track that goes through the given frame while overlapping with the annotated box by IOU ≥ 0.5. Table 8.2(a) shows recall on the Training, Validation and Test parts of the annotations.

We analyze the missing recall of our head detector on the Training annotations. We find that there are multiple failure modes, such as motion blur, occlusion and head size (both small and large) contributing to the missing recall. On the well visible heads we achieve 93.2% recall. The tracking recall is slightly lower than the detection recall, due to the short track rejection (see Section 8.4.1.2). In particular tracking can be hard when the head is observed from an unusual angle. Overall, we find that our annotations are rather challenging and the obtained performance is reasonable. We also note that our approach works already works with one good track for each character.

8.6.2 Linking characters with tracks

For every clip we restrict the number of tracks to at most 50, if more than 50 tracks are available we sort them by length and keep the longest, otherwise we zero-complete the missing tracks. For the previous track we consider at most 7 candidate tracks in addition to the "null" track (no match among the previous tracks). Thus there are 8×50 possible choices to predict the character grounding and co-reference during sentence generation. We first train the GroundeR (Chapter 6) approach on Training movies only in order to estimate the hyper parameters. Next we combine the Training, Validation and Test movies and train GroundeR on this joint set. We evaluate the accuracy of the obtained predictions on the annotated pairs name/bounding box presented in Section 8.3. For a given name we choose the

Recall	Training	Validation	Test
Detection	82.00	65.78	84.73
Tracking	78.53	61.65	81.41

(a)

Accuracy	Training	Validation	Test
GroundeR	78.12	84.46	80.35

(b)

Table 8.2: (a) Detection and tracking recall on the annotated character heads. (b) GroundeR accuracy on the annotated names/bounding boxes (evaluated on the boxes covered by the tracks). In %.

top scoring track as the grounding prediction. For this track we then check whether it contains the annotated frame and overlaps with the annotated box by IOU ≥ 0.5. Table 8.2(b) shows that GroundeR is able to quite robustly predict the correct track for a given character name.

8.6.3 Evaluating description quality

We evaluate our approach in terms of description quality and compare it to a few baselines as well as prior work via an automatic as well as human evaluation. We report all the standard automatic measures in Table 8.3. For human evaluation the human judges were provided with pairs of a reference sentence and a predicted sentence, and asked to compare them w.r.t. being helpful for a blind person to follow the events in the video. The judges can decide that one sentence is better than the other or both are similar. Each pair is evaluated by three human judges. Afterwards for every system we compute the percentage of times when at least 2 out of 3 judges decided that the predicted sentence is similar or better than the reference. Table 8.3 presents the results of human evaluation in the last column.

The top part of the table contains the reference numbers from prior works on the standard version of the corpus. We cannot use attention supervision or evaluate grounding on standard MPII-MD, which are our core contributions. It is encouraging that our reduced model "Our w/o α" achieves similar scores to prior work.

The middle and bottom part of the table presents results on MPII-MD Co-ref+Gender, thus the numbers between the two settings are not directly comparable as the references changed which strongly affect the automatic evaluation measures. To address this we evaluate our approach Visual-Labels (Chapter 5), on the transformed corpus. Unlike Chapter 5, here we do not ensemble multiple models. For a fair comparison with the Visual-Labels in the middle part of Table 8.3, we provide ablations that do not have access to the previous clip character grounding but instead select the 7 biggest previous tracks if sorted by track length multiplied by an average track area. We compare a variant of our approach without the body context features ("Our"), one with body features ("Our + Activity") as described in Section 8.4.1.3, and one which removes the attention mechanism but uses the activity feature and encodes it jointly with the holistic feature ("Our + Activity w/o attention & co-reference"). In the bottom part of Table 8.3 we use the automatically

Approach	Automatic Bleu-4	Metor	Rouge	CIDEr	Human judgment
Standard MPII-MD with "Someone"					
SMT-Best (Chapter 5)	0.47	5.59	13.21	8.14	-
Visual-Labels (Chapter 5)	0.80	7.03	16.02	9.98	-
S2VT (Venugopalan *et al.*, 2015a)	0.64	7.10	15.69	6.96	-
Our w/o $\hat{\alpha}$	0.84	6.43	16.10	10.66	-
MPII-MD Co-ref+Gender					
without previous clip character grounding					
Visual-Labels (no ensemble)	0.66	5.21	13.94	10.34	11.8
Our + Act. w/o att.&co-ref.	0.74	5.58	14.49	10.22	11.0
Our	0.67	5.06	13.17	10.89	14.8
Our + Activity	0.71	5.31	14.14	11.33	15.0
with previous clip character grounding					
Our w/o $\hat{\alpha}$	0.66	5.82	14.29	10.48	10.8
Our w/o statistic features	0.75	5.81	14.97	11.65	-
Our	0.68	5.81	15.33	11.70	14.0
Our + Activity	0.82	6.17	16.12	12.64	14.5
Our + ResNet	0.88	6.00	15.70	11.76	13.0

Table 8.3: Left: automatic / right: human evaluation of description generation on the test set of MPII-MD; for discussion see Section 8.6.3.

obtained previous clip grounding (via Section 8.5.2, which has access to the previous ground-truth sentence), so that different variants of our approach are comparable, as they obtain the same previous information. Here we compare "Our" and two variants of our approach with body features ("Our+Activity", "Our+ResNet"). We also ablate the impact of the grounding and co-reference supervision ("Our w/o $\hat{\alpha}$") and the statistic features ("Our w/o statistic features").

From Table 8.3 we see that: a) the ablation systems "Our" / "Our + Activity" (without previous clip character grounding) achieve similar or better sentence quality than the Visual-Labels baseline; b) the variant with extra body context but without attention mechanism gets lower human score than our full system (11.0 vs. 15.0); c) providing grounding and co-reference supervision $\hat{\alpha}$ benefits the sentence quality; d) overall, body context features benefit the scores, while the statistic features do not make a significant impact. The best result, according to human evaluation, is achieved by the variant of our approach "Our + Activity" *without previous clip grounding*, significantly improving over the Visual-Labels baseline. A possible explanation for this is as follows. For the automatically obtained previous clip's character grounding we might: a) link correctly; b) link the characters to tracks incorrectly; c) miss some links if names are absent. In a) we follow the storyline of the movie. If we instead use the largest tracks in the previous clip, we bias the description of the current clip in a different way, e.g. focus on the most salient characters. Thus, in some cases the obtained descriptions are ranked higher by the humans, as they only

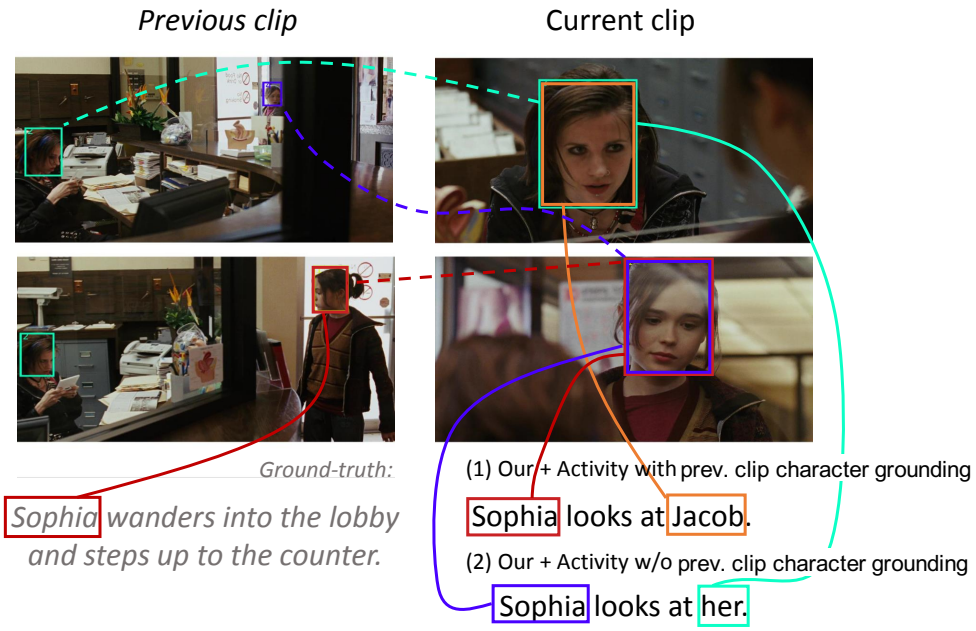


Figure 8.3: Supported by a visual co-reference to the previous clip, (2) correctly refers to a receptionist as *'her'*, rather than *'Jacob'*(1).

see the current clip in isolation (no story-line). In b), c) it is naturally more difficult to obtain a correct description of the current clip. See Figure 8.3 for an example.

8.6.4 Evaluating grounding quality

In this section we evaluate the correctness of the predicted grounding, co-reference and the generated character specific word $w_\tau \in \{$*MaleCoref, FemaleCoref, MaleName, FemaleName*$\}$. We evaluate our predictions with respect to the manually obtained ground-truth (Section 8.3) or automatically obtained ground-truth (Section 8.5.2). For each of the named bounding boxes we obtain the track which overlaps with it most, for every character mention we obtain one or more associated ground-truth tracks. In total we obtain a set of 186 sentences with manually obtained grounding and co-reference. For the automatic annotations we evaluate on a complete MPII-MD Test set (6,578 sentences).

We break down the evaluation in three parts: *Grounding*, *Grounding + Co-Reference*, *Grounding + Co-Reference +* w_τ (generated word). We compute precision and recall for each of these tasks and report the $F1$ score. Precision is computed as a percentage of predictions $\{\alpha_{p,c,\tau}, w_\tau\}$, $w_\tau \in \{$*MaleCoref, FemaleCoref, MaleName, FemaleName*$\}$, which are present in ground-truth. For the *grounding* task we only check whether the track t_c is present among ground-truth tracks. For *co-reference* it has to be also correctly linked to the track t_p from a previous clip. For the final task the predicted word w_τ with the track t_c and predicted co-reference t_p has to be present in the ground-truth. Recall is computed in a reversed way: for every ground-truth pair $\{\hat{\alpha}_{p,c,\tau}, \hat{w}_\tau\}$ we check whether it is in the predictions.

F1 score	manual labeled subset Ground.	+Co-Ref	$+ w_\tau$	automatic gt, full set Ground.	+Co-Ref	$+ w_\tau$
Baselines with heuristic attention						
Visual-Labels Center	59.21	19.33	13.83	36.17	24.52	17.26
Visual-Labels LxA	69.58	23.93	18.80	41.62	27.58	19.82
Visual-Labels LxA,Sim	69.58	39.05	6.07	41.62	29.76	13.11
Our w/o $\hat{\alpha}$	64.60	21.75	13.47	46.19	28.88	20.41
Our w/o stat.feat.	70.77	50.34	44.57	46.34	38.14	32.87
Our	69.17	53.92	49.55	47.24	38.47	33.88
Our + Activity	71.99	50.54	45.63	53.12	42.15	37.23
Our + ResNet	69.76	51.51	46.54	54.73	43.17	37.92
GroundeR gt	89.10	84.36	84.13			

Table 8.4: Grounding evaluation on test set. For discussion see Section 8.6.4.

The top part of Table 8.4 shows a set of baselines where we aim to obtain the grounding and co-reference resolution as a post-processing step after the sentence was generated. We use Visual-Labels as a sentence generation baseline. We consider multiple heuristics to select the track: central position, length, length multiplied with the average area. Additionally we use a simple co-reference resolution method: if there are tracks in the previous clip we pick the one which is most similar to the selected track as a co-reference. The similarity is estimated as a $1 - cosine(v^{head}(t_c), v^{head}(t_p))$. The bottom part of the table lists the variants of our approach introduced earlier.

Table 8.4(left) presents the evaluation with the manually obtained ground-truth. As we can see: a) the baselines are rather competitive in the grounding task, however they fall far below our approach in the co-reference task; b) grounding and co-reference supervision $\hat{\alpha}$ is very important to learn the co-reference prediction; c) statistics features, although they did not impact the description quality significantly, benefit the co-reference resolution; d) our approach is doing quite well in the final task, meaning that the language model correctly learns when to use co-references and recognizes the gender information.

In the last line of Table 8.4 we evaluate the quality of automatic ground-truth predictions from Section 8.5.2 with respect to our tasks. As we can see the predictions are overall quite reliable. Encouraged by that we perform the evaluation on this automatic ground-truth for the complete Test set, Table 8.4(right). We note, that the manually annotated set covers only 2.8% of the full test set, so the results on the full test are more stable. We make the following observations: a) an ablation w/o statistic features again slightly drops in performance; b) all the baselines fall below our best approaches in all three tasks; this can be attributed to a more challenging data distribution: the complete test set contains sentences/clips where characters are absent and that has to be recognized correctly, while the manually annotated set always contains characters and is biased towards co-references; c) on this larger and more challenging test set we see that "Our + Activity" and "Our + ResNet"

Figure 8.4: Qualitative results of our approach on the grounded movie description task. Given a previous grounding we predict a sentence, grounding and co-reference.

benefit from additional body features and achieve better performance than the basic variant "Our"; one observation we make is that these two variants are more accurate with respect to presence/absence of people in the sentence/video which impacts the precision and thus the F1 score. In Figure 8.4 we provide some qualitative examples with the predictions from our approach.

8.7 CONCLUSIONS

In the previous chapters we separately studied the problem of video description (Chapters 4, 5) and visual grounding (Chapters 6, 7). In this chapter we look at a novel task, namely generating descriptions with joint grounding and co-reference resolution of person mentions. We propose a novel approach, which relies on an attention mechanism that jointly learns to solve the grounding and co-reference

resolution while learning to describe the video clip. By learning to automatically link names and tracks we provide supervision into our approach which significantly improves its ability to perform co-reference resolution. We demonstrate encouraging results in a complex task of grounded movie description and achieve improvements over several baselines. Our approach generates sentences of better quality than the baselines as shown by automatic and human evaluation. Overall, our approach can describe video, reason about persons identities, recognize their genders, and localize them in video. We believe that this work is a first step towards fully coupling generation and grounding while performing image/video description. We will release the annotations and extracted tracks and hope that this will benefit other researchers who work on linguistic and/or visual co-reference resolution, movie question answering, visual storytelling, and multi-sentence video description.

9 CONCLUSIONS AND FUTURE PERSPECTIVES

Language description of visual content is a long standing problem in computer vision which received significant attention only recently. In the last few years image captioning and video description became standard tasks, and generally, *Language and Vision* became one of the most exciting and fast evolving research directions. This creates a lot of opportunities for collaboration and knowledge exchange between computer vision and computational linguistics. Indeed, many ideas from language processing have been applied to computer vision problems (Duygulu *et al.*, 2002; Hofmann, 2001; Sivic *et al.*, 2005; Vinyals *et al.*, 2015) and vice versa (Barnard and Johnson, 2005; Johnson and Zhang, 2015; Pu *et al.*, 2007). While the early works have only studied limited interactions between language and vision (Barbu *et al.*, 2012; Kulkarni *et al.*, 2011), now, the increasingly popular deep learning techniques allow us to bridge the gap between different modalities more easily (Karpathy and Fei-Fei, 2015; Kiros *et al.*, 2015a; Tapaswi *et al.*, 2016). At the same time, new tasks have emerged and quickly became popular, such as visual question answering (Antol *et al.*, 2015; Malinowski and Fritz, 2014) and localization of textual phrases or referring expressions (Mao *et al.*, 2016; Plummer *et al.*, 2015). Although most research focuses on the image domain, recent deep learning techniques and new parallel corpora of video and sentences allow us to study video-language interactions more closely (Maharaj *et al.*, 2017; Yao *et al.*, 2016).

In this thesis we have looked at three research directions, which we shortly summarize in the following. In the first direction, *video description of fine-grained cooking activities*, we focused on long cooking video understanding and description. We proposed approaches to recognize fine-grained and composite cooking activities, and describe them in a coherent way at variable level of detail. We also contributed with a parallel dataset of videos and multi-level descriptions to study the aforementioned problem. In the second direction, *large-scale movie description*, we moved to the large-scale open movie domain. We contributed with a new large dataset of movies with aligned professional descriptions, and proposed an approach to provide reliable descriptions for short movie clips. In the third direction, *language grounding and grounded video description*, we addressed the visual grounding problem. We showed how to approach textual phrase localization in images with little or no localization supervision. Furthermore, we proposed an approach to video description with grounded and co-referenced people. In all three directions we have advanced the state-of-the-art on multiple challenging benchmarks.

In this chapter we discuss the contributions of the thesis in more detail (Section 9.1) and then review open issues and possible future research directions (Section 9.2).

9.1 DISCUSSION OF CONTRIBUTIONS

In this thesis we have explored two high-level topics: automatic natural language video description and grounding of natural language in visual data. We tackled three specific sub-topics, namely *video description of fine-grained cooking activities*, *large-scale movie description* and *language grounding and grounded video description*, as introduced earlier. We now discuss the main results and insights that this thesis contributes.

9.1.1 Video description of fine-grained cooking activities

The first two chapters of the thesis are devoted to understanding and describing fine-grained cooking activities. We introduced our MPII Cooking 2 dataset in Chapter 3. The dataset includes 273 videos of 30 different human subjects performing various cooking activities. We addressed both, the fine-grained activity (small steps) classification and detection, as well as composite activity (high-level topic) classification. We showed that our hand-centric approach, namely detecting hands of the subject and extracting visual representations around them, benefits fine-grained activity and object recognition. We found that Dense Trajectory features extracted around hands outperform the holistic Dense Trajectories. When combined with additional hand-centric color Sift features we further improved the recognition of manipulated objects. Towards composite activity recognition, we took the attribute-based approach, namely we modeled the composites with the smaller steps that they include (fine-grained activities) as well as participating objects. We also showed how to improve the recognition and even perform zero-shot recognition of composite activities by exploiting linguistic data (scripts).

In Chapter 4 we proceeded to automatically describing the cooking videos in the MPII Cooking 2 dataset. To train and evaluate our models we collected sentence descriptions for each video at three levels of detail, resulting in our TACoS Multi-Level dataset. As our videos are long, each depicting a certain composite activity (i.e. dish preparation), it is natural to generate multi-sentence descriptions for them. To that end we proposed an automatic temporal segmentation approach, based on agglomerative clustering of visual attributes, and then described each segment relying on the translation approach of Rohrbach *et al.* (2013b). However this does not guarantee coherence across generated sentences. We showed that modeling the composite activity (high level topic shared by all segments) in our probabilistic formulation improves consistency of the descriptions. We additionally post-processed the generated descriptions to increase their fluency and cohesiveness. We also improved the visual recognition in two ways. First, we trained semantic role-aware classifiers for visual attributes to distinguish e.g. if a knife is used as a tool ("he cuts a carrot *with* the knife") or an object ("he rinses the knife"). Second, we exploited the hand centric approach to fine-grained activity and object recognition, described above. This lead to substantial improvement in recognizing manipulated objects. One the language generation side, we used a probabilistic input to the

translation system in a form of a word lattice and showed its benefits over the single best visual prediction as done by Rohrbach *et al.* (2013b). We also addressed the multi-level description generation, namely describing videos at three levels of detail. We analyzed the collected descriptions in TACoS Multi-Level corpus, and found that across different types of descriptions the language statistics changes. We showed that the high level topic of the video can guide us to extract the most informative segments that summarize the video. We applied this idea to generate short 3 sentence summaries by extracting the respective sentences from the detailed descriptions. To obtain the single sentence descriptions, however, we showed that it is necessary to learn a targeted translation model to capture the language statistics.

9.1.2 Large-scale movie description

Chapter 5 of the thesis focused on a challenging problem of movie description. A central contribution is a new large-scale dataset of movie video with manually aligned professional descriptions, the Large Scale Movie Description Challenge (LSMDC), based on the MPII Movie Description (MPII-MD) and M-VAD datasets. We sourced the descriptions from movie scripts and Audio Descriptions (AD) for the visually impaired. Our dataset comprises 200 movies of diverse genres and has over 150 hours of video. We showed that AD is a better language resource than scripts to study the movie description task. Over 200 research groups from all over the world have requested access to our dataset. We have organized two editions of the Large Scale Movie Description and Understanding Challenge (LSMDC)[2] to maintain progress in video understanding research, LSMDC15 at ICCV 2015 and LSMDC16 at ECCV 2016. The evaluation protocol for LSMDC15 involved all the standard automatic measures and human evaluation. Humans were asked to rank the provided descriptions w.r.t. their correctness, grammar, and relevance to the video. We also proposed to evaluate according to the new criteria, "Helpful for the blind", namely asking which description is more helpful for a blind person to follow the movie. We changed the evaluation protocol in LSMDC16, converging on the "Helpful for the blind" criteria and switching from ranking to a pairwise comparison of evaluated systems and human references, similar to "M1" metric in the MS COCO Challenge (Chen *et al.*, 2015), to allow comparison of a higher number of systems. We found that all the competing approaches, except the retrieval-based, struggle to capture the long-tail distribution of the LSMDC dataset.

We also presented a movie description approach, *Visual-Labels*, which relies on visual classifiers, and, similar to Donahue *et al.* (2015), provides the classifier scores as input to an LSTM. We used our semantic parser to extract the most visual semantic concepts from the sentence descriptions, and trained the respective visual classifiers. We showed that maintaining the most reliable classifiers and training different semantic concepts disjointly benefits the performance. We also showed the benefits of applying dropout after the LSTM module in the generation pipeline, the importance of selecting training iteration w.r.t. the linguistic evaluation metric, and the advantage of using LSTM ensembles. *Visual-Labels* obtained state-of-the-art

performance on the MPII-MD and M-VAD datasets according to automatic and human evaluation. We analyzed the challenges in the movie description task using our and two other approaches. We found that the factors which contribute to higher performance include presence of frequent words, sentence length and simplicity as well as presence of "visual" verbs (e.g. "nod", "walk", "sit", "smile"). We also found a high bias in the data towards humans as subjects and verbs like "look", "stare", etc. All the compared approaches suffer from the issue of a low vocabulary size, compared to reference descriptions.

Beyond our study on single sentences, the LSMDC opens new possibilities to understand stories and plots across multiple sentences in an open domain scenario on large scale.

9.1.3 Language grounding and grounded video description

The remaining three chapters of the thesis addressed different aspects of the visual grounding problem. In Chapter 6 we tackled the task of textual phrase localization in images. We proposed GroundeR (GROUNDing by Reconstruction), an approach which jointly learns visual and language representations in one end-to-end architecture. We addressed the challenge of limited localization supervision (phrases paired with bounding boxes) by introducing the Reconstruction loss. This loss penalizes incorrectly attended image regions if the phrase generated from this region does not match the query phrase. This enabled our approach to work in all supervision regimes: with no, little, or full supervision. In order to handle the complexity of natural language queries we suggested to encode them with an LSTM network and learned a phrase representation end-to-end, jointly with the representations for individual words. We showed experimentally that with little available localization supervision we can already achieve state-of-the-art results, and our fully supervised version significantly improved over the state-of-the-art on two datasets. We showed the advantage of the Reconstruction loss also compared to a fully supervised version, which only relied on a classification objective.

Chapter 7 studied how to combine the language and vision representation in a neural architecture, an important aspect relevant to many language and vision problems. We introduced the Multimodal Compact Bilinear Pooling (MCB), which approximates the outer product between the two vector representations. We showed two application scenarios, visual question answering (VQA) and visual grounding. In both cases we improved over the state-of-the-art on multiple datasets. We extensively ablated the proposed pooling strategy, compared it to e.g. concatenation or element-wise product, and found MCB to work best. It also outperformed all other approaches by a significant margin submitted to VQA challenge, and won both the open ended and multiple choice track on real images.

In Chapter 8 of the thesis we proposed a new task of generating video descriptions with grounded and co-referenced people. We proposed a novel end-to-end approach which addresses multiple tasks jointly, namely it generates a video description, while grounding and locally co-referencing the described people, and predicting

their genders. In the core of our approach is a soft attention mechanism, which reasons across multiple visual tracks in the two neighbouring video clips. We also automatically generated a linking between names and tracks in a movie in a weakly-supervised manner, using our approach GroundeR. With this linking we provided supervision into our approach which significantly improved its ability to perform co-reference resolution. To facilitate research in this direction we annotated the name mentions, i.e. proper human names, their co-references (aliases, pronouns) and genders in the MPII-MD dataset. We also annotated a subset of names with the corresponding bounding boxes. We evaluated individual steps of our pipeline on the set of manually annotated bounding boxes with character names. We then evaluated our complete approach on a large and challenging test set of MPII-MD dataset. We compared to our approach to video description, Visual-Labels, and showed that we can achieve better performance, w.r.t. automatic and human evaluation.

9.2 FUTURE PERSPECTIVES

In this section we first outline the open issues and next steps towards video description (Section 9.2.1) and visual grounding (Section 9.2.2), and then provide a broader look on the field, including possible research directions in Section 9.2.3.

9.2.1 Video description

In the thesis we have discussed many challenges towards automatic video description and provided solutions to address them (Chapters 3, 4, 5). However, our research in this thesis points to open problems and new emerging tasks. We discuss the most prominent ones in the following.

Addressing long tail distribution. One of the main challenges of describing open domain video data, e.g. movies from our LSMDC dataset, is to address the challenging long-tail distribution. The data is rather imbalanced: some words are extremely frequent (e.g. look, turn), while others are rare (e.g. transform, boil). Most approaches struggle to output diverse descriptions with a vocabulary size compared to human references. One idea could be to decompose sentences into smaller semantic concepts, e.g. verb phrases (*"he runs"*, *"car drives"*), and try to model the semantic relationships between them. We could then transfer the knowledge from more common concepts like *"he runs"*, to less common, e.g. *"she jogs"*. Another idea is to employ the external text and visual corpora, to learn the semantics of each respective domain first. Some recent works (Hendricks *et al.*, 2016b; Venugopalan *et al.*, 2017) look into that, i.e. they study how to generate image captions for rare and even novel (unseen) concepts.

Coherent multi-sentence generation. We have addressed multi-sentence video description in a scenario of cooking videos (Chapter 4). However, describing open

domain videos, such as movies, with coherent multi-sentence descriptions remains an open problem. A few recent works approach it as they attempt to generate paragraph-long descriptions for open domain videos (Shin *et al.*, 2016; Yu *et al.*, 2016a). Multiple aspects need to be addressed to this end, e.g. a decoder which can handle temporal connectives, anaphora, etc. in the language generation pipeline. We studied co-reference resolution in Chapter 8, but we limited it to two neighbouring video clips and sentences. Next steps include to extend the time horizon to a few previous clips, which should be sufficient to decide on the usage of language pronouns (*"he"/"she"* etc). Second, we need to obtain consistent person IDs in the entire movie, rather than only locally, as we did in this thesis.

Video temporal structure. Another challenge of describing long video and/or generating multi-sentence description, is to design an encoder, which can capture the video temporal structure and exploit long- and short-range context. A few recent works propose approaches to encode the short-term temporal structure (Baraldi *et al.*, 2017; Pan *et al.*, 2016a; Peris *et al.*, 2016). Still, capturing the long-term information, such as a movie storyline, remains an open issue. The encoder should model video hierarchical structure, i.e. frames, shots, scenes and sequences, discover which ones are related and thus should be used as context to allow for longe-range reasoning.

Bringing in other modalities. One important challenge of our LSMDC dataset is to obtain not purely descriptive, but also more emotion-coloured descriptions, e.g. *"She tries to hide her excitement."* This is especially challenging as the visual representations tend to focus on objects and actions, but not on human emotions. The recent MSR Video to Language Challenge also revealed the benefit of exploiting multiple modalities, such as audio and metadata (high-level video category). Overall, audio (sound), speech (dialog), emotions, body language, and meta-data can benefit video description and should be exploited, if available. Our approach, Visual Labels, which targets three semantic categories (actions, objects, locations) could be easily extended to include representations from other categories/modalities.

Automatic AD generation. To fully address the automatic Audio Description (AD) generation one needs to tackle all the challenges listed above. Moreover, one needs to model the context, such as dialog and storyline. AD has to be carefully placed between the dialogs, also considering the music and other sounds in the video. Movie-specific information, e.g. movie synopsis could be extremely helpful and can be obtained from external online resources. Additionally, named entity recognition (e.g. people, places, objects) has to be built in the description pipeline. In this thesis we took the first step to locally co-referencing re-appearing people (Chapter 8). In future work the true names could be extracted either from dialog, or from one or a few annotations per character.

Multi-level video description. The ability to produce long and detailed descriptions or short summaries on demand is an important property of a video description

system. In Chapter 4 we introduced our approach to multi-level description of cooking videos. However, multiple issues still need to be addressed. First, we rely on a high level topic prediction to produce summaries, which is challenging in open domain videos. E.g. this requires understanding of a complex activity or event shown in a video. Second, we rely on an extractive summarization technique, while it would be more flexible to incorporate variable levels of detail into the language generation system, and *learn* to adapt to a needed level of detail.

9.2.2 Visual grounding

We have discussed different aspects of visual grounding problem in this thesis (Chapters 6, 7, 8). In the following we review the possible extensions of the proposed approaches.

Constraints within sentences at training time. In our grounding approach (Chapter 6) we apply post-processing in a form of a constraint, that multiple phrases from the same sentence should not be grounded in the same region. In situations when such a sentence context is available, one should model the constraints at training time. Moreover, sometimes the sentence contains information which is necessary to disambiguate the individual phrases. Thus grounding phrases would benefit from incorporating full sentence context.

Model relationships between objects. A number of recent works exploit context in a form of pairwise relationships between objects (Hu *et al.*, 2017; Nagaraja *et al.*, 2016; Wang *et al.*, 2016a; Yu *et al.*, 2016b), by taking into account their visual similarity or spatial relations. Future work should explore other types of relationships, i.e. going beyong pairwise and spatial relationships. Recent datasets, such as the Visual Genome (Krishna *et al.*, 2016) allow us to learn diverse inter-object relationships, that would be beneficial for visual grounding.

Compositionality. An important condition for correct understanding of complex textual phrases is an ability to capture fine details, e.g. to distinguish *"a man on a horse"* from *"a man next to a horse"*. Additionally, we should be able to generalize to previously unseen phrases and configurations, e.g. *"a person on top of a car"*. This indicates a need for compositional approaches, which would extract detailed representations of the individual phrase elements and relate them to the visual scene. A recent work of Hu *et al.* (2017) makes a first step towards this goal.

Language grounding in video. Grounding of natural language expressions in video has not yet received a lot of attention in the community. The existing approaches target restricted scenarios (Lin *et al.*, 2014a; Yu and Siskind, 2013). Grounded video description generation is still in its infancy, as we (Chapter 8) and others (Ramanishka *et al.*, 2017; Zanfir *et al.*, 2016) take the first steps to address it. In addition to the challenge of relating two modalities to each other, we need to tackle the

temporal aspect, while recognizing humans and objects in video is significantly more challenging than in static images. Our effort so far focused on humans in video, aiming to localize and locally disambiguate them. Next steps could include grounding human manipulated objects and subjects other than humans.

Grounding human interactions in video. As we performed people grounding in this thesis, naturally the next step is to model relationships between multiple people. Spatial and temporal proximity of characters in a movie can indicate interactions, which are an important aspect in describing movies. As part of the future work we plan to integrate grounded person relationships in the movie description pipeline.

9.2.3 A broader outlook

Finally, in this section we look at some recent trends in vision and language research and speculate about promising research directions.

Explainability of neural models. Despite the fact that deep neural architectures provide state-of-the-art results for many computer vision problems, inspecting and understanding their decision making process remains challenging. Some works visualize the back-propagated signal in the image for the object classification task Selvaraju *et al.* (2016); Zhang *et al.* (2016); Zhou *et al.* (2016). Others aim to provide explicit textual explanations for making object classification decisions (Hendricks *et al.*, 2016a). One recent work analyzes the decision making of a VQA system, enabling it to provide counter-examples, i.e. similar images where an answer to the same question is different Goyal *et al.* (2017). Other works aim to evaluate the correctness of latent mappings between objects and words, learned by the captioning systems (Liu *et al.*, 2017; Ramanishka *et al.*, 2017). Future work should make the decision making process of the deep networks more transparent, e.g. in context of video description, by providing grounding for the generated linguistic concepts.

Leveraging external knowledge. As the complexity of language and vision tasks increases, more and more frequently we have to reason about high level concepts and facts, which are trivial for humans, but challenging for machines. In particular, incorporating external knowledge in deep architectures is not straightforward. A few works take different steps towards this end. Wu *et al.* (2016) enhance their VQA system by enabling it to access an external knowledge base. Other works take a different approach, enabling one language and vision task supervise another, e.g. video description helps VQA (Zeng *et al.*, 2017) or VQA helps image-sentence retrieval Lin and Parikh (2016). Future works should look into providing external knowledge into a wider range of tasks, e.g. movie description, as discussed earlier.

Pragmatic description generation. Describing visual content with natural language essentially is ill-defined when no specific task or context is given. Recently, a few works proposed to study pragmatic or task-specific description generation.

One line of work looks into generating non-ambiguous captions (Andreas and Klein, 2016) or referring expressions (Luo and Shakhnarovich, 2017) for images and their regions. Other works look into generating explanations instead of captions for images, as also discussed above (Hendricks *et al.*, 2016a). In the context of this thesis, generating AD for the blind is a better defined target, than generic video description. Bringing in pragmatics could also benefit the evaluation procedure of description generation, by providing proxy tasks which can be evaluated automatically.

Limited multi-modal supervision. Most supervised deep architectures are strongly dependent on the amount of available training data. At the same time, often the available supervision is limited, in particular, when we require aligned visual and linguistic corpora. In this thesis we proposed a solution to un- and semi-supervised textual phrase localization in images (Chapter 6). Future work should be able to learn from little parallel data by introducing alternative (e.g. reconstruction) objectives and exploiting the unlabeled uni-modal data.

What is a good video representation? Evaluating video representation in terms of its ability to understand the video is not straightforward. Activity recognition is only one aspect of the video content. Video description goes beyond individual category predictions, and provides a concise textual representation of the video. At the same time, evaluating video description is tedious, as it requires human evaluation: most existing automatic evaluation measures only to some extent correlate with human judgements. Other tasks, such as video/sentence retrieval, video question answering and video fill-in-the-blank are easier to evaluate automatically. Thus, architectures which target multiple tasks jointly would be most beneficial. We hope that our LSMDC benchmark, which, in addition to a movie description track, features movie annotation and retrieval (Torabi *et al.*, 2016), and movie fill-in-the-blank (Maharaj *et al.*, 2017) track, will help to evaluate and develop better video representations.

LIST OF FIGURES

LIST OF TABLES

BIBLIOGRAPHY

J.-B. Alayrac, P. Bojanowski, N. Agrawal, I. Laptev, J. Sivic, and S. Lacoste-Julien (2016). Unsupervised learning from Narrated Instruction Videos, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on page 13.

S. Amin, M. Andriluka, M. Rohrbach, and B. Schiele (2013). Multi-view Pictorial Structures for 3D Human Pose Estimation, in *Proceedings of the British Machine Vision Conference (BMVC) 2013*. Cited on pages 32, 38, and 41.

W. Ammar, C. Dyer, and N. A. Smith (2014). Conditional random field autoencoders for unsupervised structured prediction, in *Advances in Neural Information Processing Systems (NIPS) 2014*. Cited on page 122.

P. Anderson, B. Fernando, M. Johnson, and S. Gould (2016). SPICE: Semantic Propositional Image Caption Evaluation, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on page 100.

J. Andreas and D. Klein (2016). Reasoning about pragmatics with neural listeners and speakers, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) 2016*. Cited on pages 24 and 173.

J. Andreas, M. Rohrbach, T. Darrell, and D. Klein (2016a). Learning to Compose Neural Networks for Question Answering, in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) 2016*. Cited on pages 135 and 143.

J. Andreas, M. Rohrbach, T. Darrell, and D. Klein (2016b). Neural Module Networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on page 143.

M. Andriluka, S. Roth, and B. Schiele (2009). Pictorial Structures Revisited: People Detection and Articulated Pose Estimation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009*. Cited on page 47.

M. Andriluka, S. Roth, and B. Schiele (2011). Discriminative Appearance Models for Pictorial Structures, *International Journal of Computer Vision (IJCV)*. Cited on pages 43, 45, and 46.

S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh (2015). VQA: Visual Question Answering, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on pages 7, 10, 139, 140, 143, and 165.

O. Aubert and Y. Prié (2007). Advene: an open-source framework for integrating and visualising audiovisual metadata, in *Proceedings of the ACM international conference on Multimedia (MM) 2007*. Cited on page 41.

G. Awad, J. Fiscus, M. Michel, D. Joy, W. Kraaij, A. F. Smeaton, G. Quénot, M. Eskevich, R. Aly, G. J. F. Jones, R. Ordelman, B. Huet, and M. Larson (2016). TRECVID 2016: Evaluating Video Search, Video Event Detection, Localization, and Hyperlinking, in *Proceedings of TRECVID 2016*. Cited on page 12.

M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt (2011). Sequential deep learning for human action recognition, in *Human Behavior Understanding 2011*, pp. 29–39, Springer. Cited on page 36.

D. Bahdanau, K. Cho, and Y. Bengio (2015). Neural machine translation by jointly learning to align and translate, in *Proceedings of the International Conference on Learning Representations (ICLR) 2015*. Cited on pages 121, 124, and 149.

N. Ballas, L. Yao, C. Pal, and A. Courville (2016). Delving Deeper into Convolutional Networks for Learning Video Representations, in *Proceedings of the International Conference on Learning Representations (ICLR) 2016*. Cited on page 20.

L. Baraldi, C. Grana, and R. Cucchiara (2017). Hierarchical Boundary-Aware Neural Encoder for Video Captioning, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on pages 20 and 170.

A. Barbu, A. Bridge, Z. Burchill, D. Coroian, S. Dickinson, S. Fidler, A. Michaux, S. Mussman, S. Narayanaswamy, D. Salvi, L. Schmidt, J. Shangguan, J. M. Siskind, J. Waggoner, S. Wang, J. Wei, Y. Yin, and Z. Zhang (2012). Video in sentences out, in *Proceedings of the conference on Uncertainty in Artificial Intelligence (UAI) 2012*. Cited on pages 11, 16, 17, and 165.

K. Barnard and M. Johnson (2005). Word sense disambiguation with pictures, *Artificial Intelligence*, vol. 167(1-2), pp. 13–30. Cited on page 165.

A. Barr and E. Feigenbaum (1981). *The Handbook of Artificial Intelligence, Volume 1*, William Kaufman Inc., Los Altos, CA. Cited on page 42.

S. Bergsma and D. Lin (2006). Bootstrapping path-based pronoun resolution, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2006*. Cited on page 149.

M. Blaschko, A. Vedaldi, and A. Zisserman (2010). Simultaneous object detection and ranking with weak supervision, in *Advances in Neural Information Processing Systems (NIPS) 2010*. Cited on page 121.

J. Bloem, M. Regneri, and S. Thater (2012). Robust processing of noisy web-collected data, in *KONVENS 2012*. Cited on page 43.

P. Bojanowski, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic (2013). Finding Actors and Actions in Movies, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on pages 14, 148, and 150.

P. Bojanowski, R. Lagugie, E. Grave, F. Bach, I. Laptev, J. Ponce, and C. Schmid (2015). Weakly-Supervised Alignment of Video With Text, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on page 1.

P. Bojanowski, R. Lajugie, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic (2014). Weakly Supervised Action Labeling in Videos Under Ordering Constraints, in *Proceedings of the European Conference on Computer Vision (ECCV) 2014*. Cited on pages 14 and 38.

W. Brendel and S. Todorovic (2011). Learning Spatiotemporal Graphs of Human Activities, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2011*. Cited on pages 36 and 38.

M. Bruni, T. Uricchio, L. Seidenari, and A. Del Bimbo (2016). Do Textual Descriptions Help Action Recognition?, in *Proceedings of the ACM international conference on Multimedia (MM) 2016*. Cited on page 16.

L. Campbell and A. Bobick (1995). Recognition of human body motion using phase space constraints, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 1995*. Cited on page 36.

B. Chakraborty, M. Holte, T. Moeslund, J. Gonzalez, and X. Roca (2011). A Selective Spatio-Temporal Interest Point Detector for Human Action Recognition in Complex Scenes, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2011*. Cited on page 35.

J. Chaquet, E. Carmona, and A. Fernández-Caballero (2013). A survey of video datasets for human action and activity recognition, *Computer Vision and Image Understanding*, vol. 117(6), pp. 633 – 659. Cited on page 33.

M. Charikar, K. Chen, and M. Farach-Colton (2002). Finding frequent items in data streams, in *Automata, languages and programming 2002*, pp. 693–703, Springer. Cited on pages 134 and 136.

D. Chen and W. Dolan (2011). Collecting Highly Parallel Data for Paraphrase Evaluation, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2011*. Cited on pages 11, 13, 37, 82, and 93.

X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick (2015). Microsoft COCO captions: Data collection and evaluation server, *arXiv preprint arXiv:1504.00325*. Cited on pages 5, 85, 110, 114, 115, and 167.

X. Chen and A. Gupta (2015). Webly supervised learning of convolutional networks, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on page 121.

X. Chen and C. L. Zitnick (2015). Mind's Eye: A Recurrent Visual Representation for Image Caption Generation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 85 and 121.

A. Cherian, J. Mairal, K. Alahari, and C. Schmid (2014). Mixing Body-Part Sequences for Human Pose Estimation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 32.

K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio (2014). On the properties of neural machine translation: Encoder-decoder approaches, in *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8) 2014*. Cited on page 19.

S. Chopra and M. Rao (1993). The partition problem, *Mathematical Programming*, vol. 59(1–3), pp. 87–115. Cited on page 153.

R. G. Cinbis, J. Verbeek, and C. Schmid (2014). Multi-fold MIL training for weakly supervised object localization, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 121.

T. Cour, C. Jordan, E. Miltsakaki, and B. Taskar (2008). Movie/Script: Alignment and Parsing of Video and Text Transcription, in *Proceedings of the European Conference on Computer Vision (ECCV) 2008*. Cited on pages 82, 84, and 87.

T. Cour, B. Sapp, C. Jordan, and B. Taskar (2009). Learning from ambiguously labeled images, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009*. Cited on pages 14, 148, and 150.

N. Dalal, B. Triggs, and C. Schmid (2006). Human Detection Using Oriented Histograms of Flow and Appearance, in *Proceedings of the European Conference on Computer Vision (ECCV) 2006*. Cited on pages 35 and 50.

DARPA (2011). *Mind's Eye*, *http://www.visint.org/*. Cited on pages 11 and 17.

P. Das, C. Xu, R. Doell, and J. Corso (2013). Thousand Frames in Just a Few Words: Lingual Description of Videos through Latent Topics and Sparse Object Stitching, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2013*. Cited on pages 12, 16, 18, 21, 22, 37, 69, 82, and 93.

G. de Melo and N. Tandon (2016). Seeing is Believing: The Quest for Multimodal Knowledge, *SIGWEB Newsl.*, (Spring), pp. 4:1–4:9. Cited on page 14.

L. Del Corro and R. Gemulla (2013). ClausIE: Clause-based Open Information Extraction, in *Proceedings of the International World Wide Web Conference (WWW) 2013*. Cited on page 96.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei (2009). ImageNet: A Large-Scale Hierarchical Image Database, in *Proceedings of the IEEE Conference on*

Computer Vision and Pattern Recognition (CVPR) 2009. Cited on pages 5, 11, 82, 88, 126, 130, 137, and 154.

J. Deng, J. Krause, A. Berg, and L. Fei-Fei (2012). Hedging Your Bets: Optimizing Accuracy-Specificity Trade-offs in Large Scale Visual Recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2012*. Cited on page 18.

J. Devlin, H. Cheng, H. Fang, S. Gupta, L. Deng, X. He, G. Zweig, and M. Mitchell (2015). Language Models for Image Captioning: The Quirks and What Works, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2015*. Cited on page 85.

S. Divvala, A. Efros, and M. Hebert (2012). How important are 'Deformable Parts' in the Deformable Parts Model?, in *Proceedings of the European Conference on Computer Vision Workshops (ECCV Workshops) 2012*. Cited on page 44.

S. Divvala, A. Farhadi, and C. Guestrin (2014). Learning everything about anything: Webly-supervised visual concept learning, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on pages 24 and 121.

P. Dogan, M. Gross, and J.-C. Bazin (2016). Label-Based Automatic Alignment of Video with Narrative Sentences, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on page 1.

J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell (2015). Long-term recurrent convolutional networks for visual recognition and description, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 19, 22, 84, 85, 98, 124, 129, 130, and 167.

J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell (2013). Decaf: A deep convolutional activation feature for generic visual recognition, in *Proceedings of the International Conference on Machine Learning (ICML) 2013*. Cited on page 133.

J. Dong, X. Li, W. Lan, Y. Huo, and C. G. Snoek (2016a). Early Embedding and Late Reranking for Video Captioning, in *Proceedings of the ACM international conference on Multimedia (MM) 2016*. Cited on page 21.

J. Dong, X. Li, and C. G. M. Snoek (2016b). Word2VisualVec: Image and Video to Sentence Matching by Visual Feature Prediction, *arxiv:1604.06838*. Cited on page 19.

O. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce (2009). Automatic annotation of human actions in video, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2009*. Cited on pages 14 and 82.

P. Duygulu, K. Barnard, N. de Freitas, and D. A. Forsyth (2002). Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary, in *Proceedings of the European Conference on Computer Vision (ECCV) 2002*. Cited on page 165.

C. Dyer, S. Muresan, and P. Resnik (2008). Generalizing Word Lattice Translation, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2008*. Cited on pages 22 and 75.

M. Elhoseiny, B. Saleh, and A. Elgammal (2013). Write a Classifier: Zero-Shot Learning Using Purely Textual Descriptions, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on pages 37 and 38.

D. Elliott and F. Keller (2013). Image Description using Visual Dependency Representations., in *EMNLP 2013*. Cited on pages 101 and 114.

H. J. Escalante, C. A. Hernández, J. A. Gonzalez, A. López-López, M. Montes, E. F. Morales, L. E. Sucar, L. Villaseñor, and M. Grubinger (2010). The segmented and annotated IAPR TC-12 benchmark, *Computer Vision and Image Understanding*, vol. 114(4), pp. 419–428. Cited on page 144.

M. Everingham, J. Sivic, and A. Zisserman (2006). "Hello! My name is... Buffy" - Automatic naming of characters in TV video, in *Proceedings of the British Machine Vision Conference (BMVC) 2006*. Cited on pages 14, 148, and 150.

M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman (2011). *The PASCAL Action Classification Taster Competition*. Cited on page 33.

M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman (2010). The pascal visual object classes (VOC) challenge, *International Journal of Computer Vision (IJCV)*, vol. 88(2), pp. 303–338. Cited on page 126.

H. Fang, S. Gupta, F. N. Iandola, R. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig (2015). From Captions to Visual Concepts and Back, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 16 and 85.

A. Farhadi, I. Endres, and D. Hoiem (2010a). Attribute-Centric Recognition for Cross-Category Generalization, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2010*. Cited on page 35.

A. Farhadi, M. Hejrati, M. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth (2010b). Every Picture Tells a Story: Generating Sentences from Images, in *Proceedings of the European Conference on Computer Vision (ECCV) 2010*. Cited on pages 69 and 85.

A. Fathi, A. Farhadi, and J. Rehg (2011). Understanding egocentric activities, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2011*. Cited on pages 35 and 36.

C. Fellbaum (Ed.) (1998). *WordNet: An Electronic Lexical Database*, The MIT Press. Cited on pages 18, 56, 71, 95, 108, and 130.

P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan (2010). Object Detection with Discriminatively Trained Part-Based Models, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 32. Cited on pages 2, 18, 44, and 49.

P. Felzenszwalb and D. Huttenlocher (2005). Pictorial Structures for Object Recognition, *International Journal of Computer Vision (IJCV)*. Cited on page 45.

V. Ferrari, M. Marin, and A. Zisserman (2008). Progressive Search Space Reduction for Human Pose Estimation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2008*. Cited on pages 46 and 49.

J. Ferryman (2007). Pets 2007 video database, in *Proceedings of the Tenth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance PETS 2007*. Cited on page 34.

J. R. Finkel, T. Grenager, and C. Manning (2005). Incorporating non-local information into information extraction systems by gibbs sampling, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2005*. Cited on page 150.

M. Fischler and R. Elschlager (1973). The Representation and Matching of Pictorial Structures, *IEEE Trans. Comput'73*. Cited on page 45.

A. Frome, G. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov (2013). DeViSE: A Deep Visual-Semantic Embedding Model, in *Advances in Neural Information Processing Systems (NIPS) 2013*. Cited on pages 37 and 135.

Y. Fu, T. Hospedales, T. Xiang, and S. Gong (2013). Learning Multi-modal Latent Attributes, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. Cited on pages 37 and 38.

A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach (2016). Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) 2016*. Cited on page 10.

L. Gagnon, C. Chapdelaine, D. Byrns, S. Foucher, M. Heritier, and V. Gupta (2010). A computer-vision-assisted system for Videodescription scripting, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops) 2010*. Cited on page 15.

Q. Gao, M. Doering, S. Yang, and J. Y. Chai (2016a). Physical causality of action verbs in grounded language understanding, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2016*. Cited on page 16.

Y. Gao, O. Beijbom, N. Zhang, and T. Darrell (2016b). Compact Bilinear Pooling, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 134, 135, and 136.

R. Girshick (2015a). Fast R-CNN, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on pages 126 and 144.

R. Girshick (2015b). Fast r-cnn, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on page 152.

R. Girshick, J. Donahue, T. Darrell, and J. Malik (2014). Rich feature hierarchies for accurate object detection and semantic segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 25.

G. Gkioxari, P. Arbelaez, L. Bourdev, and J. Malik (2013). Articulated Pose Estimation using Discriminative Armlet Classifiers, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2013*. Cited on pages 31 and 36.

G. Gkioxari, R. Girshick, and J. Malik (2015). Contextual action recognition with r* cnn, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on page 154.

X. Glorot and Y. Bengio (2010). Understanding the difficulty of training deep feedforward neural networks, in *International conference on artificial intelligence and statistics 2010*. Cited on page 126.

Y. Gong, L. Wang, M. Hodosh, J. Hockenmaier, and S. Lazebnik (2014). Improving image-sentence embeddings using large weakly annotated photo collections, in *Proceedings of the European Conference on Computer Vision (ECCV) 2014*. Cited on pages 23 and 135.

Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh (2017). Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on page 172.

M. Grötschel and Y. Wakabayashi (1989). A cutting plane algorithm for a clustering problem, *Mathematical Programming*, vol. 45(1), pp. 59–96. Cited on page 153.

M. Grubinger, P. Clough, H. Müller, and T. Deselaers (2006). The iapr tc-12 benchmark: A new evaluation resource for visual information systems, in *International Workshop OntoImage 2006*. Cited on page 144.

S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko (2013). YouTube2Text: Recognizing and Describing Arbitrary Activities Using Semantic Hierarchies and Zero-Shoot Recognition, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on pages 18, 22, 37, and 69.

S. Guadarrama, E. Rodner, K. Saenko, N. Zhang, R. Farrell, J. Donahue, and T. Darrell (2014). Open-vocabulary object retrieval, in *Robotics: science and systems 2014*. Cited on pages 129 and 130.

A. Gupta, P. Srinivasan, J. Shi, and L. Davis (2009). Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009*. Cited on pages 21, 36, and 38.

A. Habibian, T. Mensink, and C. G. Snoek (2014). Videostory: A new multimedia embedding for few-example recognition and translation of events, in *Proceedings of the ACM international conference on Multimedia (MM) 2014*. Cited on page 13.

L. Han, A. Kashyap, T. Finin, J. Mayfield, and J. Weese (2013). UMBC EBIQUITY-CORE: Semantic textual similarity systems, in *Proceedings of the Second Joint Conference on Lexical and Computational Semantics 2013*. Cited on page 12.

P. Hanckmann, K. Schutte, and G. J. Burghouts (2012). Automated Textual Descriptions for a Wide Range of Video Events with 48 Human Actions, in *Proceedings of the European Conference on Computer Vision Workshops (ECCV Workshops) 2012*. Cited on page 11.

D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor (2004). Canonical correlation analysis: An overview with application to learning methods, *Neural computation*, vol. 16(12), pp. 2639–2664. Cited on page 135.

K. He, X. Zhang, S. Ren, and J. Sun (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on page 126.

K. He, X. Zhang, S. Ren, and J. Sun (2016). Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 25, 133, 137, and 154.

L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, and T. Darrell (2016a). Generating visual explanations, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on pages 172 and 173.

L. A. Hendricks, S. Venugopalan, M. Rohrbach, R. Mooney, K. Saenko, and T. Darrell (2016b). Deep Compositional Captioning: Describing Novel Object Categories without Paired Training Data, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 118 and 169.

G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov (2012). Improving neural networks by preventing co-adaptation of feature detectors, *arXiv:1207.0580*. Cited on page 99.

S. Hochreiter and J. Schmidhuber (1997). Long short-term memory, *Neural computation*, vol. 9(8), pp. 1735–1780. Cited on pages 6, 19, 84, 123, and 155.

P. Hodosh, A. Young, M. Lai, and J. Hockenmaier (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions, *Transactions of the Association for Computational Linguistics (TACL)*. Cited on pages 82 and 143.

J. Hoffman, S. Guadarrama, E. Tzeng, J. Donahue, R. Girshick, T. Darrell, and K. Saenko (2014). LSDA: Large Scale Detection through Adaptation, in *Advances in Neural Information Processing Systems (NIPS) 2014*. Cited on pages 88, 98, and 154.

T. Hofmann (2001). Unsupervised learning by probabilistic latent semantic analysis, *Machine learning*, vol. 42(1-2), pp. 177–196. Cited on page 165.

R. Hu, M. Rohrbach, J. Andreas, T. Darrell, and K. Saenko (2017). Modeling Relationships in Referential Expressions with Compositional Modular Networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on pages 24, 26, 27, and 171.

R. Hu, M. Rohrbach, and T. Darrell (2016a). Segmentation from Natural Language Expressions, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on page 135.

R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell (2016b). Natural Language Object Retrieval, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 23, 26, 27, 119, 125, 126, 127, 128, 129, 130, 131, and 144.

T.-H. Huang, F. Ferraro, N. Mostafazadeh, I. Misra, A. Agrawal, J. Devlin, R. Girshick, X. He, P. Kohli, D. Batra, C. L. Zitnick, D. Parikh, L. Vanderwende, M. Galley, and M. Mitchell (2016). Visual Storytelling, in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) 2016*. Cited on pages 14 and 147.

I. Ilievski, S. Yan, and J. Feng (2016). A Focused Dynamic Attention Model for Visual Question Answering, *arXiv:1604.01485*. Cited on page 143.

S. Ioffe and C. Szegedy (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift, *arXiv:1502.03167*. Cited on pages 126 and 139.

H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. Black (2013). Towards understanding action recognition, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on pages 31, 36, and 38.

S. Ji, W. Xu, M. Yang, and K. Yu (2013). 3D convolutional neural networks for human action recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 35(1), pp. 221–231. Cited on page 36.

Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell (2014). Caffe: Convolutional architecture for fast feature embedding, in *Proceedings of the ACM international conference on Multimedia (MM) 2014*. Cited on pages 22 and 126.

J. Jin, K. Fu, R. Cui, F. Sha, and C. Zhang (2015). Aligning where to see and what to tell: image caption with region-based attention and scene factorization, *arXiv:1506.06272*. Cited on pages 121 and 123.

Q. Jin, J. Chen, S. Chen, Y. Xiong, and A. Hauptmann (2016). Describing Videos using Multi-modal Fusion, in *Proceedings of the ACM international conference on Multimedia (MM) 2016*. Cited on page 21.

J. Johnson, A. Karpathy, and L. Fei-Fei (2016). Densecap: Fully convolutional localization networks for dense captioning, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on page 26.

J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei (2015). Image Retrieval using Scene Graphs, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on page 24.

R. Johnson and T. Zhang (2015). Effective Use of Word Order for Text Categorization with Convolutional Neural Networks, in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies 2015*. Cited on page 165.

A. Joulin, K. Tang, and L. Fei-Fei (2014). Efficient image and video co-localization with frank-wolfe algorithm, in *Proceedings of the European Conference on Computer Vision (ECCV) 2014*. Cited on page 121.

V. Kantorov and I. Laptev (2014). Efficient feature extraction, encoding and classification for action recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 36.

L. Karlinsky, M. Dinerstein, and S. Ullman (2010). Using body-anchored priors for identifying actions in single images, in *Advances in Neural Information Processing Systems (NIPS) 2010*. Cited on page 36.

A. Karpathy and L. Fei-Fei (2015). Deep visual-semantic alignments for generating image descriptions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 16, 25, 84, 85, 119, 135, and 165.

A. Karpathy, A. Joulin, and L. Fei-Fei (2014a). Deep Fragment Embeddings for Bidirectional Image Sentence Mapping, in *Advances in Neural Information Processing Systems (NIPS) 2014*. Cited on pages 25, 119, 127, and 128.

A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei (2014b). Large-scale video classification with convolutional neural networks, in *Proceedings*

of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014. Cited on pages 29, 34, and 36.

D. Kaufman, G. Levi, T. Hassner, and L. Wolf (2016). Temporal Tessellation for Video Annotation and Summarization, *arXiv:1612.06950*. Cited on pages 19, 22, 110, 111, 112, 114, 116, 117, 118, and 176.

S. Kazemzadeh, V. Ordonez, M. Matten, and T. L. Berg (2014). Referitgame: Referring to objects in photographs of natural scenes, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) 2014*. Cited on pages 1, 23, 119, 120, 125, and 144.

M. U. G. Khan, L. Zhang, and Y. Gotoh (2011). Human Focused Video Description, in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops) 2011*. Cited on pages 21, 22, and 69.

D. Kingma and J. Ba (2014). Adam: A method for stochastic optimization, *arXiv:1412.6980*. Cited on pages 126 and 140.

K. Kipper, A. Korhonen, N. Ryant, and M. Palmer (2006). Extending VerbNet with Novel Verb Classes, in *Proceedings of the International Conference on Language Resources and Evaluation (LREC) 2006*. Cited on page 96.

R. Kiros, R. Salakhutdinov, and R. Zemel (2014). Multimodal Neural Language Models, in *Proceedings of the International Conference on Machine Learning (ICML) 2014*. Cited on pages 85 and 135.

R. Kiros, R. Salakhutdinov, and R. S. Zemel (2015a). Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models, *Transactions of the Association for Computational Linguistics (TACL)*, vol. 9, pp. 595–603. Cited on pages 84, 85, and 165.

R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler (2015b). Skip-Thought Vectors, in *Advances in Neural Information Processing Systems (NIPS) 2015*. Cited on page 133.

B. Klein, G. Lev, G. Sadeh, and L. Wolf (2015). Associating Neural Word Embeddings With Deep Image Representations Using Fisher Vectors, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 111 and 135.

D. Klein and C. D. Manning (2003). Accurate unlexicalized parsing, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2003*. Cited on page 76.

O. Kliper-Gross, T. Hassner, and L. Wolf (2012). The Action Similarity Labeling Challenge, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 34(3), pp. 615–621. Cited on pages 33 and 34.

P. Koehn (2010). *Statistical Machine Translation*, Cambridge University Press. Cited on page 4.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst (2007). Moses: Open Source Toolkit for Statistical Machine Translation, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2007*. Cited on pages 19, 94, and 97.

A. Kojima, T. Tamura, and K. Fukunaga (2002). Natural Language Description of Human Activities from Video Images Based on Concept Hierarchy of Actions, *International Journal of Computer Vision (IJCV)*, vol. 50(2), pp. 171–184. Cited on pages 11, 16, and 17.

C. Kong, D. Lin, M. Bansal, R. Urtasun, and S. Fidler (2014). What are you talking about? text-to-image coreference, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on pages 23 and 119.

R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei (2016). Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations, *International Journal of Computer Vision (IJCV)*. Cited on pages 140 and 171.

N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney, K. Saenko, and S. Guadarrama (2013). Generating Natural-Language Video Descriptions Using Text-Mined Knowledge, in *Proceedings of the Conference on Artificial Intelligence (AAAI) 2013*. Cited on pages 16, 18, and 69.

J. Krishnamurthy and T. Kollar (2013). Jointly learning to parse and perceive: connecting natural language to the physical world, *Transactions of the Association for Computational Linguistics (TACL)*. Cited on page 119.

A. Krizhevsky, I. Sutskever, and G. E. Hinton (2012). ImageNet Classification with Deep Convolutional Neural Networks, in *Advances in Neural Information Processing Systems (NIPS) 2012*. Cited on page 82.

H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre (2011). HMDB: A Large Video Database for Human Motion Recognition, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2011*. Cited on pages 33, 34, 35, and 36.

G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg (2011). Baby talk: Understanding and generating simple image descriptions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on pages 69, 85, and 165.

A. Kumar, O. Irsoy, J. Su, J. Bradbury, R. English, B. Pierce, P. Ondruska, I. Gulrajani, and R. Socher (2016). Ask me anything: Dynamic memory networks for natural language processing, in *Proceedings of the International Conference on Machine Learning (ICML) 2016*. Cited on page 135.

P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi (2012). Collective generation of natural image descriptions, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2012*. Cited on pages 69 and 85.

P. Kuznetsova, V. Ordonez, T. L. Berg, and Y. Choi (2014). TREETALK: Composition and Compression of Trees for Image Descriptions., *Transactions of the Association for Computational Linguistics (TACL)*, vol. 2(10), pp. 351–362. Cited on page 85.

S. Kwak, M. Cho, I. Laptev, J. Ponce, and C. Schmid (2015). Unsupervised Object Discovery and Tracking in Video Collections, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on page 121.

F. D. la Torre, J. Hodgins, J. Montano, S. Valcarcel, R. Forcada, and J. Macey (2009). Guide to the CMU Multimodal Activity Database, Technical report CMU-RI-TR-08-22, Robotics Institute. Cited on pages 1, 34, and 35.

Lakritz and Salway (2006). The Semi-Automatic Generation of Audio Description from Screenplays, Technical report CS-06-05, University of Surrey. Cited on pages 15 and 81.

C. Lampert, H. Nickisch, and S. Harmeling (2013). Attribute-Based Classification for Zero-Shot Learning of Object Categories, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. Cited on pages 31 and 54.

I. Laptev (2005). On Space-Time Interest Points, in *International Journal of Computer Vision (IJCV) 2005*. Cited on page 35.

I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld (2008). Learning realistic human actions from movies, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2008*. Cited on pages 14, 18, 35, 50, 82, 84, and 87.

I. Laptev and P. Pérez (2007). Retrieving actions in movies, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2007*. Cited on pages 33 and 34.

M. D. A. Lavie (2014). Meteor Universal: Language Specific Translation Evaluation for Any Target Language, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2014*. Cited on pages 12, 14, and 100.

Q. Le, W. Zou, S. Yeung, and A. Ng (2011). Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on page 36.

G. Lev, G. Sadeh, B. Klein, and L. Wolf (2015). RNN Fisher Vectors for Action Recognition and Image Annotation, in *Proceedings of the European Conference on Computer Vision (ECCV) 2015*. Cited on page 111.

G. Li, S. Ma, and Y. Han (2015). Summarization-based Video Caption via Deep Neural Networks, in *Proceedings of the ACM international conference on Multimedia (MM) 2015*. Cited on page 21.

L.-J. Li and F.-F. Li (2007). What, where and who? Classifying events by scene and object recognition, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2007*. Cited on page 35.

L.-J. Li, H. Su, E. P. Xing, and F.-F. Li (2010). Object Bank: A High-Level Image Representation for Scene Classification & Semantic Feature Sparsification, in *Advances in Neural Information Processing Systems (NIPS) 2010*. Cited on page 18.

S. Li, G. Kulkarni, T. Berg, A. Berg, and Y. Choi (2011). Composing simple image descriptions using web-scale N-grams, in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning (CoNLL) 2011*. Cited on page 85.

W. Li, R. Zhao, T. Xiao, and X. Wang (2014). DeepReID: Deep filter pairing neural network for person re-identification, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 149.

Y. Li, Y. Song, L. Cao, J. Tetreault, L. Goldberg, A. Jaimes, and J. Luo (2016). TGIF: A New Dataset and Benchmark on Animated GIF Description, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 14, 15, and 93.

C. Liang, C. Xu, J. Cheng, and H. Lu (2011). TVParser: An automatic TV video parsing method, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on page 82.

C.-Y. Lin (2004). Rouge: A package for automatic evaluation of summaries, in *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop 2004*. Cited on pages 14 and 100.

D. Lin, S. Fidler, C. Kong, and R. Urtasun (2014a). Visual semantic search: Retrieving videos via complex textual queries, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on pages 25, 119, and 171.

D. Lin, S. Fidler, C. Kong, and R. Urtasun (2015a). Generating multi-sentence natural language descriptions of indoor scenes, in *Proceedings of the British Machine Vision Conference (BMVC) 2015*. Cited on page 26.

T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick (2014b). Microsoft COCO: Common objects in context, in *Proceedings of the European Conference on Computer Vision (ECCV) 2014*. Cited on pages 82, 130, and 139.

T.-Y. Lin, A. RoyChowdhury, and S. Maji (2015b). Bilinear CNN models for fine-grained visual recognition, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on pages 134 and 135.

X. Lin and D. Parikh (2016). Leveraging visual question answering for image-caption ranking, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on page 172.

C. Liu, J. Mao, F. Sha, and A. Yuille (2017). Attention Correctness in Neural Image Captioning, in *Proceedings of the Conference on Artificial Intelligence (AAAI) 2017*. Cited on pages 26, 27, and 172.

J. Liu, B. Kuipers, and S. Savarese (2011). Recognizing Human Actions by Attributes, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on page 31.

J. Liu, J. Luo, and M. Shah (2009). Recognizing realistic actions from videos 'in the wild', in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009*. Cited on pages 29, 33, and 35.

J. Liu, S. McCloskey, and Y. Liu (2012). Training data recycling for multi-level learning, in *Proceedings of the International Conference on Pattern Recognition (ICPR) 2012*. Cited on page 38.

C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei (2016). Visual relationship detection with language priors, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on page 24.

J. Lu, C. Xiong, D. Parikh, and R. Socher (2017). Knowing When to Look: Adaptive Attention via A Visual Sentinel for Image Captioning, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on page 25.

J. Lu, J. Yang, D. Batra, and D. Parikh (2016). Hierarchical Co-Attention for Visual Question Answering, in *Advances in Neural Information Processing Systems (NIPS) 2016*. Cited on pages 135 and 143.

B. D. Lucas and T. Kanade (1981). An iterative image registration technique with an application to stereo vision, in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) 1981*. Cited on page 152.

R. Luo and G. Shakhnarovich (2017). Comprehension-guided referring expressions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on pages 24, 26, 27, and 173.

T. Maharaj, N. Ballas, A. Rohrbach, A. Courville, and C. Pal (2017). A dataset and exploration of models for understanding video data through fill-in-the-blank question-answering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on pages 6, 16, 92, 165, and 173.

M. Malinowski and M. Fritz (2014). A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input, in *Advances in Neural Information Processing Systems (NIPS) 2014*. Cited on page 165.

M. Malinowski, M. Rohrbach, and M. Fritz (2016). Ask Your Neurons: A Deep Learning Approach to Visual Question Answering, *arXiv: 1605.02697*. Cited on page 143.

J. Malmaud, J. Huang, V. Rathod, N. Johnston, A. Rabinovich, and K. Murphy (2015). What's Cookin'? Interpreting Cooking Videos using Text, Speech and Vision, in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) 2015*. Cited on page 13.

J. Mao, J. Huang, A. Toshev, O. Camburu, A. Yuille, and K. Murphy (2016). Generation and Comprehension of Unambiguous Object Descriptions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 23, 24, 26, 27, and 165.

J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille (2015). Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN), in *Proceedings of the International Conference on Learning Representations (ICLR) 2015*. Cited on pages 85 and 135.

M. Marszalek, I. Laptev, and C. Schmid (2009). Actions in context, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009*. Cited on pages 14, 16, 29, 33, 34, 35, 82, and 87.

C. Matuszek, N. Fitzgerald, L. Zettlemoyer, L. Bo, and D. Fox (2012). A Joint Model of Language and Perception for Grounded Attribute Learning, in *Proceedings of the International Conference on Machine Learning (ICML) 2012*. Cited on page 119.

R. Messing, C. Pal, and H. Kautz (2009). Activity recognition using the velocity histories of tracked keypoints, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2009*. Cited on pages 34, 35, and 50.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean (2013). Distributed Representations of Words and Phrases and their Compositionality, in *Advances in Neural Information Processing Systems (NIPS) 2013*. Cited on pages 18, 19, and 24.

M. Mitchell, J. Dodge, A. Goyal, K. Yamaguchi, K. Stratos, X. Han, A. Mensch, A. C. Berg, T. L. Berg, and H. D. III (2012). Midge: Generating Image Descriptions From Computer Vision Detections, in *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL) 2012*. Cited on page 85.

A. Mittal, A. Zisserman, and P. Torr (2011). Hand detection using multiple proposals, in *Proceedings of the British Machine Vision Conference (BMVC) 2011*. Cited on pages 31, 36, and 48.

T. S. Motwani and R. J. Mooney (2012). Improving Video Activity Recognition using Object Recognition and Text Mining, in *Proceedings of the European Conference on Artificial Intelligence (ECAI) 2012*. Cited on page 37.

V. K. Nagaraja, V. I. Morariu, and L. S. Davis (2016). Modeling context between objects for referring expression understanding, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on pages 24, 26, 27, and 171.

P. Natarajan and R. Nevatia (2008). View and scale invariant action recognition using multiview shape-flow models, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2008*. Cited on pages 33 and 34.

J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng (2011). Multimodal deep learning, in *Proceedings of the International Conference on Machine Learning (ICML) 2011*. Cited on page 135.

J. Niebles, C.-W. Chen, and L. Fei-Fei (2010). Modeling Temporal Structure of Decomposable Motion Segments for Activity Classification, in *Proceedings of the European Conference on Computer Vision (ECCV) 2010*. Cited on pages 33 and 36.

M.-E. Nilsback and A. Zisserman (2008). Automated flower classification over a large number of classes, in *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP) 2008*. Cited on page 37.

H. Noh, P. H. Seo, and B. Han (2015). Image Question Answering using Convolutional Neural Network with Dynamic Parameter Prediction, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 135 and 143.

S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis, *et al.* (2011). A large-scale benchmark dataset for event recognition in surveillance video, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on page 34.

V. Ordonez, G. Kulkarni, and T. L. Berg (2011). Im2Text: Describing Images Using 1 Million Captioned Photographs, in *Advances in Neural Information Processing Systems (NIPS) 2011*. Cited on page 82.

P. Over, G. Awad, J. Fiscus, B. Antonishek, M. Michel, A. F. Smeaton, W. Kraaij, and G. Quéenot (2010). TRECVID 2010 – An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics, in *Proceedings of TRECVID 2010*. Cited on pages 12 and 17.

P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, B. Shaw, A. F. Smeaton, and G. Quéenot (2012). TRECVID 2012 – An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics, in *Proceedings of TRECVID 2012*. Cited on pages 12, 33, and 82.

B. Packer, K. Saenko, and D. Koller (2012). A combined pose, object, and feature model for action understanding, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2012*. Cited on page 36.

P. Pan, Z. Xu, Y. Yang, F. Wu, and Y. Zhuang (2016a). Hierarchical Recurrent Neural Encoder for Video Representation With Application to Captioning, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 20 and 170.

Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui (2016b). Jointly Modeling Embedding and Translation to Bridge Video and Language, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on page 20.

K. Papineni, S. Roukos, T. Ward, and W. jing Zhu (2002). BLEU: a Method for Automatic Evaluation of Machine Translation, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2002*. Cited on pages 12, 14, and 100.

O. M. Parkhi, E. Rahtu, and A. Zisserman (2015a). It's in the bag: Stronger supervision for automated face labelling, in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops) 2015*. Cited on pages 148, 150, 154, and 157.

O. M. Parkhi, A. Vedaldi, and A. Zisserman (2015b). Deep Face Recognition, in *Proceedings of the British Machine Vision Conference (BMVC) 2015*. Cited on pages 149 and 153.

A. Patron-Perez, M. Marszalek, A. Zisserman, and I. D. Reid (2010). High Five: Recognising human interactions in TV shows, in *Proceedings of the British Machine Vision Conference (BMVC) 2010*. Cited on pages 33 and 34.

J. Pennington, R. Socher, and C. D. Manning (2014). GloVe: Global Vectors for Word Representation, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) 2014*. Cited on page 142.

Á. Peris, M. Bolaños, P. Radeva, and F. Casacuberta (2016). Video description using bidirectional recurrent neural networks, in *Proceedings of the International Conference on Artificial Neural Networks (ICANN) 2016*. Cited on pages 20 and 170.

N. Pham and R. Pagh (2013). Fast and Scalable Polynomial Kernels via Explicit Feature Maps, in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2013*. Cited on pages 135 and 137.

H. Pirsiavash and D. Ramanan (2012). Detecting activities of daily living in first-person camera views, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2012*. Cited on page 35.

H. Pirsiavash and D. Ramanan (2014). Parsing videos of actions with segmental grammars, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 36.

L. Pishchulin, M. Andriluka, and B. Schiele (2014). Fine-grained activity recognition with holistic and pose based features, in *Proceedings of the German Confeence on Pattern Recognition (GCPR) 2014*. Cited on page 153.

B. Plummer, L. Wang, C. Cervantes, J. Caicedo, J. Hockenmaier, and S. Lazebnik (2015). Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on pages 22, 23, 119, 120, 125, 127, 128, 129, 135, 143, 144, and 165.

B. Plummer, L. Wang, C. Cervantes, J. Caicedo, J. Hockenmaier, and S. Lazebnik (2016). Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models, *International Journal of Computer Vision (IJCV)*. Cited on pages 23, 26, 143, and 144.

W. Pu, N. Liu, S. Yan, J. Yan, K. Xie, and Z. Chen (2007). Local word bag model for text categorization, in *Seventh IEEE International Conference on Data Mining 2007*. Cited on page 165.

V. Ramanathan, A. Joulin, P. Liang, and L. Fei-Fei (2014). Linking people in videos with "their" names using coreference resolution, in *Proceedings of the European Conference on Computer Vision (ECCV) 2014*. Cited on pages 15, 148, 149, 150, and 151.

V. Ramanathan, P. Liang, and L. Fei-Fei (2013). Video Event Understanding using Natural Language Descriptions, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on page 37.

V. Ramanishka, A. Das, D. H. Park, S. Venugopalan, L. A. Hendricks, M. Rohrbach, and K. Saenko (2016). Multimodal Video Description, in *Proceedings of the ACM international conference on Multimedia (MM) 2016*. Cited on page 20.

V. Ramanishka, A. Das, J. Zhang, and K. Saenko (2017). Top-down Visual Saliency Guided by Captions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on pages 26, 27, 171, and 172.

M. Raptis and L. Sigal (2013). Poselet Key-framing: A Model for Human Activity Recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2013*. Cited on page 36.

M. Regneri, A. Koller, and M. Pinkal (2010). Learning Script Knowledge with Web Experiments, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2010*. Cited on pages 38 and 43.

M. Regneri, M. Rohrbach, D. Wetzel, S. Thater, B. Schiele, and M. Pinkal (2013). Grounding Action Descriptions in Videos, *Transactions of the Association for Computational Linguistics (TACL)*, vol. 1, pp. 25–36. Cited on pages 11, 12, 15, 32, 41, 71, 76, and 93.

M. Ren, R. Kiros, and R. Zemel (2015). Image Question Answering: A Visual Semantic Embedding Model and a New Dataset, in *Advances in Neural Information Processing Systems (NIPS) 2015*. Cited on page 26.

M. Rodriguez, J. Ahmed, and M. Shah (2008). Action MACH a spatio-temporal Maximum Average Correlation Height filter for action recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2008*. Cited on pages 29, 33, and 36.

D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Forster, G. Troster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, J. Doppler, C. Holzmann, M. Kurz, G. Holl, R. Chavarriaga, H. Sagha, H. Bayati, M. Creatura, and J. del R. Millan (2010). Collecting complex activity data sets in highly rich networked sensor environments, in *Proceedings of the International Conference on Networked Sensing Systems (INSS) 2010*. Cited on page 35.

A. Rohrbach, M. Rohrbach, R. Hu, T. Darrell, and B. Schiele (2016a). Grounding of Textual Phrases in Images by Reconstruction, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on page 9.

A. Rohrbach, M. Rohrbach, W. Qiu, A. Friedrich, M. Pinkal, and B. Schiele (2014). Coherent Multi-Sentence Video Description with Variable Level of Detail, in *Proceedings of the German Confeence on Pattern Recognition (GCPR) 2014*. Cited on pages 8 and 19.

A. Rohrbach, M. Rohrbach, and B. Schiele (2015a). The Long-Short Story of Movie Description, in *Proceedings of the German Confeence on Pattern Recognition (GCPR) 2015*. Cited on page 9.

A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele (2015b). A Dataset for Movie Description, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on page 9.

A. Rohrbach, M. Rohrbach, S. Tang, S. J. Oh, and B. Schiele (2017a). Generating Descriptions with Grounded and Co-Referenced People, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on page 10.

A. Rohrbach, A. Torabi, M. Rohrbach, N. Tandon, C. Pal, H. Larochelle, A. Courville, and B. Schiele (2017b). Movie Description, *International Journal of Computer Vision (IJCV)*. Cited on page 9.

M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele (2012a). A database for fine grained activity detection of cooking activities, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2012*. Cited on pages 8, 39, 42, and 47.

M. Rohrbach, S. Ebert, and B. Schiele (2013a). Transfer Learning in a Transductive Setting, in *Advances in Neural Information Processing Systems (NIPS) 2013*. Cited on pages 55 and 56.

M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele (2013b). Translating Video Content to Natural Language Descriptions, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on pages 4, 16, 19, 22, 32, 37, 39, 69, 73, 74, 75, 76, 77, 78, 79, 82, 84, 94, 95, 97, 101, 166, 167, and 176.

M. Rohrbach, M. Regneri, M. Andriluka, S. Amin, M. Pinkal, and B. Schiele (2012b). Script data for attribute-based recognition of composite activities, in *Proceedings of the European Conference on Computer Vision (ECCV) 2012*. Cited on pages 1, 8, 12, 15, and 39.

M. Rohrbach, A. Rohrbach, M. Regneri, S. Amin, M. Andriluka, M. Pinkal, and B. Schiele (2016b). Recognizing Fine-Grained and Composite Activities Using Hand-Centric Features and Script Data, *International Journal of Computer Vision (IJCV)*. Cited on page 8.

M. Rohrbach, M. Stark, and B. Schiele (2011). Evaluating Knowledge Transfer and Zero-Shot Learning in a Large-Scale Setting, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on pages 54, 55, and 58.

M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele (2010). What helps Where - and Why? Semantic Relatedness for Knowledge Transfer, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2010*. Cited on page 37.

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei (2015). ImageNet Large Scale Visual Recognition Challenge, *International Journal of Computer Vision (IJCV)*, vol. 115(3), pp. 211–252. Cited on pages 5 and 88.

M. Ryoo and J. Aggarwal (2009). Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2009*. Cited on page 34.

F. Sadeghi, S. K. Divvala, and A. Farhadi (2015). Viske: Visual knowledge extraction and question answering by visual verification of relation phrases, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on page 24.

G. Salton and C. Buckley (1988). Term-weighting approaches in automatic text retrieval, in *Information Processing And Management 1988*. Cited on page 56.

A. Salway (2007). A corpus-based analysis of audio description, *Media for all: Subtitling for the deaf, audio description and sign language*, vol. 1(2), p. 3. Cited on pages 2, 15, 81, and 84.

A. Salway, B. Lehane, and N. E. O'Connor (2007). Associating characters with events in films, in *Proceedings of the ACM international conference on Image and video retrieval (CIVR) 2007*. Cited on pages 15 and 84.

B. Sapp, A. Toshev, and B. Taskar (2010). Cascaded models for articulated pose estimation, in *Proceedings of the European Conference on Computer Vision (ECCV) 2010*. Cited on pages 46 and 47.

R. Schank and R. Abelson (1977). *Scripts, Plans, Goals and Understanding*, Psychology Press. Cited on page 42.

M. Schmidt (2013). *UGM: Matlab code for undirected graphical models*, *di.ens.fr/∼mschmidt/Software/UGM.html*. Cited on page 73.

F. Schroff, D. Kalenichenko, and J. Philbin (2015). FaceNet: A Unified Embedding for Face Recognition and Clustering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on page 149.

C. Schuldt, I. Laptev, and B. Caputo (2004). Recognizing human actions: a local SVM approach, in *Proceedings of the International Conference on Pattern Recognition (ICPR) 2004*. Cited on pages 29, 33, 34, and 35.

K. K. Schuler, A. Korhonen, and S. W. Brown (2009). VerbNet overview, extensions, mappings and applications., in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) 2009*. Cited on page 96.

R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra (2016). Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization, in *Advances in Neural Information Processing Systems Workshops (NIPS Workshops) 2016*. Cited on page 172.

R. Shetty and J. Laaksonen (2015). Video captioning with recurrent networks based on frame-and video-level features and visual content classification, *arXiv:1512.02949*. Cited on pages 20, 110, 111, 112, 113, 114, 115, 116, 118, and 176.

R. Shetty and J. Laaksonen (2016). Frame- and segment-level features and candidate pool evaluation for video caption generation, in *Proceedings of the ACM international conference on Multimedia (MM) 2016*. Cited on pages 20, 110, 111, 112, and 114.

A. Shin, K. Ohnishi, and T. Harada (2016). Beyond Caption To Narrative: Video Captioning With Multiple Sentences, in *Proceedings of the IEEE International Conference on Image Processing (ICIP) 2016*. Cited on pages 21 and 170.

J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake (2011). Real-time human pose recognition in parts from single depth images, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on page 36.

N. Siddharth, A. Barbu, and J. M. Siskind (2014). Seeing What You're Told: Sentence-Guided Activity Recognition In Video, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 17.

G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta (2016). Hollywood in homes: Crowdsourcing data collection for activity understanding, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on page 13.

J. Sill, G. Takács, L. Mackey, and D. Lin (2009). Feature-weighted linear stacking, *arXiv:0911.0460*. Cited on page 38.

K. Simonyan and A. Zisserman (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition, in *Proceedings of the International Conference on Learning Representations (ICLR) 2015*. Cited on pages 26, 111, 126, 144, and 153.

P. Singh, T. Lin, E. T. Mueller, G. Lim, T. Perkins, and W. L. Zhu (2002). Open Mind Common Sense: Knowledge acquisition from the general public, in *OTM Confederated International Conferences "On the Move to Meaningful Internet Systems" 2002*. Cited on page 43.

V. Singh and R. Nevatia (2011). Action Recognition in Cluttered Dynamic Scenes using Pose-Specific Part Models, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2011*. Cited on page 33.

J. Sivic, M. Everingham, and A. Zisserman (2009). "Who are you?"-Learning person specific classifiers from video, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009*. Cited on pages 15, 148, and 150.

J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman (2005). Discovering objects and their location in images, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2005*. Cited on page 165.

R. Socher and L. Fei-Fei (2010). Connecting Modalities: Semi-supervised Segmentation and Annotation of Images Using Unaligned Text Corpora, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2010*. Cited on page 37.

R. Socher, M. Ganjoo, C. D. Manning, and A. Ng (2013). Zero-Shot Learning Through Cross-Modal Transfer, in *Advances in Neural Information Processing Systems (NIPS) 2013*. Cited on page 37.

R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng (). Grounded Compositional Semantics for Finding and Describing Images with Sentences, *Transactions of the Association for Computational Linguistics (TACL)*, vol. 2, pp. 207–218. Cited on pages 85 and 135.

H. O. Song, R. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darrell (2014). On learning to localize objects with minimal supervision, in *Proceedings of the International Conference on Machine Learning (ICML) 2014*. Cited on page 121.

K. Soomro, A. R. Zamir, and M. Shah (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild, *arXiv:1212.0402*. Cited on pages 33 and 34.

A. Srikantha and J. Gall (2014). Discovering Object Classes from Activities, in *Proceedings of the European Conference on Computer Vision (ECCV) 2014*, pp. 415–430, Springer. Cited on page 32.

S. Stein and S. McKenna (2013). Combining Embedded Accelerometers with Computer Vision for Recognizing Food Preparation Activities, in *Proceedings of the ACM international joint conference on Pervasive and ubiquitous computing (UbiComp) 2013*. Cited on page 35.

C. Sun and R. Nevatia (2014). Semantic aware video transcription using random forest classifiers, in *Proceedings of the European Conference on Computer Vision (ECCV) 2014*. Cited on page 18.

Y. Sun, X. Wang, and X. Tang (2015). Deeply learned face representations are sparse, selective, and robust, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on page 149.

J. Sung, C. Ponce, B. Selman, and A. Saxena (2011). Human Activity Detection from RGBD Images., *Plan, activity, and intent recognition*, vol. 64. Cited on page 36.

I. Sutskever, O. Vinyals, and Q. V. Le (2014). Sequence to sequence learning with neural networks, in *Advances in Neural Information Processing Systems (NIPS) 2014*. Cited on pages 19, 123, and 133.

C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich (2015). Going deeper with convolutions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on page 111.

Y. Taigman, M. Yang, M. Ranzato, and L. Wolf (2014). Deepface: Closing the gap to human-level performance in face verification, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 149.

C. C. Tan, Y.-G. Jiang, and C.-W. Ngo (2011). Towards textually describing complex video contents with audio-visual concept classifiers, in *Proceedings of the ACM international conference on Multimedia (MM) 2011*. Cited on pages 12, 17, 21, 22, and 69.

N. Tandon, G. de Melo, A. De, and G. Weikum (2015). Knowlywood: Mining activity knowledge from hollywood narratives, in *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM) 2015*. Cited on page 14.

N. Tandon, C. D. Hariman, J. Urbani, A. Rohrbach, M. Rohrbach, and G. Weikum (2016). Commonsense in Parts: Mining Part-Whole Relations from the Web and Image Tags, in *Proceedings of the Conference on Artificial Intelligence (AAAI) 2016*. Cited on page 1.

K. Tang, L. Fei-Fei, and D. Koller (2012). Learning Latent Temporal Structure for Complex Event Detection, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2012*. Cited on pages 29, 36, and 38.

K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei (2014). Co-localization in real-world images, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014*. Cited on page 121.

K. Tang, B. Yao, L. Fei-Fei, and D. Koller (2013). Combining the Right Features for Complex Event Recognition, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on page 36.

S. Tang, B. Andres, M. Andriluka, and B. Schiele (2015). Subgraph decomposition for multi-target tracking, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on page 153.

M. Tapaswi, M. Baeuml, and R. Stiefelhagen (2012). "Knock! Knock! Who is it?" probabilistic person identification in TV-series, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2012*. Cited on pages 15, 148, and 150.

M. Tapaswi, M. Bäuml, and R. Stiefelhagen (2015). Book2movie: Aligning video scenes with book chapters, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 15 and 119.

M. Tapaswi, Y. Zhu, R. Stiefelhagen, A. Torralba, R. Urtasun, and S. Fidler (2016). MovieQA: Understanding Stories in Movies through Question-Answering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 15, 16, and 165.

G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler (2010). Convolutional learning of spatio-temporal features, in *Proceedings of the European Conference on Computer Vision (ECCV) 2010*, pp. 140–153, Springer. Cited on page 36.

J. B. Tenenbaum and W. T. Freeman (2000). Separating style and content with bilinear models, *Neural computation*, vol. 12(6), pp. 1247–1283. Cited on pages 134 and 136.

M. Tenorth, J. Bandouch, and M. Beetz (2009). The TUM Kitchen Data Set of Everyday Manipulation Activities for Motion Tracking and Action Recognition, in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops) 2009*. Cited on pages 1, 34, and 35.

C. L. Teo, Y. Yang, H. Daume, C. Fermuller, and Y. Aloimonos (2012). Towards a Watson that sees: Language-guided action recognition for robots, in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) 2012*. Cited on page 37.

J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. J. Mooney (2014). Integrating Language and Vision to Generate Natural Language Descriptions of Videos in the Wild, in *Proceedings of the International Conference on Computational Linguistics (COLING) 2014*. Cited on page 18.

B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li (2015). The new data and new challenges in multimedia research, *arXiv:1503.01817*, vol. 1(8). Cited on pages 14 and 140.

K. M. Ting and I. H. Witten (1997). Stacked Generalization: when does it work?, in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) 1997*. Cited on page 38.

C. Tomasi and T. Kanade (1991). Detection and tracking of feature points, Technical report CMU-CS-91-132, Carnegie Mellon University. Cited on page 152.

A. Torabi, C. Pal, H. Larochelle, and A. Courville (2015). Using Descriptive Video Services to Create a Large Data Source for Video Annotation Research, *arXiv:1503.01070v1*. Cited on pages 6, 9, 15, 85, and 86.

A. Torabi, N. Tandon, and L. Sigal (2016). Learning Language-Visual Embedding for Movie Understanding with Natural-Language, *arXiv:1609.08124*. Cited on pages 6, 16, 92, and 173.

K. Toutanova, D. Klein, C. D. Manning, and Y. Singer (2003). Feature-rich part-of-speech tagging with a cyclic dependency network, in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) 2003*. Cited on pages 71 and 93.

J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders (2013). Selective search for object recognition, *International Journal of Computer Vision (IJCV)*, vol. 104(2), pp. 154–171. Cited on pages 126 and 144.

A. Vedaldi and B. Fulkerson (2008). *VLFeat: An Open and Portable Library of Computer Vision Algorithms*, *http://www.vlfeat.org/*. Cited on page 75.

A. Vedaldi and A. Zisserman (2010). Efficient Additive Kernels via Explicit Feature Maps, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2010*. Cited on page 58.

R. Vedantam, C. L. Zitnick, and D. Parikh (2015). CIDEr: Consensus-based Image Description Evaluation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 14, 100, 101, and 114.

S. Venugopalan, L. A. Hendricks, R. Mooney, and K. Saenko (2016). Improving LSTM-based Video Description with Linguistic Knowledge Mined from Text, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) 2016*. Cited on page 20.

S. Venugopalan, L. A. Hendricks, M. Rohrbach, R. Mooney, T. Darrell, and K. Saenko (2017). Captioning images with diverse objects, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on page 169.

S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko (2015a). Sequence to Sequence – Video to Text, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on pages 1, 19, 20, 110, 111, 112, 113, 114, 115, 116, 160, and 176.

S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko (2015b). Sequence to Sequence – Video to Text, *arXiv:1505.00487v2*. Cited on pages 105, 106, 107, and 176.

S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko (2015c). Translating Videos to Natural Language Using Deep Recurrent Neural Networks, in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) 2015*. Cited on pages 16 and 19.

P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol (2008). Extracting and composing robust features with denoising autoencoders, in *Proceedings of the International Conference on Machine Learning (ICML) 2008*. Cited on page 121.

O. Vinyals, A. Toshev, S. Bengio, and D. Erhan (2015). Show and Tell: A Neural Image Caption Generator, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015*. Cited on pages 1, 84, 85, 125, and 165.

P. Viola and M. Jones (2001). Rapid object detection using a boosted cascade of simple features, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2001*. Cited on page 2.

H. Wang, A. Kläser, C. Schmid, and C. Liu (2013a). Dense Trajectories and Motion Boundary Descriptors for Action Recognition, *International Journal of Computer Vision (IJCV)*, vol. 103(1), pp. 60–79. Cited on pages 18, 20, 31, 35, 36, 38, 49, 50, 51, 59, 75, and 111.

H. Wang, A. Kläser, C. Schmid, and C.-L. Liu (2011). Action Recognition by Dense Trajectories, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on page 50.

H. Wang and C. Schmid (2013). Action Recognition with Improved Trajectories, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on pages 29, 35, 39, 88, 97, and 154.

H. Wang, M. Ullah, A. Klaser, I. Laptev, and C. Schmid (2009a). Evaluation of local spatio-temporal features for action recognition, in *Proceedings of the British Machine Vision Conference (BMVC) 2009*. Cited on page 35.

J. Wang, K. Markert, and M. Everingham (2009b). Learning Models for Object Recognition from Natural Language Descriptions, in *Proceedings of the British Machine Vision Conference (BMVC) 2009*. Cited on page 37.

L. Wang, Y. Li, and S. Lazebnik (2016a). Learning Deep Structure-Preserving Image-Text Embeddings, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 1, 23, 26, 127, 128, 129, 144, and 171.

L. Wang, Y. Qiao, and X. Tang (2013b). Mining Motion Atoms and Phrases for Complex Action Recognition, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2013*. Cited on page 29.

M. Wang, M. Azab, N. Kojima, R. Mihalcea, and J. Deng (2016b). Structured matching for phrase localization, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on page 23.

P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona (2010). Caltech-UCSD birds 200, Technical report CNS-TR-201, California Institute of Technology. Cited on page 37.

J. Weston, S. Bengio, and N. Usunier (2011). Wsabie: Scaling up to large vocabulary image annotation, in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) 2011*. Cited on page 135.

Q. Wu, P. Wang, C. Shen, A. v. d. Hengel, and A. Dick (2016). Ask Me Anything: Free-form Visual Question Answering Based on Knowledge from External Sources, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 143 and 172.

J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba (2010). SUN database: Large-scale scene recognition from abbey to zoo, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Cited on pages 18 and 82.

C. Xiong, S. Merity, and R. Socher (2016). Dynamic Memory Networks for Visual and Textual Question Answering, in *Proceedings of the International Conference on Machine Learning (ICML) 2016*. Cited on pages 135 and 143.

H. Xu and K. Saenko (2016). Ask, Attend and Answer: Exploring Question-Guided Spatial Attention for Visual Question Answering, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on pages 138 and 143.

J. Xu, T. Mei, T. Yao, and Y. Rui (2016). MSR-VTT: A Large Video Description Dataset for Bridging Video and Language, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 11, 13, 15, 82, 93, and 111.

K. Xu, J. Ba, R. Kiros, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio (2015a). Show, attend and tell: Neural image caption generation with visual attention, in *Proceedings of the International Conference on Machine Learning (ICML) 2015*. Cited on pages 23, 25, 85, 111, 121, 123, 134, 138, 149, and 156.

R. Xu, C. Xiong, W. Chen, and J. J. Corso (2015b). Jointly modeling deep video and compositional text to bridge vision and language in a unified framework, in *Proceedings of the Conference on Artificial Intelligence (AAAI) 2015*. Cited on page 18.

W. Yang, Y. Wang, and G. Mori (2011). Recognizing Human Actions from Still Images with Latent Poses, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on page 36.

Y. Yang and D. Ramanan (2011). Articulated pose estimation with flexible mixtures-of-parts., in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2011*. Cited on pages 45, 46, and 47.

Y. Yang and D. Ramanan (2013). Articulated Human Detection with Flexible Mixtures of Parts, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 35. Cited on page 36.

Z. Yang, X. He, J. Gao, L. Deng, and A. Smola (2016a). Stacked Attention Networks for Image Question Answering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 134, 138, and 143.

Z. Yang, Y. Yuan, Y. Wu, R. Salakhutdinov, and W. W. Cohen (2016b). Encode, Review, and Decode: Reviewer Module for Caption Generation, in *Advances in Neural Information Processing Systems (NIPS) 2016*. Cited on page 25.

A. Yao, J. Gall, G. Fanelli, and L. V. Gool (2011a). Does Human Action Recognition Benefit from Pose Estimation?, in *Proceedings of the British Machine Vision Conference (BMVC) 2011*. Cited on page 36.

B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. J. Guibas, and L. Fei-Fei (2011b). Action Recognition by Learning Bases of Action Attributes and Parts, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2011*. Cited on page 33.

B. Yao and F.-F. Li (2012). Recognizing Human-Object Interactions in Still Images by Modeling the Mutual Context of Objects and Human Poses, *IEEE Transactions on*

Pattern Analysis and Machine Intelligence (TPAMI), vol. 34(9), pp. 1691–1703. Cited on pages 35 and 36.

L. Yao, N. Ballas, K. Cho, J. R. Smith, and Y. Bengio (2016). Empirical performance upper bounds for image and video captioning, in *Proceedings of the International Conference on Learning Representations (ICLR) 2016*. Cited on pages 20 and 165.

L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville (2015). Describing videos by exploiting temporal structure, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on pages 16, 20, 25, 106, 110, 111, 112, 113, 114, 115, 116, 121, and 176.

L. Yeffet and L. Wolf (2009). Local Trinary Patterns for human action recognition, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2009*. Cited on page 35.

S. Yeung, O. Russakovsky, N. Jin, M. Andriluka, G. Mori, and L. Fei-Fei (2015). Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos, *arXiv:1507.05738*. Cited on page 121.

Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo (2016). Image captioning with semantic attention, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 20 and 25.

P. Young, A. Lai, M. Hodosh, and J. Hockenmaier (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions, *Transactions of the Association for Computational Linguistics (TACL)*, vol. 2, pp. 67–78. Cited on pages 23 and 85.

H. Yu and J. M. Siskind (2013). Grounded language learning from videos described with sentences, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2013*. Cited on pages 17, 18, 25, 69, and 171.

H. Yu and J. M. Siskind (2015a). Learning to Describe Video with Weak Supervision by Exploiting Negative Sentential Information, in *Proceedings of the Conference on Artificial Intelligence (AAAI) 2015*. Cited on page 17.

H. Yu and J. M. Siskind (2015b). Sentence Directed Video Object Codetection, *arXiv:1506.02059*. Cited on page 121.

H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu (2016a). Video Paragraph Captioning using Hierarchical Recurrent Neural Networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on pages 21 and 170.

L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg (2016b). Modeling context in referring expressions, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on pages 23, 26, and 171.

L. Yu, H. Tan, M. Bansal, and T. L. Berg (2017a). A Joint Speaker-Listener-Reinforcer Model for Referring Expressions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on pages 24 and 27.

Y. Yu, H. Ko, J. Choi, and G. Kim (2017b). End-to-end Concept Word Detection for Video Captioning, Retrieval, and Question Answering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*. Cited on pages 20, 110, 111, 112, and 114.

J. Yuan, Z. Liu, and Y. Wu (2009). Discriminative subvolume search for efficient action detection, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009*. Cited on pages 33 and 34.

M. Zanfir, E. Marinoiu, and C. Sminchisescu (2016). Spatio-Temporal Attention Models for Grounded Video Captioning, in *Proceedings of the Asian Conference on Computer Vision (ACCV) 2016*. Cited on pages 23, 26, 27, and 171.

W. Zaremba and I. Sutskever (2014). Learning to execute, *arXiv preprint arXiv:1410.4615*. Cited on page 155.

K.-H. Zeng, T.-H. Chen, C.-Y. Chuang, Y.-H. Liao, J. C. Niebles, and M. Sun (2017). Leveraging Video Descriptions to Learn Video Question Answering, in *Proceedings of the Conference on Artificial Intelligence (AAAI) 2017*. Cited on page 172.

K.-H. Zeng, T.-H. Chen, J. C. Niebles, and M. Sun (2016). Title Generation for User Generated Videos, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on pages 11, 14, 15, and 93.

J. Zhang, Z. Lin, J. Brandt, X. Shen, and S. Sclaroff (2016). Top-down neural attention by excitation backprop, in *Proceedings of the European Conference on Computer Vision (ECCV) 2016*. Cited on pages 24 and 172.

L. Zhang, M. U. G. Khan, and Y. Gotoh (2011). Video scene classification based on natural language description, in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops) 2011*. Cited on pages 37 and 38.

Z. Zhong and H. T. Ng (2010). It makes sense: A wide-coverage word sense disambiguation system for free text, in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) 2010*. Cited on page 96.

B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba (2016). Learning deep features for discriminative localization, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on page 172.

B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva (2014). Learning Deep Features for Scene Recognition using Places Database., *Advances in Neural Information Processing Systems (NIPS)*. Cited on pages 5, 82, 88, 98, and 154.

B. Zhou, Y. Tian, S. Sukhbaatar, A. Szlam, and R. Fergus (2015a). Simple Baseline for Visual Question Answering, *arXiv:1512.02167*. Cited on pages 134 and 143.

D. Zhou, O. Bousquet, T. N. Lal, Jason Weston, and B. Schölkopf (2004). Learning with Local and Global Consistency, in *Advances in Neural Information Processing Systems (NIPS) 2004*. Cited on page 56.

E. Zhou, Z. Cao, and Q. Yin (2015b). Naive-Deep Face Recognition: Touching the Limit of LFW Benchmark or Not?, *arXiv:1501.04690*. Cited on page 149.

L. Zhu, Z. Xu, Y. Yang, and A. G. Hauptmann (2015a). Uncovering Temporal Context for Video Question and Answering, *arXiv:1511.04670*. Cited on page 16.

Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei (2016). Visual7W: Grounded Question Answering in Images, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*. Cited on page 140.

Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler (2015b). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2015*. Cited on pages 15, 16, and 119.

A. Zinnen, U. Blanke, and B. Schiele (2009). An Analysis of Sensor-Oriented vs. Model-Based Activity Recognition, in *Proceedings of the International Semantic Web Conference (ISWC) 2009*. Cited on pages 31, 38, and 49.

C. L. Zitnick and P. Dollár (2014). Edge boxes: Locating object proposals from edges, in *Proceedings of the European Conference on Computer Vision (ECCV) 2014*, pp. 391–405, Springer. Cited on pages 126 and 144.

I. Zukerman and D. Litman (2001). Natural language processing and user modeling: Synergies and limitations, *User Modeling and User-Adapted Interaction*. Cited on page 70.