

UCLA

UCLA Electronic Theses and Dissertations

Title

Joint Image-Text Representation Learning

Permalink

<https://escholarship.org/uc/item/66f282s6>

Author

Ren, Zhou

Publication Date

2016

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
Los Angeles

Joint Image-Text Representation Learning

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Computer Science

by

Zhou Ren

2016

© Copyright by

Zhou Ren

2016

ABSTRACT OF THE DISSERTATION

Joint Image-Text Representation Learning

by

Zhou Ren

Doctor of Philosophy in Computer Science

University of California, Los Angeles, 2016

Professor Alan Loddon Yuille, Chair

It was a dream to make computers intelligent. Like humans who are capable of understanding information of multiple modalities such as video, text, audio, *etc.*, teaching computers to jointly understand multi-modal information is a necessary and essential step towards artificial intelligence. And how to jointly represent multi-modal information is critical to such step. Although a lot of efforts have been devoted to exploring the representation of each modality individually, it is an open and challenging problem to learn joint multi-modal representation.

In this dissertation, we explore joint image-text representation models based on Visual-Semantic Embedding (VSE). VSE has been recently proposed and shown to be effective for joint representation. The key idea is that by learning a mapping from images into a semantic space, the algorithm is able to learn a compact and effective joint representation. However, existing approaches simply map each text concept and each whole image to single points in the semantic space. We propose several novel visual-semantic embedding models that use (1) text concept modeling, (2) image-level modeling, and (3) object-level modeling. In particular, we first introduce a novel Gaussian Visual-Semantic Embedding (GVSE) model that leverages the visual information to model text concepts as density distributions rather than single points in semantic space. Then, we propose Multiple Instance Visual-Semantic Embedding (MIVSE) via image-level

modeling, which discovers and maps the semantically meaningful image sub-regions to their corresponding text labels. Next, we present a fine-grained object-level representation in images, Scene-Domain Active Part Models (SDAPM), that reconstructs and characterizes 3D geometric statistics between objects parts in 3D scene-domain. Finally, we explore advanced joint representations for other visual and textual modalities, including joint image-sentence representation and joint video-sentence representation.

Extensive experiments have demonstrated that the proposed joint representation models are superior to existing methods on various tasks involving image, video and text modalities, including image annotation, zero-shot learning, object and parts detection, pose and viewpoint estimation, image classification, text-based image retrieval, image captioning, video annotation, and text-based video retrieval.

The dissertation of Zhou Ren is approved.

Yingnian Wu

Douglas S. Parker

Demetri Terzopoulos

Alan Loddon Yuille, Committee Chair

University of California, Los Angeles

2016

To Xinzhu, and my family

TABLE OF CONTENTS

1	Introduction	1
1.1	Background	2
1.1.1	Image and video representation	2
1.1.2	Text representation	2
1.1.3	Multi-modal joint representation	3
1.2	Organization	4
1.3	Contributions	6
2	Basics: Visual-Semantic Embedding Model	8
2.1	Overview	8
3	Text Concept Modeling: Gaussian Visual-Semantic Embedding	11
3.1	Introduction	11
3.2	Background	13
3.3	Gaussian Visual-Semantic Embedding	14
3.3.1	Constructing the semantic label space	14
3.3.2	Modeling text concepts as Gaussian distributions	14
3.3.3	Training and inference with GVSE	15
3.4	Experiments	17
3.4.1	Dataset and implementation	17
3.4.2	Experiments on image classification	18
3.4.3	Experiments on text-based image retrieval	18
3.5	Summary	22
4	Image-Level Modeling: Multiple Instance Visual-Semantic Embedding	24
4.1	Introduction	24
4.2	Background	26

4.2.1	Multi-label image annotation	27
4.2.2	Zero-shot learning	28
4.3	Semantic Space and Our Multi-label Baseline	29
4.3.1	Constructing the semantic label space	29
4.3.2	Our embedding baseline model: from single-label to multi-label	29
4.4	Multiple Instance Visual-Semantic Embedding	30
4.4.1	Modeling subregion-to-label correspondence	31
4.4.2	Rank-weighted loss	32
4.4.3	Learning multiple instance visual-semantic embedding	33
4.4.4	Inference with multiple instance visual-semantic embedding	35
4.5	Experiments	36
4.5.1	Implementation	36
4.5.2	Experiments on multi-label image annotation	36
4.5.3	Experiments on zero-shot learning	42
4.6	Summary	46
5	Object-Level Modeling: Scene-Domain Active Part Models	47
5.1	Introduction	47
5.2	Background	50
5.3	Scene-Domain Active Part Models	52
5.3.1	Preliminary: 2D part-based object models	52
5.3.2	Modeling active parts in the 3D scene-domain	55
5.3.3	Modeling appearance with occlusions	57
5.4	Inference	58
5.4.1	3D landmark shape and viewpoint recovery	59
5.5	Learning	59
5.5.1	Learning the 2D image-domain parameter Θ	60
5.5.2	Learning the 3D geometric subspace \mathbf{B}	60
5.6	Experiments	61

5.6.1	Datasets	61
5.6.2	Implementation details	62
5.6.3	Experiments on object and parts detection	63
5.6.4	Experiments on pose and viewpoint estimation	67
5.6.5	More qualitative results	71
5.7	Summary	71
6	Advanced: Joint Image-Sentence Representation & Joint Video-Sentence Representation	74
6.1	Joint Image-Sentence Representation	74
6.1.1	Image-sentence embedding	74
6.1.2	Image captioning	77
6.1.3	Embedding-Guided Image Captioning	78
6.1.4	Experiments on image captioning	79
6.2	Joint Video-Sentence Representation	82
6.2.1	Experiments on video annotation	83
6.2.2	Experiments on text-based video retrieval	85
6.3	Summary	85
7	Conclusion and Future Research	86
7.1	Summary	86
7.2	Future Research Directions	87
	Appendix A Derivation of Inference Solution of Scene-Domain Active Part Models	89

LIST OF FIGURES

1.1	Joint Image-Text Representation.	4
3.1	Existing Visual-Semantic Embedding models simply represent the text concepts as single points in the semantic space, as indicated by the circular symbols (we use different colors to indicate different categories). We propose the Gaussian Visual-Semantic Embedding (GVSE) model, which however leverages the visual information indicated as the pentagram symbols (pentagram symbols of the same color indicate various images of the same category) to model text concepts as Gaussian distributions. Modeling text concepts as densities innately represents uncertainties of text concepts, and has the benefit of geometric interpretation such as inclusion and intersection.	12
3.2	Training framework of the proposed Gaussian Visual-Semantic Embedding model.	16
3.3	The top-9 image retrieval results of searching trained text “ <i>airport terminal</i> ”. The incorrect image shares very similar visual appearance. . . .	19
3.4	The top-9 image retrieval results of searching trained text “ <i>kitchen</i> ”. The incorrect images belong to “ <i>kitchenette</i> ” which is a very close concept.	20
3.5	The top-9 retrieval results of searching untrained text “ <i>infant</i> ”. The returned images belong to “ <i>nursery</i> ” and “ <i>hospital room</i> ”, which are conceptually related.	21
3.6	The top-9 retrieval results of searching untrained text “ <i>sports</i> ”. The returned images belong to “ <i>football stadium</i> ”, “ <i>baseball stadium</i> ” and “ <i>stadium</i> ”, which are conceptually related.	22

4.1	(a) An example of an image with multiple labels that are listed in the middle. (b) We observe that different labels may correspond to various image subregions, but not necessarily the whole image, such as the labels <i>clouds</i> , <i>sun</i> , <i>bird</i> , which are associated with the subregions in the bounding boxes.	25
4.2	Illustration of our Multiple Instance Visual-Semantic Embedding model, which is composed of two key components: (a) construct image subregion set; (b) establish the subregion-to-label correspondence by embedding semantically meaningful subregions close to their corresponding labels in the semantic space (the red symbols illustrate the embedding of text labels, and symbols of other colors indicate that of different image subregions.). Note that the bounding boxes are for visualization only, and they are not provided in training.	31
4.3	The deep network architecture of MIVSE model, composed of 4 components: (a) subregion image features extraction; (b) image features embedding; (c) text features embedding; (d) joint embedding function learning guided by the MIVSE loss layer.	33
4.4	The image annotation results using MIVSE. The predicted labels are listed according to the ranking (we show top-3 predictions if the number of ground truth labels is smaller than or equal to 3, and show top-5 predictions otherwise.). The semantically meaningful subregions of predicted labels are shown in bounding boxes of the same color indicated with the labels. The ground truth labels (GT) are listed according to alphabetic order. The last row shows examples where the predicted annotations are reasonable even they are not included in GT. Better viewed in color.	38

4.5	Zero-shot learning results on Places205 images using our MIVSE model and the ranking loss baseline, respectively. The correctly predicted labels are shown in blue (note that there is only one ground truth label for each image in Places205.). In each image, the semantically meaningful image subregion of each correctly predicted label using MIVSE is shown in green bounding box.	45
5.1	(a) A motivation of SDAPM: geometric variations lead to varying part configurations. Thus, by modeling in the scene-domain, SDAPM can better capture objects’ geometric statistics and provides richer object descriptions, including 2D parts localization, 3D landmark shape as well as camera viewpoint estimation, in addition to the 2D object bounding box, as shown in (b).	49
5.2	Our method provides a richer object representation and improves the detection results. The blue bounding boxes correspond to the whole object detection, and boxes of other colors correspond to semantic parts respectively, which may indicate different parts across classes. (a) shows one representative result for each of the six animal classes. (b) shows detection results of the cat class to illustrate the ability of our model to robustly represent objects under non-rigid deformations, viewpoint changes, and occlusions. (c) shows typical examples that are correctly localized by our Scene-Domain Active Part Model (SDAPM) but missed by DPM.	65
5.3	Our method provides a richer object representation including 2D pose, 3D landmark shape, and viewpoint. (a) shows typical poses that are correctly estimated by our method w. HOG but mis-estimated by [Hejrati and Ramanan, 2012] (<i>e.g.</i> the right arm). (b) shows the 2D pose and 3D landmark shape estimations of our method for human in varying viewpoints. (c) gives some failed examples of our method.	69

5.4	More detection results of our model with HOG feature on the six animal classes. Each row shows one of the six classes respectively. The blue bounding box indicates the whole object detection result, and the boxes of other colors indicate the parts. As we see, our method can robustly represent the objects under non-rigid deformations, viewpoint changes, and occlusions, with parts localization.	72
5.5	Our method offers a richer object representation in 3D, in addition to the 2D pose. For each test image, we show, in a sub-figure on the right side of the image, the estimated 3D landmark shape using a different viewpoint from that of the test image, so as to illustrate the 3D configuration of the estimated landmark shapes.	73
6.1	The general framework of joint image-sentence embedding.	75
6.2	The general framework of image captioning.	77
6.3	The framework of the proposed Embedding-Guided Image Captioning.	78
6.4	The framework of the proposed video-sentence embedding.	82

LIST OF TABLES

3.1	Image classification results on the MIT Places205 dataset, shown in %.	17
4.1	Image annotation results on NUS-WIDE shown in %, with $k = 3$ and $k = 5$ annotated labels per image, respectively. See text for the definition of “Upper bound”.	39
4.2	Image annotation results of MIVSE and its four variants on NUS-WIDE shown in %, with $k = 3$ and $k = 5$ annotated labels per image, respectively.	41
4.3	Zero-shot learning results on the MIT Places205 dataset, shown in %.	44
5.1	Average precision for animal detection on PASCAL VOC 2010. Our method outperforms all baselines of part-based models.	63
5.2	Part localization performance on PASCAL VOC 2010. The numbers are “PCP of our method” / “PCP of SSDPM [Azizpour and Laptev, 2012]”.	66
5.3	Average detection precision of SDAPM and its three variants on the six animal classes of PASCAL VOC 2010 dataset.	67
5.4	2D pose estimation performance on Human3.6M dataset, comparing with MoP [Yang and Ramanan, 2012] and MH-Car [Hejrati and Ramanan, 2012]. The reported numbers are PCP (Probability of Correct Pose).	68
5.5	3D landmark shape estimation on Human3.6M dataset and camera view-point estimation on PASCAL Car 2007 dataset, comparing with MA-N [Arie-Nachimson and Basri, 2009] and MH-Car [Hejrati and Ramanan, 2012].	70

6.1	The image captioning results in MSCOCO dataset [Lin et al., 2014] of the proposed Embedding-Guided Image Captioning approach, of different values of λ , with beam size set to be 50.	79
6.2	The image captioning results in MSCOCO dataset [Lin et al., 2014] of the proposed Embedding-Guided Image Captioning approach, of different beam sizes. The last two rows are the results of the Google NIC baseline [Vinyals et al., 2015].	80
6.3	The image captioning results in MSCOCO dataset [Lin et al., 2014] of the proposed Embedding-Guided Image Captioning approach, comparing with several existing approaches including Google NIC [Vinyals et al., 2015], M-RNN [Mao et al., 2015], LRCN [Donahue et al., 2015], MSR/CMU [Chen and Zitnick, 2015], Spatial_ATT [Xu et al., 2015], Semantic_ATT [You et al., 2016].	81
6.4	The video annotation results of joint video-sentence embedding in the MSR-VTT dataset [Xu et al., 2016], using different video features.	83
6.5	The text-based video retrieval results of joint video-sentence embedding in the MSR-VTT dataset [Xu et al., 2016], using different video features.	84

ACKNOWLEDGMENTS

Foremost, I would like to express my sincere gratitude to my wonderful PhD advisor Professor Alan Yuille. I would like to thank him for the continuous encouragement and support of my PhD study and research. His insight and expertise greatly helped me grow to be a researcher. I also thank him for providing me flexibility and a lot of opportunities for my career. Besides, I would like to thank Professor Demetri Terzopoulos, Professor Stott Parker and Professor Yingnian Wu for serving on my thesis committee and providing suggestions on improving my work.

I was fortunate enough to have worked closely with a number of great and amazing researchers in the field of computer vision and machine learning, during and before my PhD study. I would like to sincerely thank my master advisor Professor Junsong Yuan at Nanyang Technological University for his countless support and encouragement on both my life and career, to Professor Wenyu Liu at Huazhong University of Science and Technology, China, Dr. Zhengyou Zhang at Microsoft Research Redmond, Professor Raquel Urtasun and Professor Sanja Fidler at University of Toronto, Professor Chaohui Wang at Université Paris-Est, LIGM - CNRS, Dr. Hailin Jin, Dr. Zhe Lin, and Dr. Chen Fang at Adobe Research, Dr. Xiaoyu Wang, Dr. Ning Zhang, Dr. Jia Li, Dr. Vitor Carvalho, and Dr. Xutao Lv at Snapchat Research, for sharing their knowledge and guiding me on various exciting research projects.

My appreciation also goes to my fellow labmates at UCLA Center for Cognition, Vision, and Learning (CCVL) and my friends at UCLA, for the insightful discussions, for their help, and for the great time we had together: Dr. Liang-Chieh Chen, Weichao Qiu, Dr. Xianjie Chen, Junhua Mao, Dr. Jun Zhu, Xiaochen Lian, Fangting Xia, Jianyu Wang, Peng Wang, Alex Wong, Chenxi Liu, Zhuotun Zhu, Dr. Boyan Bonev, Dr. Xiaodi Hou, Yu Zhu, Dr. Xuan Dong, Dr. Vittal Premachandran, Dr. Chunyu Wang, Dr.

Bo Xin, Dr. Lingxi Xie, Dr. George Papandreou, Dr. Quannan Li, Yixin Zhu, Dr. Chenfangfu Jiang, Nikolaos Karianakis, Zheng Xing, Dr. Youjin Hu, Dr. Seyoung Park, Lae Un Kim, Yan Chen, and Dr. Yinyin Chen. I also thank my friends I made during the internships at TTIC, Adobe Research, and Snapchat Research: Dr. Xianming Liu, Dr. Luoqi Liu, Quanzeng You, Dr. Haoxiang Li, Ning Xu, Jie Feng, Yibing Song, Dr. Jianming Zhang, Dr. Jia Xu, Wei Han, Dr. Linjie Yang, whose friendship would be a lifetime fortune to me.

Last but not least, I want to dedicate this thesis to my parents, my paternal and maternal grandparents, my uncles and aunties, my cousins, my girlfriend Xinzhu, and Xinzhu's family for their unconditional love and support along this journey. It is their trust, patience, and encouragement that make me happy and confident. I cherish them, who make everything in my life possible and meaningful.

VITA

- 2010 B.E. in Communication Engineering, Huazhong University of Science and Technology, Wuhan, China.
- 2012 M.E. in Electrical & Electronic Engineering, Nanyang Technological University, Singapore.
- 2014 M.S. in Computer Science, UCLA.
- 2010–2012 Researcher & Project Officer, Media Technology Lab, Nanyang Technological University, Singapore.
- 2013–2014 Teaching Assistant/Associate, UCLA.
- 2012–2016 Research Assistant, UCLA.
- Summer 2013 Research Intern, Toyota Technological Institute, Chicago, IL.
- Summer 2015 Deep Learning Research Intern, Adobe Research, San Jose, CA.
- Spring 2016 Research Intern, Snapchat, Venice, CA.

PUBLICATIONS

Zhou Ren, Hailin Jin, Zhe Lin, Chen Fang, and Alan Yuille: Joint Image-Text Representation by Gaussian Visual-Semantic Embedding. In ACM Multimedia Conference, 2016.

Zhou Ren, Hailin Jin, Zhe Lin, Chen Fang, and Alan Yuille: Multi-Instance Visual-Semantic Embedding. In arXiv:1512.06963, 2015.

Zhou Ren, Chaohui Wang and Alan Yuille: Scene-Domain Active Part Models for Object Representation. In IEEE International Conference on Computer Vision (ICCV), 2015.

Xiaowei Ding, Jianing Pang, Zhou Ren, Mariana Diaz-Zamudio, Chenfangfu Jiang, Zhaoyang Fan, Daniel Berman, Debiao Li, Demetri Terzopoulos, Piotr Slomka, and Damini Dey: Automated Pericardial Fat Quantification from Coronary Magnetic Resonance Angiography. In Journal of Medical Imaging (JMI), 2016.

Xiaowei Ding, Jianing Pang, Zhou Ren, Mariana Zamudio, Daniel Berman, Debiao Li, Demetri Terzopoulos, Piotr Slomka, and Damini Dey: Automated Pericardial Fat Quantification from Coronary Magnetic Resonance Angiography. In Medical Image Understanding and Analysis (MIUA), 2015.

CHAPTER 1

Introduction

It was a dream to make computers intelligent. Researchers in various fields of artificial intelligence have been sharing the work and focusing on different aspects [Marr, 1981]. To name a few, research in computer vision explores technologies to capture, transmit, interpret, and understand visual information, such as images, videos, *etc.* And research in natural language processing provides techniques to analyze, retrieve, process, and understand textual information, such as words, phrases, sentences, *etc.* Among them, how to represent the input information such as images and sentences is a fundamental problem, which has been investigated for many years.

It is important to focus on each specific problem. However, it is also important to collaborate with one another. As human beings, we live in a world filled of information from multiple modalities [Ren et al., 2013b], *e.g.*, video, text, audio, *etc.* Thus, it is important and beneficial to teach computers to be intelligent, by utilizing multi-modal information.

This dissertation concentrates on an important problem: learning the joint representation of visual and textual information. How to jointly represent multi-modal information is an essential problem, which is crucial for the research and applications of intelligent multi-modal processing and understanding, such as image annotation, image classification, image captioning, video understanding, text-based image retrieval, text-based video retrieval, *etc.*

1.1 Background

1.1.1 Image and video representation

Given visual input, representation means to extract the key information and represent it into a compact form. Image and video representation is a fundamental problem in computer vision. Researchers have been focusing on this problem to provide better representations that can capture the visual information in a compact way. From the traditional color histogram, to SIFT [Lowe, 2004], HOG [Dalal and Triggs, 2005], STIPs [Laptev and Lindeberg, 2003], *etc*, various carefully designed hand-crafted features have been proposed for various vision problems. Inspired by other fields including text analysis and information theory, other advanced image and video representations have been proposed, *e.g.*, sparse coding [Yang et al., 2009], fisher vectors [Perronnin and Dance, 2007], bag-of-words [Chen et al., 2012].

Recently, the learnt representation with deep convolutional neural network (CNN) structures has shown superior generalization power and expressivity in various vision recognition tasks [LeCun et al., 1990; Krizhevsky et al., 2012; Zeiler and Fergus, 2014; Ren et al., 2015b; Szegedy et al., 2015; Simonyan and Zisserman, 2015; Girshick, 2015; Tran et al., 2015], including AlexNet [Krizhevsky et al., 2012], VGGNet [Simonyan and Zisserman, 2015], GoogleNet [Szegedy et al., 2015], C3D [Tran et al., 2015], *etc*.

1.1.2 Text representation

Traditional short-text representation, such as words and phrases, is one-hot representation, given a dictionary containing all words. And in text analysis, researchers focus on improving the representation for long-text, such as paragraph and document, *e.g.*, bag-of-words, term frequency-inverse document frequency (TF-IDF) [Rajaraman and Ullman, 2011], *etc*. Also, modeling techniques based on Hidden Markov Models (HMM)

[Baum and Petrie, 1966] have been proposed, to capture the temporal transition model of texts. Recently, researchers have been proposed Recurrent Neural Network (RNN) models for long-texts representation, such as Long-Short Term Memory (LSTM) model [Hochreiter and Schmidhuber, 1997] and Gated Recurrent Unit (GRU) model [Chung et al., 2014].

Representation of short-text, such as words and phrases, is a challenging problem. Recently, distributed representations, such as N-gram models [Bengio et al., 2003a; Schwenk, 2007; Mikolov et al., 2011], word2vec [Mikolov et al., 2013], GloVe [Pennington et al., 2014], have shown the capacity to provide semantically meaningful embedding features for short text terms, in learning from unannotated text data from the Internet. These methods are able to learn similar embedding vectors for semantically related words because of the fact that those words are more likely to appear in similar semantic contexts.

1.1.3 Multi-modal joint representation

Learning the joint representation of inputs of multiple modalities is important. In the case of joint image-text representation, data from both modalities are mapped into a common joint embedding space, where the embedding of images with similar textual descriptions should be close to each other, as shown in Figure 1.1. Some of the earliest research on joint images-texts representation has focused on learning the co-occurrences between image regions and tags using a generative model [Barnard and Forsyth, 2001; Blei and Jordan, 2003]. Recently, a number of successful recent approaches [Hardoon et al., 2004; Rasiwasia et al., 2010; Hwang and Grauman, 2011; Gong et al., 2014b] to learning an joint embedding rely on Canonical Correlation Analysis (CCA) [Hotelling, 1936], a classic technique that maps the visual and textual features, into a common embedding space where the correlation between the two modalities is maximized.

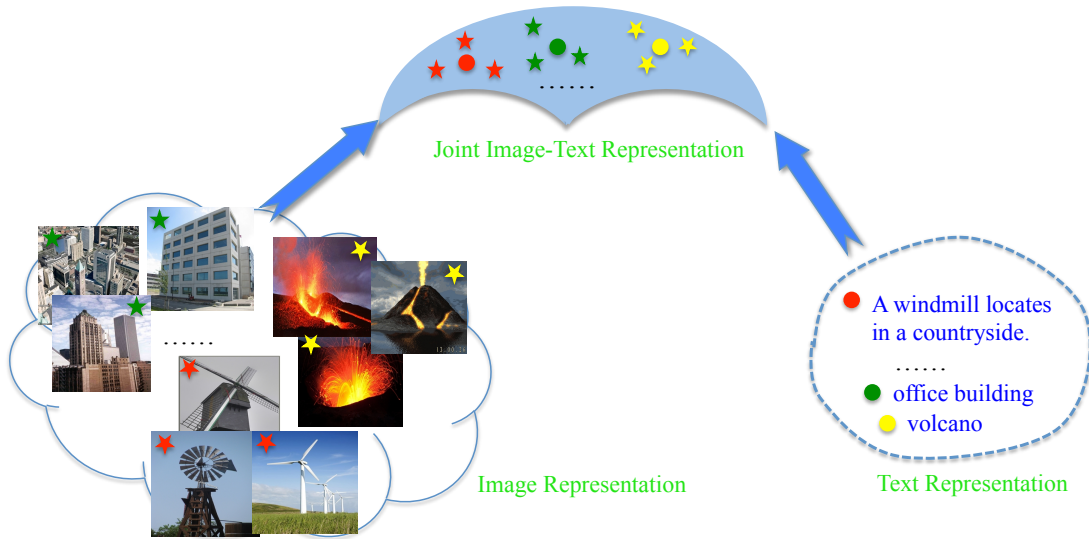


Figure 1.1: *Joint Image-Text Representation.*

More recently, Visual-Semantic Embedding models for image classification were proposed by leveraging the distributed representation of labels, *e.g.*, Frome *et al.* [Frome *et al.*, 2013] and Norouzi *et al.* [Norouzi *et al.*, 2014], which possesses richer representation and better generalizing ability to unseen classes. Ren *et al.* [Ren *et al.*, 2015a] proposed multiple instance visual-semantic embedding for multi-label image annotation. Vilnis *et al.* [Vilnis and McCallum, 2015] presented Gaussian embedding models for word representation. Ren *et al.* [Ren *et al.*, 2016] proposed Gaussian Visual-Semantic Embedding for joint image-text representation. Image-sentence embedding models [Karpathy and Fei-Fei, 2015; Kiros *et al.*, 2015] were proposed for image captioning.

1.2 Organization

In this dissertation, three modeling techniques of joint image-text representation based on Visual-Semantic Embedding will be presented, including text concept modeling, image-level modeling, and object-level modeling. This dissertation will also present

the joint representation of other advanced visual and textual inputs, such as joint image-sentence representation, joint video-sentence representation. The dissertation is organized as follows:

- Chapter 2 gives an overview of Visual-Semantic Embedding models. The mathematical background and intuition is also explained.
- Chapter 3 presents a novel joint image-text representation model via text concept modeling: Gaussian Visual-Semantic Embedding. The research on multi-modal embedding is discussed in Section 3.2. Section 3.3 introduces Gaussian Visual-Semantic Embedding as well as its training and inference procedures, which instead of maps each textual concept into a single point in the semantic space, maps that to a density. In Section 3.4, we experimentally validate the effectiveness of GVSE in two multi-modal tasks, *i.e.*, image classification and text-based image retrieval. Finally, we summarize in Section 3.5 and discuss in future directions.
- Chapter 4 describes a novel joint image-text representation model via image-level modeling: Multiple Instance Visual-Semantic Embedding. Section 4.2 reviews literature on two related applications. And Section 4.3 introduces how to construct semantic space and the multi-label embedding baseline. Then, in Section 4.4, Multiple Instance Visual-Semantic Embedding and its training, inference procedures are discussed in details. Then, Section 4.5 validates the effectiveness of MIVSE in two multi-modal tasks, *i.e.*, multi-label image annotation and zero-shot learning on image classification. Finally, we summarize in Section 4.6.
- Chapter 5 focuses on object representation learning in images. The related work is introduced in Section 5.2. In Section 5.3, a novel object model, Scene-Domain Active Part Models, is presented. Then, the inference and learning procedures are discussed in Section 5.4 and 5.5. Next, experiments on two fine-grained object

tasks are introduced in Section 5.6, including object and part detection as well as pose and viewpoint estimation. Finally a summary is given in Section 5.7.

- Chapter 6 explores joint representation of advanced visual and textual modalities, including joint image-sentence representation and joint video-sentence representation. In Section 6.1, we introduce joint image-sentence representation and present an Embedding-Guided Image Captioning framework. Then, joint video-sentence representation is presented in Section 6.2 and we apply it to video annotation and text-based video retrieval. Finally, we draw conclusions in Section 6.3.
- Chapter 7 summarizes the dissertation as proposing three levels of modeling to learn an elegant joint image-sentence representation, including text concept modeling, image-level modeling, and object modeling. Some interesting future research topics are given at the end.

1.3 Contributions

Original contributions presented in this dissertation span the areas of joint image-text representation, object representation, image understanding, video understanding, *etc.* In particular, the following issues have been addressed.

- A novel text-concept modeling algorithm in joint image-text representation has been developed, where only visual information is leveraged.
- A novel joint representation for multi-label images has been presented by discovering and mapping semantically meaningful image subregions to the corresponding labels.
- A novel object representation approach, which reconstructs and characterizes the 3D geometric statistics between objects' parts in a 3D scene-domain, has been

proposed.

- A novel algorithm that learns 3D statistics from 2D training data in the image-domain alone has been developed.
- We have developed joint image-sentence representation and joint video-sentence representation for advanced visual and textual modalities, with advantages that have been experimentally validated in various multi-modal tasks.

This dissertation mainly concentrates on the issues in joint representation of visual and textual modalities. However, many techniques developed in this work are easily extended to many other tasks and areas. For example, the joint representation of visual and acoustic information can be developed using similar techniques proposed in this dissertation, which could be beneficial for video analysis, *etc.*

CHAPTER 2

Basics: Visual-Semantic Embedding Model

In recent years, Visual-Semantic Embedding (VSE) models [Frome et al., 2013; Norouzi et al., 2014; Kiros et al., 2015; Ren et al., 2015a] have shown impressive performance for joint image-text representation. By leveraging the semantic information contained in unannotated text data, VSE models explicitly map images into a rich semantic space, with the goal that images of the same category are mapped to nearby locations around the corresponding text label, and text labels are embedded in such a smooth space with respect to some measure of similarity, that is, similar text concepts should be mapped into nearby locations. In this chapter, we recap the mathematical overview of Visual-Semantic Embedding.

2.1 Overview

Given a multi-label image dataset $\mathcal{D} \equiv \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, each image is represented by a d -dimensional feature vector, $\mathbf{x}_i \in \mathcal{X} \stackrel{\text{def}}{=} \mathbb{R}^d$, and each image is associated with multiple labels, $\mathbf{y}_i = (y_i^1, \dots, y_i^{T_i})$, where the label set size T_i can be varied across different images (namely different images can have different number of labels). There are totally n distinct labels in dataset \mathcal{D} , *i.e.*, $y_i^t \in \mathcal{Y} \equiv \{1, 2, \dots, n\}, \forall i, t$.

Previous methods [Makadia et al., 2008; Guillaumin et al., 2009; Gong et al., 2014a] formulate multi-label image annotation as a classification problem, which predefine a

fixed set of class labels \mathcal{Y} , and learn to predict the labels given image input, *i.e.*, $\mathcal{X} \rightarrow \mathcal{Y}$. However, these classification-based approaches lack the ability of generalizing to unseen labels, and need to be retrained when a new label emerges. For example, given a training dataset \mathcal{D} as above, and a test dataset $\mathcal{D}' \equiv \{(\mathbf{x}'_j, \mathbf{y}'_j)\}_{j=1}^{N'}$ where $\mathbf{x}'_j \in \mathcal{X}$ and all test labels are distinct from the training labels in dataset \mathcal{D} , *i.e.*, $\mathbf{y}'_j \in \mathcal{Y}' \equiv \{n+1, \dots, n+n'\}$. The test labels are untrained as $\mathcal{Y} \cap \mathcal{Y}' = \emptyset$. Clearly, without side information about the relationships between labels in \mathcal{Y} and \mathcal{Y}' , it is infeasible to generalize a classification-based method to unseen labels without retraining it.

Fortunately, visual-semantic embedding models [Frome et al., 2013; Norouzi et al., 2014] have been proposed to address this issue for single-label image classification. Instead of learning a mapping from images to the labels ($\mathcal{X} \rightarrow \mathcal{Y}$), it aims to construct a continuous semantic space $\mathcal{S} \equiv \mathbb{R}^e$ which captures the semantic relationship among all labels in $\mathcal{Y} \cup \mathcal{Y}'$, and explicitly learn the embedding function from images to such space, $f : \mathcal{X} \rightarrow \mathcal{S}$. The semantic space \mathcal{S} is constructed such that two labels y and y' are semantically similar if and only if their semantic embeddings $s(y)$ and $s(y')$ are close in \mathcal{S} , where $s(y)$ is the semantic embedding vector of label y in \mathcal{S} . Thus the trained and unseen test labels become related via the semantic space \mathcal{S} .

Once $f(\cdot)$ is learned, it can be applied to a test image \mathbf{x}' to obtain $f(\mathbf{x}')$, and this image embedding vector of \mathbf{x}' is then compared with the unseen label embedding vectors, $\{s(y'); y' \in \mathcal{Y}'\}$, to search for the most relevant test labels. This allows us to generalize the visual-semantic embedding models to unseen labels.

In sum, there are two key components of Visual-Semantic Embedding models: 1) how to construct the continuous semantic space \mathcal{S} of image labels; and 2) how to learn the embedding function $f(\cdot)$. Existing VSE models [Frome et al., 2013; Norouzi et al., 2014; Kiros et al., 2015; Ren et al., 2015a] utilize the text distributed representation, such as Glove [Pennington et al., 2014] or word2vec [Mikolov et al., 2013], to construct the semantic space, which will be introduced in details in later chapters. One the other

hand, existing VSE models were only proposed for joint representation of images with single labels. The embedding function $f(\cdot)$ was learnt by a ranking loss, to encourage the embedding of an image to be closer to its ground truth label than other negative labels, where the loss function is defined as below:

$$L_{\text{VSE}}(\mathbf{x}_i, y_i) = \sum_{y_n \in \mathcal{Y}_i^-} \max(0, m + \|f(\mathbf{x}_i) - s(y_i)\|_{L_2} - \|f(\mathbf{x}_i) - s(y_n)\|_{L_2}), \quad (2.1)$$

where m is the ranking loss margin that we cross-validate, $f(\mathbf{x})$ is the embedding vector of image \mathbf{x} in \mathcal{S} , $s(y)$ is the semantic embedding vector of label y in \mathcal{S} , \mathcal{Y}_i^- denotes the negative labels excluding the ground truth label y_i , *i.e.*, $\mathcal{Y}_i^- = \mathcal{Y}/y_i$.

CHAPTER 3

Text Concept Modeling: Gaussian Visual-Semantic Embedding

3.1 Introduction

In this chapter, we present a novel joint image-text representation model via text concept modeling: Gaussian Visual-Semantic Embedding. And we validate its effectiveness in two multi-modal tasks, *i.e.*, image classification and text-based image retrieval.

As claimed in Chapter 1, joint image-text representation is essential for tasks involving both images and texts, such as image captioning [Mao et al., 2015; Vinyals et al., 2015], text-based image retrieval [Kiros et al., 2015; Rui et al., 1999], image classification [Deng et al., 2009; Gong et al., 2014a], *etc.* In recent years, Visual-Semantic Embedding (VSE) models [Frome et al., 2013; Norouzi et al., 2014; Kiros et al., 2015; Ren et al., 2015a] have shown impressive performance in those tasks, which has been recapped in Chapter 2. In short, by leveraging the semantic information contained in unannotated text data, VSE models explicitly map images into a rich semantic space, with the goal that images of the same category are mapped to nearby locations around the corresponding text label, and text labels are embedded in such a smooth space with respect to some measure of similarity, that is, similar text concepts should be mapped into nearby locations.

Although VSE models have shown remarkable results, representing text concepts as

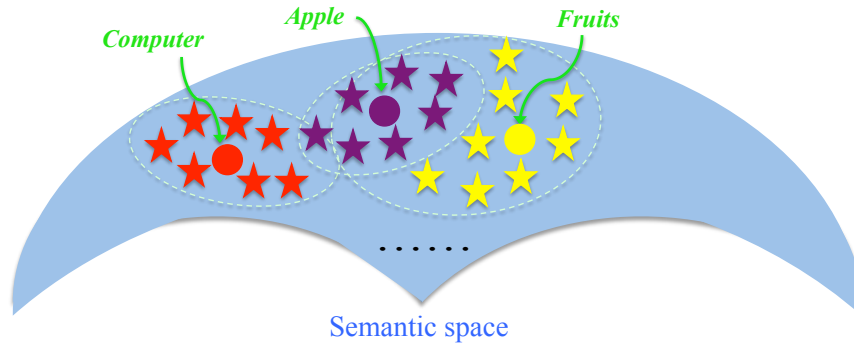


Figure 3.1: Existing Visual-Semantic Embedding models simply represent the text concepts as single points in the semantic space, as indicated by the circular symbols (we use different colors to indicate different categories). We propose the Gaussian Visual-Semantic Embedding (GVSE) model, which however leverages the visual information indicated as the pentagram symbols (pentagram symbols of the same color indicate various images of the same category) to model text concepts as Gaussian distributions. Modeling text concepts as densities innately represents uncertainties of text concepts, and has the benefit of geometric interpretation such as inclusion and intersection.

single points in semantic space carries some important limitations. Firstly, using an embedded vector to represent a text does not naturally express uncertainty about its concept with which the corresponding images may be associated. Even though various images may belong to the same text category, uncertainty is associated with them, which may reflect different aspects of a certain text concept. For instance, as shown in Figure 3.1, the *Computer* images may be from different brands, viewpoints, *etc.* Moreover, it is intrinsically problematic to map all images of a certain text label to a single point in semantic space, which would confuse the embedding function and thus harm the joint representation.

This dissertation advocates moving beyond modeling text concepts as single points to that as *densities* in semantic space. In particular, we explore Gaussian distributions (currently with diagonal covariance), in which the means are learned from unannotated

text data online and variances are learned from the visual image data. Gaussians innately embody uncertainty, and have a geometric interpretation, such as an inclusion or intersection relationships between text concepts, as shown in Figure 3.1. We name the proposed method Gaussian Visual-Semantic Embedding (GVSE) model.

To evaluate the proposed method, we have conducted experiments in two tasks on the large scale MIT Places205 dataset. In the image classification task, our model outperforms the VSE baseline [Frome et al., 2013] by 1-5%. In the task of text-based image retrieval, we illustrate the robustness of our method and the capability in generalizing to untrained texts.

3.2 Background

In order to learn a joint image-text representation, various embedding models that embeds images and labels into a common space have been developed in the literature.

Multi-modal embedding models utilize information from multiple sources, such as images and texts. By leveraging the abundant textual data available on the Internet, several lexically distributed representations of texts have been proposed to capture the semantic meaning among texts, *e.g.*, the word2vec model [Mikolov et al., 2013] and GloVe model [Pennington et al., 2014]. Image-sentence embedding models [Karpathy and Fei-Fei, 2015; Kiros et al., 2015] were proposed for image captioning. Recently, visual-semantic embedding models for image classification were proposed by leveraging the distributed representation of labels, *e.g.*, Frome *et al.* [Frome et al., 2013] and *et al.* Norouzi *et al.* [Norouzi et al., 2014]. Ren *et al.* [Ren et al., 2015a] proposed multiple instance visual-semantic embedding for multi-label image annotation. Vilnis *et al.* [Vilnis and McCallum, 2015] presented Gaussian embedding model for word representation. The main difference between [Vilnis and McCallum, 2015] and our GVSE model is that [Vilnis and McCallum, 2015] is only for word representation learned from

word data alone, while our model is for joint image-text representation.

3.3 Gaussian Visual-Semantic Embedding

As discussed in Chapter 2, here are two key components of Visual-Semantic Embedding models: 1) how to construct the continuous semantic space \mathcal{S} of image labels; and 2) how to learn the embedding function $f(\cdot)$. Now we introduce these two components of our Gaussian Visual-Semantic Embedding model.

3.3.1 Constructing the semantic label space

Distributed representations [Mikolov et al., 2013; Pennington et al., 2014] has shown the capacity to provide semantically meaningful embedding features for text terms (including words and phrases), by learning from unannotated text data from the Internet. This method is able to learn similar embedding vectors for semantically related words because of the fact that those words are more likely to appear in similar semantic contexts.

In this chapter, we utilize the GloVe model [Pennington et al., 2014] to construct a 300-dim text label space \mathcal{S} which embodies the semantic relationship among labels.

3.3.2 Modeling text concepts as Gaussian distributions

Motivated by the success of ranking loss in state-of-the-art visual-semantic embedding [Frome et al., 2013; Norouzi et al., 2014; Ren et al., 2015a], we employ ranking loss to learn the embedding function $f : \mathcal{X} \rightarrow \mathcal{S}$. The intuition is to encourage the embedding

of an image to be closer to its ground truth text label than other negative labels:

$$L_{\text{GVSE}}(\mathbf{x}_i, y_i) = \sum_{y_n \in \mathcal{Y}_i^-} \max(0, m + D(f(\mathbf{x}_i), y_i) - D(f(\mathbf{x}_i), y_n)), \quad (3.1)$$

where m is the ranking loss margin that we cross-validate, $f(\mathbf{x})$ is the embedding vector of image \mathbf{x} in \mathcal{S} , \mathcal{Y}_i^- denotes the negative labels excluding the ground truth label y_i , *i.e.*, $\mathcal{Y}_i^- = \mathcal{Y}/y_i$, and $D(f(\mathbf{x}), y)$ is the distance measure between an image embedding point and a text concept. We will introduce how we compute $D(f(\mathbf{x}), y)$ later.

Existing visual-semantic embedding methods model texts as single points in the semantic space \mathcal{S} , thus $D(f(\mathbf{x}), y)$ in those methods is just the Euclidean distance between $f(\mathbf{x})$ and the text embedding vector $s(y)$. However, as claimed in the introduction, they are limited in representation capacity. Intrinsically, it is beneficial to model text concepts as densities, which can embody the uncertainty of concepts and also have better geometric interpretation.

In this dissertation, we propose to model text concepts as Gaussian distributions with diagonal covariances, *i.e.*, $y_i \sim \mathcal{N}(\boldsymbol{\mu}_i, \Sigma_i)$. Thus, the distance between an image embedding vector $f(\mathbf{x}_i)$ and a text label y_i can be measured by Mahalanobis distance as follows:

$$D(f(\mathbf{x}_i), y_i) = (f(\mathbf{x}_i) - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (f(\mathbf{x}_i) - \boldsymbol{\mu}_i). \quad (3.2)$$

Thus, our GVSE model could be effectively learned by the loss function in Equation 3.1.

3.3.3 Training and inference with GVSE

The learning framework is shown in Figure 3.2. We jointly learn the embedding function $f(\cdot)$ and the text density parameters $\{\boldsymbol{\mu}_i\}_{i=1}^n, \{\Sigma_i\}_{i=1}^n$ in two steps.

Firstly, the mean text embedding vectors $\{\boldsymbol{\mu}_i\}$ are learned using the unannotated text

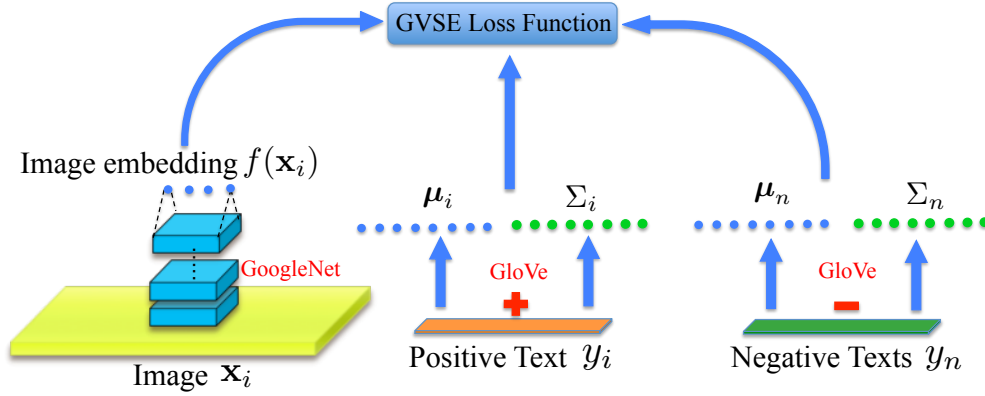


Figure 3.2: Training framework of the proposed Gaussian Visual-Semantic Embedding model.

data online alone, as introduced in Section 4.3.1. Thanks to the similarity between text concepts captured in semantic space \mathcal{S} , our model is able to generalize to unseen text concepts, which will be illustrated in Section 3.4.3. Secondly, we learn $f(\cdot)$ and $\{\Sigma_i\}$ by end-to-end training. Note that we use GoogleNet [Szegedy et al., 2015] to extract image features x_i . On top of the convolutional layers of GoogleNet, we add a fully connected layer to model the embedding function $f(\cdot)$. On the other hand, since Σ_i is a 300×300 diagonal matrix, we use a $300 \times n$ fully connected layer to encode all parameters in $\{\Sigma_i\}_{i=1}^n$, where each column of the weight corresponds to a covariance matrix Σ_i .

Given a trained GVSE model, it is straightforward to do inference on either a query image or a query text, depending on the tasks. From either direction, we map the query and testing entries into the semantic space \mathcal{S} . And Mahalanobis distance between the query and each testing entry can be computed by Equation 3.2. Finally the result is computed based on such distances.

Table 3.1: Image classification results on the MIT Places205 dataset, shown in %.

Approach	mAP@1	mAP@5
1. nearest neighbor search		
VSE with L2 loss	41.33	68.02
DeViSE [Frome et al., 2013]	51.53	82.59
The proposed GVSE	52.38	83.60
2. SVM search		
The proposed GVSE	56.86	84.46

3.4 Experiments

In this section, we report our experiments on image classification and text-based image retrieval, comparing the proposed GVSE model with visual-semantic embedding models.

3.4.1 Dataset and implementation

We test on a large-scale image dataset, MIT Places205 dataset [Zhou et al., 2014], which has 2,448,873 images from 205 scene categories. We follow the train-test split provided in the dataset.

We use Caffe [Jia et al., 2014] to implement our model. The optimization of our network is achieved by Stochastic Gradient Descent with a momentum term of weight 0.9 and with mini-batch size of 100. The initial learning rate is set to 0.1, and we update it with the “steps” policy. A weight decay of 0.0005 is applied.

3.4.2 Experiments on image classification

To quantify the image classification performance, we use $\text{mAP}@k$ as the evaluation metric, which measures the mean average precision of returning the ground truth label within top- k of the prediction list.

The results are shown in Table 3.1. We compare with two baseline models: one is visual-semantic embedding (VSE) model trained with L2 loss, the other is the DeViSE [Frome et al., 2013] model trained with ranking loss. These two models are state-of-the-art visual-semantic embedding models. As we see, with the basic nearest neighbor search in testing, the proposed GVSE model outperforms the best baseline for 0.93% on average, which validates the benefit of modeling text concepts as densities.

Moreover, we train a SVM classifier on top of the embedding representation that GVSE learned, and such classifying technique boosts the performance for 4.48% in $\text{mAP}@1$ and 0.86% in $\text{mAP}@5$. State-of-the-art classification-based method [Zhou et al., 2014] (that uses deep learning and does not use explicitly modeling) reported $\text{mAP}@1$ performance 55.50%. Our performance is 1.36% higher. Note that classification-based methods [Zhou et al., 2014] do not learn a joint representation and are not able to generalize to untrained classes, which is an advantage of our method, as shown in Figure 3.5, Figure 3.6 of Section 3.4.3.

3.4.3 Experiments on text-based image retrieval

Since GVSE model learns a joint embedding representation for both images and texts. We could either search labels given images (as in image classification of Section 3.4.2), or search images given text query. Thus we conduct a qualitative experiment on text-based image retrieval.

Thanks to the semantic space learned from unannotated text data, our model is able

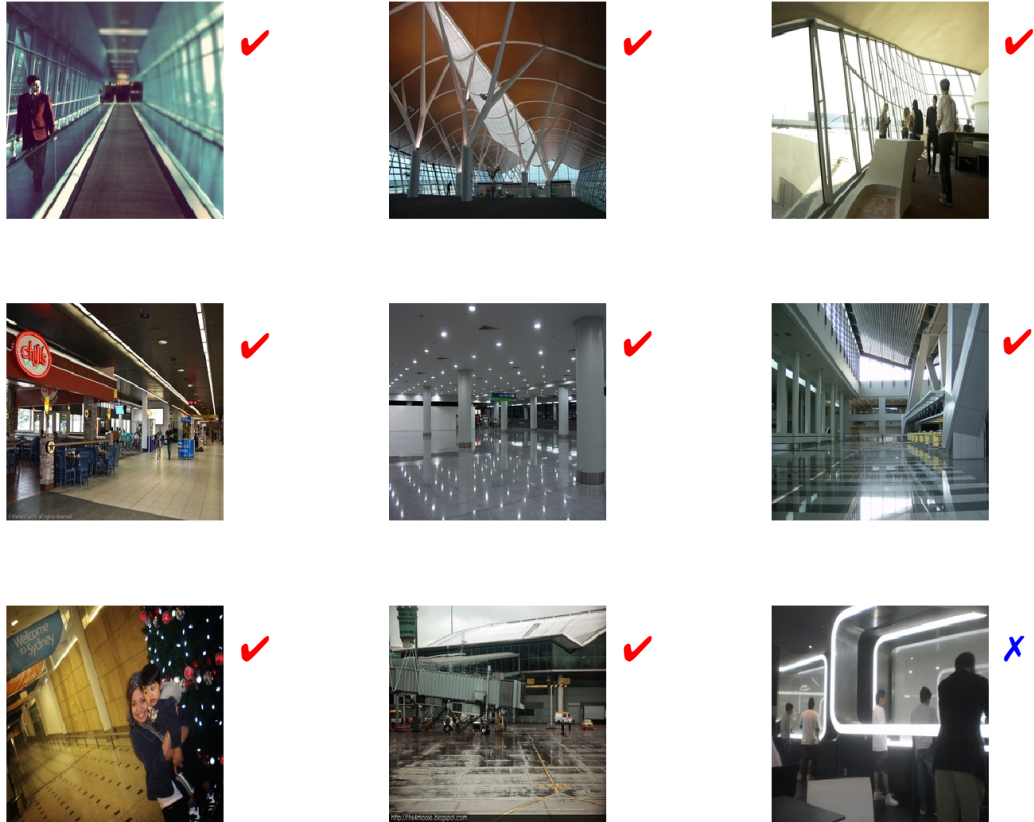


Figure 3.3: *The top-9 image retrieval results of searching trained text “airport terminal”. The incorrect image shares very similar visual appearance.*



Figure 3.4: *The top-9 image retrieval results of searching trained text “kitchen”. The incorrect images belong to “kitchenette” which is a very close concept.*

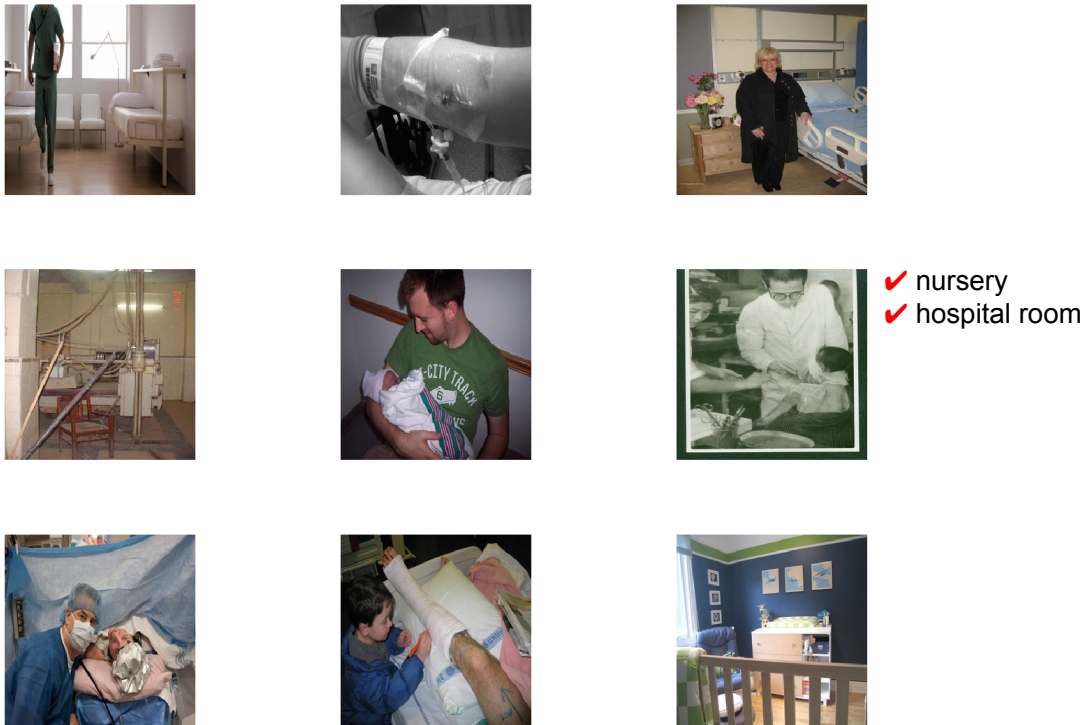


Figure 3.5: *The top-9 retrieval results of searching untrained text “infant”. The returned images belong to “nursery” and “hospital room”, which are conceptually related.*

to search both trained or untrained texts, as discussed in Chapter 2 (searching by untrained texts is known as *zero-shot learning* [Ren et al., 2015a], which is a challenging task.). Figure 3.3 and Figure 3.4 illustrate two examples of trained text searching, while Figure 3.5 and Figure 3.6 show two examples of untrained text searching.

As we see in Figure 3.3 and Figure 3.4, the retrieval results are very robust, except a few incorrect cases, which either share similar visual appearance with the query images or belong to very close text concept of the query. On the other hand, when we search untrained texts as shown in Figure 3.5 and Figure 3.6, the returned results are conceptually very related to the queries. For instance, “*sports*” is a text label that does not appear in the MIT Places205 training dataset. However, using the proposed GVSE model, the retrieved “*stadium*”’s images are very related to “*sports*”.

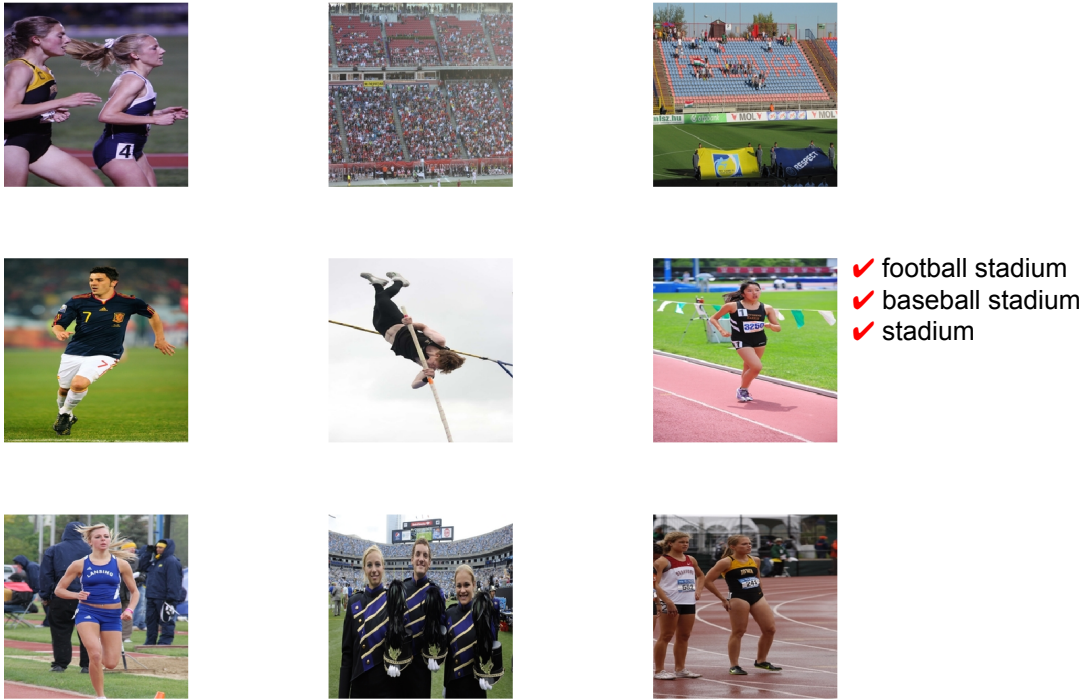


Figure 3.6: *The top-9 retrieval results of searching untrained text “sports”. The returned images belong to “football stadium”, “baseball stadium” and “stadium”, which are conceptually related.*

Both quantitative and qualitative experimental results validate the superiority of the proposed GVSE model in joint image-text representation.

3.5 Summary

In this chapter, we have proposed a novel Gaussian Visual-Semantic Embedding model for joint image-text representation. Instead of modeling text concepts as single points in the semantic space, we move beyond to modeling that as densities. Effective end-to-end training framework and testing techniques have been introduced. Experiments on multi-modal tasks in both directions of images and texts, including image classification and text-based image retrieval, have demonstrated that the proposed method outperforms

existing visual-semantic embedding models with higher accuracy, better robustness, as well as the ability to generalize to untrained text concepts.

From the experimental results above, we have validated the superiority of modeling text concepts as densities over single points. We explored text concept modeling as Gaussian. And for effective training, we constrained the densities to be diagonal Gaussians. A straightforward improvement of our method is to model text concepts as more sophisticated density distributions, such as Gaussians with arbitrary covariances and Gaussian Mixture Models. Another future improvement over the proposed method is in the training process. We learned our model parameters by end-to-end training. However, with more complicated text modelings, such as GMM, iterative training can benefit the convergence.

CHAPTER 4

Image-Level Modeling: Multiple Instance Visual-Semantic Embedding

4.1 Introduction

In this chapter, we present a novel joint image-text representation model via image-level modeling: Multiple Instance Visual-Semantic Embedding. And we validate its effectiveness in two multi-modal tasks, *i.e.*, multi-label image annotation and zero-shot learning on image classification.

Image classification is a fundamental problem in computer vision. Most existing work focuses on single-label image classification [Deng et al., 2009; Krizhevsky et al., 2012; Szegedy et al., 2015; Simonyan and Zisserman, 2015], where each image is assumed to have only one class label. Classic approaches try to categorize images into a fixed set of classes by training a multi-class classifier. However, due to the complex relationships among concepts (*e.g.*, hierarchical, disjoint, *etc*), it is hard to define a perfect classifier encoding all those semantic relations [Deng et al., 2014]. Also, since the set of classes is predefined, such systems need to be re-trained whenever a new visual entity emerges.

To address these shortcomings, visual-semantic embedding models [Frome et al., 2013; Norouzi et al., 2014] were recently proposed, which leverage semantic information contained in unannotated text data to learn semantic relationships among labels, and explicitly map images into a rich semantic space (in such space, certain semantic rela-

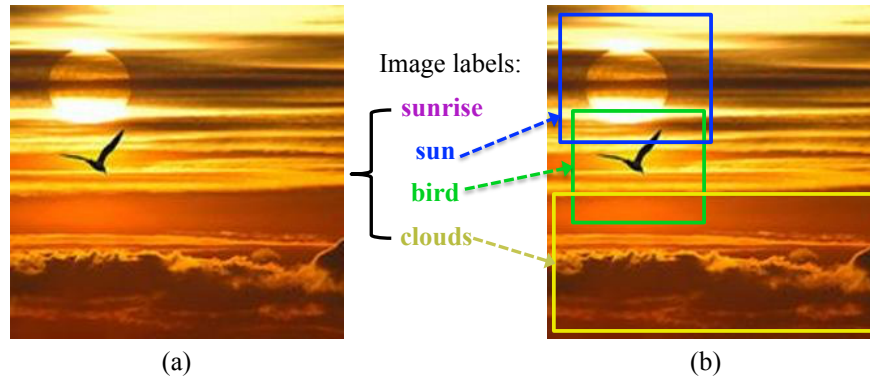


Figure 4.1: (a) An example of an image with multiple labels that are listed in the middle. (b) We observe that different labels may correspond to various image subregions, but not necessarily the whole image, such as the labels clouds, sun, bird, which are associated with the subregions in the bounding boxes.

tionships are encoded, *e.g.*, related labels like *sun* and *sunrise* locate at close positions. And images from these two classes may share some common visual appearance.). By resorting to classification in the semantic space with respect to a set of label embedding vectors, visual-semantic models have shown comparable performance to state-of-the-art visual object classifiers and demonstrated *zero-shot learning* capability, *i.e.*, the ability to predict unseen image categories without training them (which validates the ability to address the shortcomings mentioned above).

Although visual-semantic embedding models have shown impressive results for images with single labels, no attempts have been made on optimizing it for multi-label annotation. It is important to develop such a model due to the following reasons. Firstly, real-world images are often associated with multiple description labels. Multi-label annotation is a practical and challenging task, since the image labels can be diverse, which may describe the image foreground, image background, or the whole image. Secondly, it is nontrivial to extend a single-label visual-semantic embedding model to a multi-label one. The implicit assumption that each label corresponds to the whole image does

not hold for multi-label cases. For a typical multi-label image, some labels may correspond to different image subregions, instead of the whole image. For example, as shown in Figure 4.1, only the scene label *sunrise* corresponds to the whole image, and all other label concepts correspond to specific subregions in the image as shown in the bounding boxes.

Hence, we present a novel Multiple Instance Visual-Semantic Embedding (MIVSE) model for image annotation with multiple labels. Our method characterizes an image-subregion-to-label correspondence by learning an embedding function that maps semantically meaningful image subregions to their corresponding labels in the semantic space. In order to discover those semantically meaningful subregions, we construct a bag of subregion instances using state-of-the-art region-proposal method [Krähenbühl and Koltun, 2014]. And we propose an objective function that models such correspondence with a weighting scheme encoded to optimize the label prediction.

To evaluate our method, we have conducted experiments on two tasks. In the multi-label image annotation task, our model outperforms state-of-the-art method [Gong et al., 2014a] on the NUS-WIDE dataset, where semantically meaningful subregions of each tagged label can be discovered without bounding box supervision (as shown in Figure 4.4 of our experiments). In the task of zero-shot learning, our method outperforms state-of-the-art zero-shot learning method [Frome et al., 2013] in unseen category classification on the MIT Places205 dataset.

4.2 Background

We briefly outline the background of the two tasks that we validate the proposed joint representation on.

4.2.1 Multi-label image annotation

Modeling images and their corresponding textual labels have attracted increasing interest recently in the vision community. Early work in this area focused on learning statistical models based on hand-crafted features [Yang et al., 2009; Perronnin and Dance, 2007; Chen et al., 2012]. For instance, Makadia *et al.* [Makadia et al., 2008] and Guillaumin *et al.* [Guillaumin et al., 2009] proposed nonparametric nearest-neighbor methods to transfer image labels. Weston *et al.* [Weston et al., 2011] proposed WSA-BIE with a WARP loss based on bag-of-words feature. Recently, as the learned image representation using deep convolutional neural network (CNN) has shown superior performance in various vision tasks [LeCun et al., 1990; Krizhevsky et al., 2012; Zeiler and Fergus, 2014; Szegedy et al., 2015; Simonyan and Zisserman, 2015; Ren et al., 2015b; Girshick, 2015], Gong *et al.* [Gong et al., 2014a] applied the CNN architecture to multi-label image annotation problem and achieved nice performance. With the help of additional metadata, methods [Johnson et al., 2015; Hu et al., 2016] further boost the annotation performance. However, since the metadata used in [Johnson et al., 2015; Hu et al., 2016] is not public yet, we do not utilize metadata in our model for fair evaluation. Without metadata, state-of-the-art method is [Gong et al., 2014a], which is classification-based and thus lacks the ability of generalizing to unseen labels. While, our method based on visual-semantic embedding possesses such ability. In this chapter, we propose to characterize a subregion-to-label correspondence for multi-label images. There exist other works which target on modeling a similar correspondence, *i.e.*, Karpathy *et al.* [Karpathy et al., 2014; Karpathy and Fei-Fei, 2015]. However, our method differs from [Karpathy et al., 2014; Karpathy and Fei-Fei, 2015] by the facts that they target on a different task which is image captioning, and the proposed objective functions are significantly different. We will validate the effectiveness of our objective function experimentally in Table 4.2.

It is also important to note the difference between multi-label image annotation and

attribute-based image classification [Akata et al., 2013]. Attributes [Farhadi et al., 2009; Parikh and Grauman, 2011] are commonly used to encode the visual properties of objects. However, attributes are different from labels, since attributes are on object-level while labels are on image-level. For example, in Figure 4.1, *circular* can be an attribute of the object *sun*, but it is not a suitable label of this image.

4.2.2 Zero-shot learning

Zero-shot learning is commonly used to evaluate the generalizing ability of a system, whose goal is to classify images from untrained classes.

Early work [Palatucci et al., 2009; Rohrbach et al., 2011; Mensink et al., 2012] attempted to solve this problem relying on curated source of semantic information of the labels. For instance, Palatucci *et al.* [Palatucci et al., 2009] used a knowledge base to describe all labels. Rohrbach *et al.* [Rohrbach et al., 2011] and Mensink *et al.* [Mensink et al., 2012] used the WordNet hierarchy. Socher *et al.* [Socher et al., 2013] used a richer image representation [Bengio et al., 2003b] and a neural network language model [Coates and Ng, 2011]. Recently, visual-semantic embedding models [Frome et al., 2013; Norouzi et al., 2014] were proposed to leverage semantic representation learned directly from unannotated text data online.

There are various attribute-based zero-shot learning methods in the literature [Lampert et al., 2009; Li et al., 2015]. However, because of the difference between attributes and labels as explained in Section 4.2.1, in this chapter, we only compare to zero-shot learning methods in the context of image annotation.

4.3 Semantic Space and Our Multi-label Baseline

As discussed in Chapter 2, here are two key components of Visual-Semantic Embedding models: 1) how to construct the continuous semantic space \mathcal{S} of image labels; and 2) how to learn the embedding function $f(\cdot)$. Now we introduce these two components of our Multiple Instance Visual-Semantic Embedding model.

4.3.1 Constructing the semantic label space

Distributed representations [Mikolov et al., 2013; Pennington et al., 2014] has shown the capacity to provide semantically meaningful embedding features for text terms (including words and phrases), by learning from unannotated text data from the Internet. This method is able to learn similar embedding vectors for semantically related words because of the fact that those words are more likely to appear in similar semantic contexts.

Thus, we utilize the GloVe model [Pennington et al., 2014] to construct a 300-dim text label space \mathcal{S} which embodies the semantic relationship among labels.

4.3.2 Our embedding baseline model: from single-label to multi-label

The embedding function of visual-semantic models, $f : \mathcal{X} \rightarrow \mathcal{S}$, is learned by defining a loss function. Motivated by the success of ranking loss in state-of-the-art visual-semantic embedding [Frome et al., 2013; Norouzi et al., 2014] and multi-label annotation method [Gong et al., 2014a], we construct a multi-label visual-semantic embedding baseline using ranking loss. The intuition is to encourage the embedding of an image

to be closer to its ground truth labels than other negative labels:

$$L_{\text{rank}}(\mathbf{x}_i, \mathbf{y}_i) = \sum_{y_p \in \mathbf{y}_i^+} \sum_{y_q \in \mathbf{y}_i^-} \max(0, m + \|f(\mathbf{x}_i) - s(y_p)\|_2^2 - \|f(\mathbf{x}_i) - s(y_q)\|_2^2), \quad (4.1)$$

where m is the ranking loss margin that we cross-validate, $f(\mathbf{x})$ is the embedding vector of image \mathbf{x} in \mathcal{S} , $s(y)$ is the embedding vector of label y in \mathcal{S} . \mathbf{y}_i^+ denotes the ground truth label set of image \mathbf{x}_i , *i.e.*, $\mathbf{y}_i^+ = \mathbf{y}_i$, and \mathbf{y}_i^- denotes the negative label set excluding the labels in \mathbf{y}_i , *i.e.*, $\mathbf{y}_i^- = \mathcal{Y}/\mathbf{y}_i$.

4.4 Multiple Instance Visual-Semantic Embedding

Our baseline model in Equation 4.1 is seemingly plausible for multi-label image embedding. However, there is a problem: each image \mathbf{x}_i may correspond to multiple labels in \mathbf{y}_i , and one or more of those labels could be located far away from others in the semantic space \mathcal{S} . As shown on the right of Figure 4.2, label *bird* and *sun* could be distant. Trying to push the embedding of a whole image, $f(\mathbf{x}_i)$, to be close to multiple distant points in \mathcal{S} will confuse the embedding function; in the worst case, the image could be mapped to a near average position of those label embedding vectors, which might correspond to a totally different concept.

The key observation for overcoming this problem is that different image labels often correspond to different subregions in the image. For example, in Figure 4.2, the image on the left has four labels: *sunrise*, *clouds*, *sun*, and *bird*. Among them only *sunrise* corresponds to the whole image and other labels correspond to image subregions shown in the bounding boxes. This motivates us to derive a new idea for multi-label embedding in which one can generate multiple subregion proposals from an image (including the whole image) and use the resulting subregion set to match the labels in the semantic space. This requires a subregion-to-label correspondence to be constructed on-the-fly during the learning process. Let us introduce our MIVSE loss function in stages.

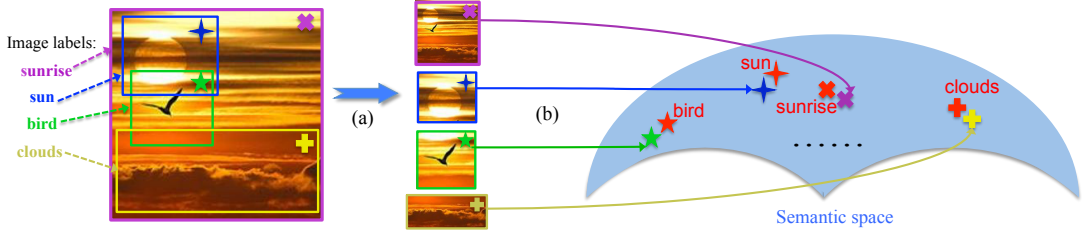


Figure 4.2: *Illustration of our Multiple Instance Visual-Semantic Embedding model, which is composed of two key components: (a) construct image subregion set; (b) establish the subregion-to-label correspondence by embedding semantically meaningful subregions close to their corresponding labels in the semantic space (the red symbols illustrate the embedding of text labels, and symbols of other colors indicate that of different image subregions.). Note that the bounding boxes are for visualization only, and they are not provided in training.*

4.4.1 Modeling subregion-to-label correspondence

Based on the motivation above, we propose Multiple Instance Visual-Semantic Embedding (MIVSE) model to learn the embedding function that characterizes subregion-to-label correspondence, as shown in Figure 4.2.

Inspired by Multiple Instance Learning [Dietterich et al., 1997], we first construct a bag of image subregion instances. In order to interpret each ground truth label, there should be at least one subregion that maps close to it. And the subregion with the closest distance to a certain label is more likely to represent that label. Thus, we define a preliminary loss function of MIVSE as follows:

$$L'_{\text{MIVSE}}(\mathbf{x}_i, \mathbf{y}_i) = \sum_{y_p \in \mathbf{y}_i^+} \sum_{y_q \in \mathbf{y}_i^-} \max(0, m + \min_{c \in \mathcal{C}} \|f(\mathbf{x}_i^c) - s(y_p)\|_2^2 - \min_{c \in \mathcal{C}} \|f(\mathbf{x}_i^c) - s(y_q)\|_2^2), \quad (4.2)$$

where \mathcal{C} is the set of all subregions in image \mathbf{x}_i (we will introduce how to obtain \mathcal{C} later in Section 4.4.3), \mathbf{x}_i^c indicates one subregion of image \mathbf{x}_i , \mathbf{y}_i^+ denotes the ground truth label set, and \mathbf{y}_i^- denotes the negative label set.

4.4.2 Rank-weighted loss

However, one limitation of the preliminary loss function in Equation 4.2 is that it does not explicitly optimize the rank of ground truth labels. In image annotation, the ranking of prediction is essential since we assign labels to the images according to the final prediction rank. Thus, we propose to employ a weighting scheme to optimize such rank. Let us define the rank r_p of a predicted label y_p first:

$$r_p = \sum_{y_t \neq y_p, y_t \in \mathcal{Y}} \mathbb{1} \left(\min_{c \in \mathcal{C}} \|f(\mathbf{x}_i^c) - s(y_t)\|_2^2 \leq \min_{c \in \mathcal{C}} \|f(\mathbf{x}_i^c) - s(y_p)\|_2^2 \right), \quad (4.3)$$

where $\mathbb{1}(\cdot)$ is the indicator function. As we see, given a label y_p , we rank it according to its minimal distance to all image subregions, *i.e.*, by $\min_{c \in \mathcal{C}} \|f(\mathbf{x}_i^c) - s(y_p)\|_2^2$. By optimizing the ranking of labels, we are supposed to encourage ground truth (positive) labels to have smaller matching-distance than the negative labels, *i.e.*, to rank the positive labels on the top of the ranking list. Thus, following [Weston et al., 2011], we give larger penalties to false predictions of ranking positive labels at the bottom. The final loss function of MIVSE is defined as follows:

$$L_{\text{MIVSE}}(\mathbf{x}_i, \mathbf{y}_i) = \sum_{y_p \in \mathbf{y}_i^+} \sum_{y_q \in \mathbf{y}_i^-} w(r_p) \cdot \max(0, m + \min_{c \in \mathcal{C}} \|f(\mathbf{x}_i^c) - s(y_p)\|_2^2 - \min_{c \in \mathcal{C}} \|f(\mathbf{x}_i^c) - s(y_q)\|_2^2), \quad (4.4)$$

where r_p is the rank of a positive label y_p , and $w(\cdot)$ is a weighting function as:

$$w(r) = \begin{cases} 1 & \text{if } r < |\mathbf{y}_i^+|, \\ r & \text{otherwise.} \end{cases} \quad (4.5)$$

Given an image with $|\mathbf{y}_i^+|$ numbers of ground truth labels, if a ground truth label is ranked within top- $|\mathbf{y}_i^+|$, we give small and constant penalty weights to its loss. While if a ground truth label is not ranked top, *i.e.*, $r \geq |\mathbf{y}_i^+|$, we assign much larger weight (linear to the rank). The intuition of this weighting scheme is to push the positive labels to top ranks, thus pushing meaningful subregions to map closer to their corresponding ground truth labels in the semantic space.

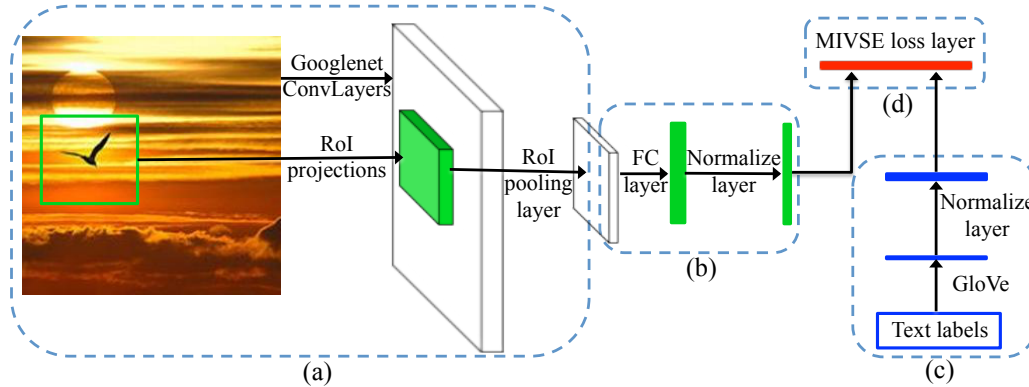


Figure 4.3: The deep network architecture of MIVSE model, composed of 4 components: (a) subregion image features extraction; (b) image features embedding; (c) text features embedding; (d) joint embedding function learning guided by the MIVSE loss layer.

4.4.3 Learning multiple instance visual-semantic embedding

In this section, we introduce the overall MIVSE training architecture and explain how we construct subregion set \mathcal{C} .

4.4.3.1 MIVSE network architecture.

Figure 4.3 illustrates the overall network architecture of our MIVSE model. As discussed in Section 4.3.1, we utilize the GloVe model [Pennington et al., 2014] to extract 300-dim label embedding features $s(y)$. And we adopt the fully convolutional layers of GoogleNet [Szegedy et al., 2015] (including convolution, pooling, and inception layers) to extract the 1024-dim image feature \mathbf{x}_i ¹. To learn the embedding from image to semantic space, *i.e.*, $f : \mathcal{X} \rightarrow \mathcal{S}$, a fully connected layer is used following the image feature output. Two L_2 normalize layers are added on both image and text embedding vectors, which makes the embeddings of different modalities comparable (we tested

¹In principal, MIVSE can be applied to any image, text representations.

MIVSE without the normalize layers and the results were worse). Finally, we add the MIVSE loss layer on the top to guide the training.

In our model, we need to extract image features for various image subregions \mathbf{x}_i^c of each single image. For efficiency, we follow the Fast RCNN [Girshick, 2015] scheme. Namely, given an image \mathbf{x}_i and the regions of interests (RoI) \mathcal{C} , we pass the image through the fully convolutional network once, and all subregions $c \in \mathcal{C}$ are pooled into a fixed-size feature map to obtain the subregion feature vectors \mathbf{x}_i^c .

4.4.3.2 Subregion set construction.

Note that we do not have bounding box annotations for training, thus a key problem of our MIVSE model is how to construct the image subregion set \mathcal{C} . Inspired by the recent region-proposal-based methods [Girshick et al., 2014a; Girshick, 2015] in object detection, we construct this set using a state-of-the-art region-proposal method [Krähenbühl and Koltun, 2014] followed by post-processing.

Semantically meaningful subregions of an image do not necessarily contain objects. For instance, label *clouds* in Figure 4.1 corresponds to a background region. Thus the object-proposal methods that focus on foreground are not suitable here, especially the ones that depend on edge information. We adopt Geodesic Object Proposals [Krähenbühl and Koltun, 2014] since we empirically find that it covers both foreground and background regions well. Then, we post-process the proposals of [Krähenbühl and Koltun, 2014] to discard regions of too small sizes or extreme aspect ratios (in experiments we constraint the subregions' side length to be at least 0.3 of the image and the extreme aspect ratio to be 1:4 or 4:1). Finally, the original image is included into the set since some labels may correspond to the whole image, such as *sunrise* in Figure 4.1.

4.4.3.3 MIE subregion set refinement

After constructing the subregion set above, there are around 100 subregions per image on average. But among them only very few correspond to the ground truth labels. Thus, in order to save computational time, for each positive label y_j , we do not need to select its corresponding image region among the entire subregion set \mathcal{C} . Instead, we refine the subregion set for all positive labels by evaluating all the subregions in \mathcal{C} with a ranking loss baseline model discussed in Equation 4.1 of Section 4.3.2 and selecting the top-10 subregions as the refined set \mathcal{C}_j for positive label y_j . Accordingly, in our MIE loss function defined in Equation 4.4, $\min_{c \in \mathcal{C}} D_{f(x_i^c), y_j}$ is replaced by $\min_{c \in \mathcal{C}_j} D_{f(x_i^c), y_j}$.

4.4.4 Inference with multiple instance visual-semantic embedding

Given a trained MIVSE model, now we introduce how to do inference on a new test image \mathbf{x}' . Firstly, the subregion set \mathcal{C}' is constructed using [Krähenbühl and Koltun, 2014]. Then, we pass \mathbf{x}' and \mathcal{C}' through our MIVSE network in Figure 4.3 to obtain the subregion embedding vectors, $\{f(\mathbf{x}'^c)\}_{c \in \mathcal{C}'}$. Then, for all testing labels $y' \in \mathcal{Y}'$, the distances between image \mathbf{x}' and y' are computed by $\min_{c \in \mathcal{C}'} \|f(\mathbf{x}'^c) - s(y')\|_2^2$. Thus, for image \mathbf{x}' there is a ranking list of label prediction according to such distances. In addition, given a predicted label y^* , we can locate the corresponding semantically meaningful image subregion c^* via, $c^* = \operatorname{argmin}_{c \in \mathcal{C}'} \|f(\mathbf{x}'^c) - s(y^*)\|_2^2$.

Intuitively, in the inference stage, we embed image subregions into the semantic space. If there is a subregion close enough to a text label, we reckon that this label is associated with the image since there is a corresponding subregion that can interpret the label concept, thus such label is assigned to the image.

4.5 Experiments

In this section, we report our experiments on multi-label image annotation and zero-shot learning, comparing with a number of state-of-the-art methods [Gong et al., 2014a; Frome et al., 2013].

4.5.1 Implementation

We use Caffe [Jia et al., 2014] to implement our model. The optimization of our network is achieved by Stochastic Gradient Descent with a momentum term of weight 0.9 and with mini-batch size of 100. The initial learning rate is set to 0.1, and we update it with the “steps” policy. A weight decay of 0.0005 is applied.

4.5.2 Experiments on multi-label image annotation

4.5.2.1 Dataset.

In the task of multi-label image annotation, Gong *et al.* [Gong et al., 2014a] reported state-of-the-art performance. We follow [Gong et al., 2014a] to test on one of the largest public multi-label image dataset, NUS-WIDE [Chua et al., 2009]. This dataset contains 209,347 images from Flickr with 81 ground-truth labels. And we follow the train-test split of [Gong et al., 2014a] to use a subset of 150,000 images for training and the rest for testing.

4.5.2.2 Evaluation metric.

For fair comparison, we adopt the 5 metrics used in [Gong et al., 2014a]. Specifically, for each image, we annotate it with the k highest-ranked labels and compare the as-

signed labels with its ground-truth. Firstly, we compute the recall and precision for each label, and report the per-label recall $Rec_L = \frac{1}{T} \sum_{i=1}^T \frac{N_i^c}{N_i^g}$, and the per-label precision $Prec_L = \frac{1}{T} \sum_{i=1}^T \frac{N_i^c}{N_i^p}$, where T is the total number of labels, N_i^c is the number of correctly annotated images for label i , N_i^g is the number of ground-truth labeling for label i , and N_i^p is the number of predictions for label i . We also report the overall recall $Rec_A = \frac{\sum_{i=1}^T N_i^c}{\sum_{i=1}^T N_i^g}$, and the overall precision $Prec_A = \frac{\sum_{i=1}^T N_i^c}{\sum_{i=1}^T N_i^p}$. Finally, the percentage of recalled labels in all labels is evaluated, denoted as N_+ . Evaluating these 5 metrics makes the evaluation less biased and more thorough.

4.5.2.3 Upper bound.

As discussed above, we assign the top k labels to each image and measure performance. However, many images do not have exactly k ground-truth labels (the average number of ground truth labels per image is approximately 2.4); this implies that no method can achieve unit precision and recall. To estimate upper bounds of these 5 metrics, we define a perfect method, in which we assume that the ground truth of test set is given: for each test image, when the number of ground-truth labels is larger than k , we randomly pick k ground-truth labels as the result; when the number of ground-truth labels is smaller than k , we pick other labels to supplement the ground truth labels as the result. This method provides the “upper bound” performance since it predicts the maximum number of top- k labels to be correct, which is shown in Table 4.1.

4.5.2.4 Quantitative results.

We compare our method with state-of-the-art method on NUS-WIDE [Gong et al., 2014a] using $k = 3$ and $k = 5$, respectively. Table 4.1 shows the results.

Since the original results reported in [Gong et al., 2014a] were based on AlexNet [Krizhevsky et al., 2012] while our model uses GoogleNet [Szegedy et al., 2015]. For

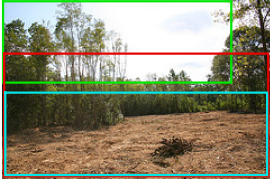
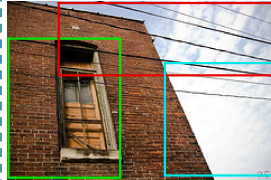
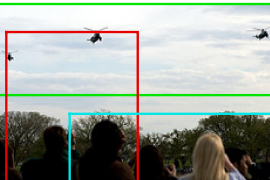
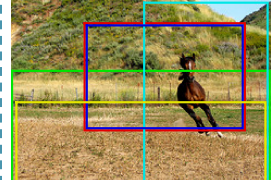
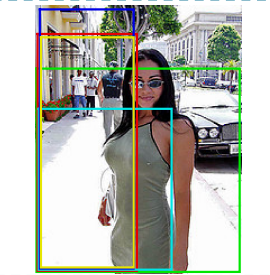

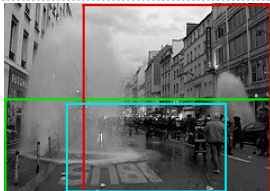
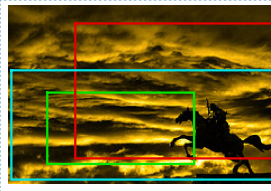
Image	Predictions	GT	Image	Predictions	GT
	<ul style="list-style-type: none"> — Sky — Plants — Grass 	Plants Sky		<ul style="list-style-type: none"> — Window — Buildings — Sky 	Clouds Sky Window
	<ul style="list-style-type: none"> — Sky — Clouds — Person 	Clouds Person Sky		<ul style="list-style-type: none"> — Grass — Animals — Sky — Valley — Plants 	Animals Grass Horses Running
	<ul style="list-style-type: none"> — Person — Buildings — Sky — Street — Road 	Buildings Cars Person Road		<ul style="list-style-type: none"> — Sky — Grass — House — Buildings — Clouds 	Clouds House Sky Water
	<ul style="list-style-type: none"> — Street — Buildings — Road 	Clouds Protest Window		<ul style="list-style-type: none"> — Sky — Clouds — Sunset 	Clouds Nighttime

Figure 4.4: The image annotation results using MIVSE. The predicted labels are listed according to the ranking (we show top-3 predictions if the number of ground truth labels is smaller than or equal to 3, and show top-5 predictions otherwise.). The semantically meaningful subregions of predicted labels are shown in bounding boxes of the same color indicated with the labels. The ground truth labels (GT) are listed according to alphabetic order. The last row shows examples where the predicted annotations are reasonable even they are not included in GT. Better viewed in color.

Table 4.1: Image annotation results on NUS-WIDE shown in %, with $k = 3$ and $k = 5$ annotated labels per image, respectively. See text for the definition of “Upper bound”.

Approach	Rec_L	$Prec_L$	Rec_A	$Prec_A$	N_+
1. $k = 3$					
CNN + Ranking[Gong et al., 2014a]	26.83	31.93	58.00	46.59	95.06
CNN + WARP[Gong et al., 2014a]	35.60	31.65	60.49	48.59	96.29
Upgraded [Gong et al., 2014a]	37.81	35.23	62.02	50.91	98.77
Our model	40.15	37.74	65.03	52.23	100.00
Upper bound	97.00	44.87	82.76	66.49	100.00
2. $k = 5$					
CNN + Ranking[Gong et al., 2014a]	42.48	22.74	72.78	35.08	97.53
CNN + WARP[Gong et al., 2014a]	52.03	22.31	75.00	36.16	100.00
Upgraded [Gong et al., 2014a]	55.27	25.93	78.01	38.04	100.00
Our model	59.81	28.26	80.94	39.00	100.00
Upper bound	99.57	28.83	96.40	46.22	100.00

fair comparison, we reimplement their model using GoogleNet, named as Upgraded [Gong et al., 2014a] in Table 4.1. Overall, our model outperforms state-of-the-art results reported in [Gong et al., 2014a] by 4.51% averaged over all metrics for $k = 3$, and 4.50% for $k = 5$. And our model outperforms the upgraded state-of-the-art [Gong et al., 2014a] by 2.08% for $k = 3$ and 2.15% for $k = 5$.

4.5.2.5 Qualitative results.

Our method can discover semantically meaningful image subregion for each label by modeling the subregion-to-label correspondence.

Figure 4.4 shows several sample results on image annotation, as well as the visualization of corresponding subregions for the predicted labels, as indicated by the bounding boxes. As we can see from the figure, the predicted labels are associated with subregions of reasonable semantics. For example, *Sky* and *Window* in the first row, *Person* and *Animals* in the second row, *Road* and *Grass* in the third row, *etc.*, are reasonably discovered using our model. There are also a few annotation errors or inaccurate bounding boxes. For instance, the right image in the second row is mistakenly annotated with *Plants*. But if we look at the bounding box of *Plants*, the subregion interprets the label concept well. It is important to note that our task is not detection and no bounding box annotation is used in the training. Thus, some objects are not tightly localized by the bounding boxes, *e.g.*, *Animals* of the right image in the second row.

4.5.2.6 Diagnostic experiments.

Our MIVSE model learns an embedding function by encoding the subregion-to-label correspondence with a rank-weighting scheme, and the subregion set is constructed by a region-proposal method.

To better justify the contribution of our method in several key components, we conduct diagnostic experiments to compare our method with its four variants, including: (i). “Our ranking loss baseline” which uses ranking loss objective as shown in Equation 4.1 that maps the whole image to each of its labels; (ii). “MIVSE w/o rank-weighting” that encodes subregion-to-label correspondence into the ranking loss objective as in Equation 4.2, while not including rank-weighting in Section 4.4.2; (iii). “MIVSE w. manual subregions”: instead of using region-proposal method [Krähenbühl and Koltun, 2014], we manually construct the subregion set by selecting subregions with minimum side length as 2, in the 4×4 rigidly defined image grid (totally 36 subregions), and rank-weighting is not included; (iv). “MIVSE w. hinge loss” that replaces the ranking loss of MIVSE with hinge loss, while keeps the subregion-to-label correspondence

Table 4.2: Image annotation results of MIVSE and its four variants on NUS-WIDE shown in %, with $k = 3$ and $k = 5$ annotated labels per image, respectively.

Approach	Rec_L	$Prec_L$	Rec_A	$Prec_A$	N_+
1. $k = 3$					
Our ranking loss baseline	31.59	34.75	60.26	49.17	98.77
MIVSE w/o rank-weighting	38.90	37.87	63.12	51.55	98.77
MIVSE w. manual subregions	34.71	35.92	61.87	50.53	98.77
MIVSE w. hinge loss	28.51	32.63	57.18	47.09	95.06
MIVSE full model	40.15	37.74	65.03	52.23	100.00
2. $k = 5$					
Our ranking loss baseline	50.25	26.08	75.62	36.94	98.77
MIVSE w/o rank-weighting	57.79	28.19	79.16	38.14	100.00
MIVSE w. manual subregions	53.92	26.83	76.81	37.78	100.00
MIVSE w. hinge loss	45.98	21.72	71.86	35.10	96.30
MIVSE full model	59.81	28.26	80.94	39.00	100.00

and rank-weighting, whose loss function is defined as $L_{\text{hinge}}(\mathbf{x}_i, \mathbf{y}_i) = \sum_{y_p \in \mathbf{y}_i^+} w(r_p) \cdot \max(0, m + \min_{c \in \mathcal{C}} \|f(\mathbf{x}_i^c) - s(y_p)\|_2^2)$.

1). *Importance of modeling subregion-to-label correspondence:* As shown in Table 4.2, “MIVSE w/o rank-weighting” outperforms “Our ranking loss baseline” by 3.13% averaged over all metrics for $k = 3$ and 3.12% for $k = 5$, which validates the benefit of modeling subregion-to-label correspondence using the proposed method.

2). *Importance of rank-weighting:* By adding a rank-weighting scheme to the loss in Equation 4.4, our “MIVSE full model” further boost the performance of “MIVSE w/o rank-weighting” by 0.99% for $k = 3$ and 0.95% for $k = 5$, which validates the

contribution of rank-weighting in our objective function.

3). *Importance of subregion set construction*: “MIVSE w/o rank-weighting” using automatic region-proposals outperforms “MIVSE w. manual subregions” that relies on manually generated subregions by 1.68% for $k = 3$ and 1.59% for $k = 5$. Thus, the effectiveness of constructing subregions using region-proposal method is validated.

4). *Importance of developing based on ranking loss*: Hinge loss was widely used in the literature [Karpathy et al., 2014; Karpathy and Fei-Fei, 2015] for image classification. However, for the problem of visual-semantic embedding with multiple labels, it tends to be sensitive to the noisy labels [Weston et al., 2011; Gong et al., 2014a; Frome et al., 2013]. We have validated the superiority of ranking loss by comparing “MIVSE full model” with “MIVSE w. hinge loss”. The performance of the variant decreases significantly by 6.94% for $k = 3$ and 7.41% for $k = 5$.

Overall, our method outperforms state-of-the-art methods in annotating multi-label images, and the corresponding subregions of predicted labels can be located. Moreover, our MIVSE model possesses the generalizing ability to make correct predictions for unseen labels, which will be validated in the following section.

4.5.3 Experiments on zero-shot learning

4.5.3.1 Dataset.

In order to validate the generalizing ability of MIVSE, we conduct experiments on zero-shot learning. As discussed in Section 4.2.2, because of the difference between attributes and labels, we only compare to annotation-based zero-shot learning methods. Unfortunately, most zero-shot learning datasets in the literature are attribute-based. And others are constructed for single-label image zero-shot learning. Thus, we have to reconstruct a zero-shot learning dataset for multi-label images.

We reconstruct such a dataset based on the NUS-WIDE dataset [Chua et al., 2009] and the MIT Places205 dataset [Zhou et al., 2014]. The NUS-WIDE dataset is used for model training, and we test the learned model on the validation set of the MIT Places205 dataset. In MIT Places205, there are 205 classes in total, and for each class there are 100 validation images. We exclude images from the following 8 classes: *bridge*, *castle*, *harbor*, *mountain*, *ocean*, *sky*, *tower*, and *valley* since they are included in NUS-WIDE, which results in 197 test classes. In order to evaluate the consistency of the generalizing ability to different scales, we conduct two experiments on different test sets. One is the whole validation set of 197 classes as above, and the other only contains images from 100 randomly selected classes. It is important to note that in our dataset setup, some test labels may be partially overlapped with some training labels in texts. Previous zero-shot learning of single-label images [Frome et al., 2013] adopted similar setup strategy in their experiments. Thus, we follow such setup. For instance, *airport terminal* is a test label, *airport* is a training label, and they partially overlap. However, they have totally different concepts. Thus, *airport terminal* is considered to be unseen even trained with *airport*.

4.5.3.2 Evaluation metric and comparing methods.

To quantify the performance, we use $\text{mAP}@k$ as the evaluation metric, which measures the mean average precision of annotating the ground truth label within the top- k prediction.

There is seldom zero-shot learning approach for multi-label images. Classification based multi-label image annotation methods do not possess the ability of generalizing to unseen labels; and previous visual-semantic embedding models are developed for single-label images only. Thus we compare the proposed MIVSE model with our ranking loss baseline as shown in Equation 4.1 and the DeVISE model [Frome et al., 2013]. DeVISE is for single-label zero-shot learning. In order to adjust it to fit the

Table 4.3: Zero-shot learning results on the MIT Places205 dataset, shown in %.

Approach	mAP@1	mAP@2	mAP@5	mAP@10
1. 100 unseen classes				
DeViSE [Frome et al., 2013]	1.27	1.98	3.81	6.08
Our ranking loss baseline	8.24	12.61	21.57	31.69
Our full model	9.12	14.07	23.49	34.11
2. 197 unseen classes				
DeViSE [Frome et al., 2013]	1.31	2.25	3.62	5.40
Our ranking loss baseline	6.53	10.18	18.92	28.17
Our full model	7.14	11.29	20.50	30.27

multi-label case, we duplicate each multi-label image as multiple single-label images, and train DeVISE following the original single-label setting.

4.5.3.3 Quantitative results.

As shown in Table 4.3, DeVISE [Frome et al., 2013] performs inferior results because it is inherently designed for single-label zero-shot learning. Separately training it with duplicated images will confuse the embedding learning. Our MIVSE model outperforms the ranking loss baseline by 1.67% averaged over all metrics in part 1 and by 1.35% in part 2, which further validates the benefit of modeling subregion-to-label correspondence. According to the results in part 1 and part 2 of Table 4.3, our MIVSE model shows consistent generalizing ability.


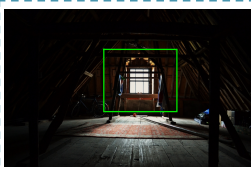


Image	MIVSE model	Ranking loss baseline	Image	MIVSE model	Ranking loss baseline
	clothing store airport terminal rice paddy cemetery construction site	cemetery Inn, outdoor dorm room baseball field chalet		cemetery bedroom attic dinette, home clothing store	dinette, home kitchenette cemetery boat deck highway
	apartment building, outdoor clothing store driveway bridge vegetable garden	mausoleum shed driveway vegetable garden harbor		rope bridge bedroom train station, platform driveway boat deck	bedroom dorm room rope bridge kitchenette driveway

Figure 4.5: Zero-shot learning results on Places205 images using our MIVSE model and the ranking loss baseline, respectively. The correctly predicted labels are shown in blue (note that there is only one ground truth label for each image in Places205.). In each image, the semantically meaningful image subregion of each correctly predicted label using MIVSE is shown in green bounding box.

4.5.3.4 Qualitative results.

Figure 4.5 shows several sample results of zero-shot learning. The two columns on the right of each image show the top-5 label predictions of our MIVSE model as well as our ranking loss baseline.

As shown in Figure 4.5, our MIVSE model correctly predicts all ground truth labels shown in blue. In addition, using our model, the semantically meaningful subregion associated with each predicted labels is located, as shown in green bounding boxes. As we can see, the localized image subregions interpret the label concepts reasonably well. For instance, in the upper right image, test label *attic* is semantically close to the training label *window*. Thus, based on the concept of *window* already learned in training and a well-located *window*-like subregion, our MIVSE model can utilize the relationship in the semantic space to transfer a learned concept of *window* to assist the prediction of an unseen label *attic*. This suggests that the subregion-to-label correspondence of our

method, which helps identify semantically meaningful subregions in image, can benefit the classification of unseen labels due to the more precise interpretation of concepts.

4.6 Summary

In this chapter, we have proposed a novel multi-label visual-semantic embedding model for image annotation with multiple labels. Instead of embedding a whole image into the semantic space, our model learns an embedding function that characterizes the subregion-to-label correspondence, which discovers and maps semantically meaningful image subregions to the corresponding labels. Experimental results on two challenging tasks, *i.e.*, multi-label image annotation and zero-shot learning, have demonstrated that the proposed method achieves superior performance over state-of-the-art methods on both tasks and possesses the generalizing ability to make correct predictions for unseen labels.

CHAPTER 5

Object-Level Modeling: Scene-Domain Active Part Models

5.1 Introduction

In this chapter, we present a novel object representation model: Scene-Domain Active Part Models. And we validate its effectiveness in two fine-grained tasks, *i.e.*, object and parts detection as well as pose and viewpoint estimation.

Object representation is a key problem in computer vision. Coarse-grained representation, such as detecting objects by bounding boxes [Felzenszwalb et al., 2010], is important for tasks such as object tracking [Breitenstein et al., 2009] and scene understanding [Hoiem et al., 2008; Lai et al., 2010]. For example, deep learning approaches [Girshick et al., 2014a; Zhu et al., 2015], based on Convolutional Neural Networks, have validated their ability to extract strong image features and obtain such coarse-grained representation. Moreover, fine-grained representation, such as locating object parts in 2D image-domain and 3D scene-domain, is helpful for further applications such as action analysis [Yang et al., 2010] and human-computer interaction [Ren et al., 2011d,b, 2013b, 2011c, 2013a, 2011a]. For example, part-based models [Azizpour and Laptev, 2012; Yang and Ramanan, 2012] have demonstrated elegant performance in obtaining fine-grained representation.

Hence, we are interested in enhancing part-based models for fine-grained object rep-

resentation. Although various part-based object models have been developed in the literature (*e.g.*, [Felzenszwalb and Huttenlocher, 2005; Sun and Savarese, 2011; Azizpour and Laptev, 2012; Yang and Ramanan, 2012; Chen et al., 2014]) and demonstrated elegant performance in obtaining fine-grained object representation, it is still far from satisfactory to robustly represent generic objects under significant “geometric variations” (this term refers to camera viewpoint changes, non-rigid deformations, intra-class variations, and occlusions in this dissertation). We observe that one main reason for this arises from the fact that in most of the existing applications of generic objects, the available training data are based on 2D images¹. In such a scenario, it is natural that one resorts to modeling objects in the 2D image-domain (*e.g.*, by a collection of 2D parts deforming around the corresponding part anchor positions) and the 3D information is discarded.

However, by modeling part deformations in the 2D image-domain, it is actually difficult to well-capture the important statistics on geometric properties of an object, due to the fact that those “geometric variations” can cause very complicated 2D variations (because of the 3D-2D projection) and make such geometric properties highly complex to be described. For instance, human arms can be foreshortened to varying sizes in different viewpoints. Furthermore, a richer description of an object in the 3D scene-domain, is increasingly in demand for further applications such as action detection, scene understanding, *etc.* For instance, it is beneficial to have the 3D part localization and camera viewpoint for scene understanding.

Accordingly, we are particularly interested in proposing such an even finer-grained object representation that can better capture objects’ geometric statistics and provides richer object representations in 3D. One important observation is that if we can characterize and learn such statistics directly from the 3D scene-domain, we will be able to

¹Only for some specific objects such as human body, human hand, car, bed, *etc.*, their models have been built in 3D by learning from CAD data or depth data (*e.g.*, [Lim et al., 2014; Fidler et al., 2012]), due to the data availability.

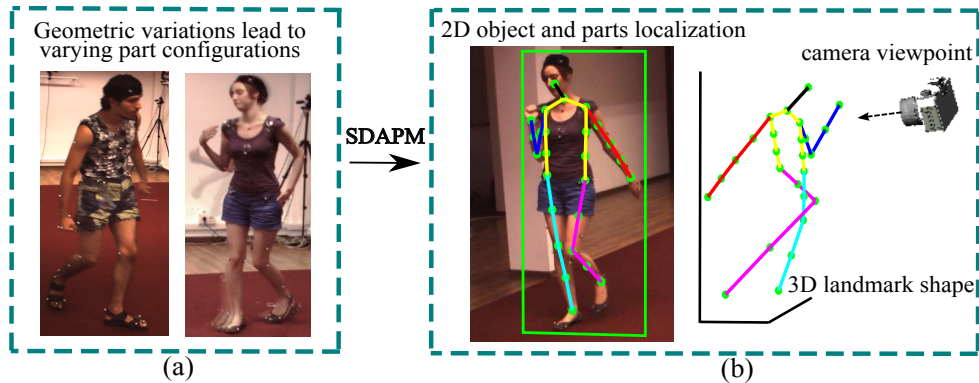


Figure 5.1: (a) A motivation of SDAPM: geometric variations lead to varying part configurations. Thus, by modeling in the scene-domain, SDAPM can better capture objects’ geometric statistics and provides richer object descriptions, including 2D parts localization, 3D landmark shape as well as camera viewpoint estimation, in addition to the 2D object bounding box, as shown in (b).

remove the viewpoint and non-rigid variations, and also obtain the 3D representation of objects. Moreover, recent progress in non-rigid structure-from-motion techniques [Akhter et al., 2008; Dai et al., 2012] provides an effective way to learn such 3D geometric statistics from 2D images. This motivates us to develop a part-based object model that characterizes the geometric variations directly in 3D scene-domain by using 2D training data alone.

Thus, in this chapter, we present Scene-Domain Active Parts Models (SDAPM), as shown in Figure 5.1. Our approach reconstructs and characterizes the 3D geometric statistics between object parts in the scene-domain by learning from 2D training data in the image-domain. And on top of this, we model such statistics together with the local appearance with occlusions. The main contributions of the proposed model are two-fold: firstly, we propose a compact and robust part-based representation for objects under geometric variations, *e.g.*, viewpoint changes, non-rigid deformations, and occlusions, by modeling active parts in the 3D scene-domain; and secondly, our method

provides a fine-grained representation of object, including 2D object and parts localization, 3D landmark shape and camera viewpoint estimation.

We have conducted experiments on various tasks. In the task of object and parts detection on PASCAL VOC 2010 dataset, our method boosts the performance of previous part-based models, *e.g.*, [Felzenszwalb et al., 2010; Bourdev et al., 2010; Parkhi et al., 2011; Azizpour and Laptev, 2012; Fidler et al., 2013; Yang and Ramanan, 2012], with three different types of features: HOG feature, Segmentation feature, and Convolutional Neural Networks (CNN) feature. In the task of 2D, 3D landmark shape and camera viewpoint estimation on Human3.6M dataset and PASCAL VOC 2007 car dataset, our method outperforms state-of-the-art methods [Hejrati and Ramanan, 2012; Arie-Nachimson and Basri, 2009] that are also learned from 2D training data alone.

5.2 Background

In the common scenario where the training data are based on 2D images, existing object representation methods usually resort to modeling objects directly in the 2D image-domain. Two main strategies have been adopted.

One line of work aims at providing coarse-grained representation, *i.e.*, detecting the objects with bounding boxes. Some work focus on developing more effective modeling and inference schemes. For example, Felzenszwalb *et al.* [Felzenszwalb et al., 2010] proposed the elegant Latent SVM framework. Bourdev *et al.* proposed the Poselets model [Bourdev et al., 2010]. Some other work focus on developing stronger image features. For instance, Dalal *et al.* [Dalal and Triggs, 2005] presented HOG descriptors for object representation. Chen *et al.* [Chen et al., 2010] proposed to combine Bag-of-Features with HOG features in a Latent SVM framework. Image segmentation and region-level cues have also been explored for objects detection (*e.g.*, [Fidler et al., 2013; Chen et al., 2014; Parkhi et al., 2011; Wang et al., 2013; Trulls et al., 2014]).

Recently, deep learning approaches, based on Convolutional Neural Networks (CNN), have validated their ability to extract strong image features and achieved state-of-the-art performance on the task of image classification [LeCun et al., 1989; Krizhevsky et al., 2012] and object detection [Girshick et al., 2014a; Zhu et al., 2015]. However, those deep learning approaches do not explicitly model object composition, thus throwing away useful fine-grained high-level part relationships for further applications.

Another line of work proposes to represent objects with fine-grained representation, *e.g.*, the localization of parts. Azizpour and Laptev [Azizpour and Laptev, 2012] proposed strongly supervised paradigm for part-based models to locate object parts. Sun and Savarese [Sun and Savarese, 2011] proposed a coarse-to-fine structure for joint object detection and pose estimation. Yang and Ramanan [Yang and Ramanan, 2012] presented the mixture-of-parts structure for pose estimation. Chen *et al.* [Chen et al., 2014] proposed to model objects with complete graph structure. And grammar models have also been explored in [Girshick et al., 2011]. Various methods have utilized the CNN feature in part-based models in order to leverage the discriminative power of CNN feature and the fine-grained modeling of part-based models [Wan et al., 2014; Savalle et al., 2013; Girshick et al., 2014b].

Despite the potential shown already, these 2D models cannot well-capture the important statistics on the geometric properties of objects under significant viewpoint changes, non-rigid deformations and occlusions, which substantially limits the performance of such models. Since it is useful to model objects in the 3D scene-domain, various 3D part-based models have been developed by modeling the 3D properties of an object directly in the 3D scene-domain using 3D training data. For instance, Fidler *et al.* [Fidler et al., 2012] and Shrivastava *et al.* [Shrivastava and Gupta, 2013] proposed 3D deformable part models for object detection which used depth data for training. 3D CAD data were utilized in [Lim et al., 2014; Pepik et al., 2012] to learn object models for detection and pose estimation. However, most 3D part-based models necessitate

3D data for training, which restrict the use of such models only to a small number of specific objects such as human body, hand, motorcycle, bed, *etc.*

We propose to model objects in the scene-domain by using 2D training data alone. In addition to the 2D representation provided by coarse-grained models and previous fine-grained models, our method provides additional finer-grained representation in 3D, including 2D object and parts localization, 3D landmark shape and camera viewpoint. There exist several works which have similar setting as ours, *e.g.*, the methods proposed by Hejrati *et al.* [Hejrati and Ramanan, 2012] and Nachimson *et al.* [Arie-Nachimson and Basri, 2009] can obtain 2D and 3D object representation from 2D images. However, the major difference between [Hejrati and Ramanan, 2012; Arie-Nachimson and Basri, 2009] and our method is that they first model objects in the image-domain and then reconstruct the 3D landmark shape, via two separate stages, while our method models objects directly in the 3D scene-domain and the 3D landmark shape is recovered via a unified process.

5.3 Scene-Domain Active Part Models

We start the presentation of our model with a preliminary on 2D part-based models, which have been widely and successfully applied in fine-grained object representation.

5.3.1 Preliminary: 2D part-based object models

2D part-based models are a category of object models where an object (category) is represented by a set of 2D parts and each part is allowed to deform around its anchor position in the 2D image-domain, which can date back to the original idea of Fischler and Elschlager [Fischler and Elschlager, 1973]. For simplicity, we introduce our model at a fixed scale; at test time we handle object of different sizes by searching over an

image pyramid. Let \mathbf{I} denote an image, \mathcal{V} be the part set in a part-based model, and p_i denote a candidate location² of part i in the image-domain. For a part hypothesis $\mathbf{p} = \{p_i\}_{i \in \mathcal{V}}$ in an image \mathbf{I} , the score function of 2D part-based models can be expressed as:

$$S(\mathbf{I}, \mathbf{p}) = \sum_{i \in \mathcal{V}} S_i(\mathbf{I}, p_i) + \sum_{ij \in \mathcal{E}} S_{ij}(p_i, p_j), \quad (5.1)$$

where $\mathcal{G}=(\mathcal{V}, \mathcal{E})$ is the tree-structure relational graph whose node set is the part set \mathcal{V} of the model, and the edge set \mathcal{E} specifies the pairs of parts between which certain geometrical constraints are imposed. $S_i(\mathbf{I}, p_i)$ is the unary term corresponding to the local appearance score for placing the i -th part template at location p_i . And $S_{ij}(p_i, p_j)$ is the pairwise term that penalizes the displacement of the i -th and j -th parts according to some prior model (*e.g.*, the deviation from their anchor position μ_{ij}).

In order to better represent generic objects, various part-based models have been proposed by enhancing the unary term $S_i(\mathbf{I}, p_i)$. For instance, $S_i(\mathbf{I}, p_i)$ is defined as $\alpha_i \phi(\mathbf{I}, p_i) + b_i$ in [Felzenszwalb et al., 2010] for object detection, where α_i is the template parameter of part i , $\phi(\mathbf{I}, p_i)$ is the image feature extracted at location p_i , and b_i is a bias term. In pose estimation, the idea of mixture of parts was adopted in [Yang and Ramanan, 2012], by defining $S_i(\mathbf{I}, p_i, t_i) = \alpha_i^{t_i} \phi(\mathbf{I}, p_i) + b_i^{t_i}$ where $\alpha_i^{t_i}$ is the template parameter of the i -th part of type t_i , and $b_i^{t_i}$ is the bias term that favors the part type assignment in the relational graph \mathcal{G} .

Regarding the pairwise term $S_{ij}(p_i, p_j)$, the following form has been commonly used in previous works³: $S_{ij}(p_i, p_j) = \beta_{ij} \psi(p_i, p_j)$, where $\psi(p_i, p_j)$ is a four-dimensional vector defining the pairwise displacement between part i and part j relative to their anchor position μ_{ij} , *i.e.*, $\psi(p_i, p_j) = (dx_{ij}, dy_{ij}, dx_{ij}^2, dy_{ij}^2)^T$ where $(dx_{ij}, dy_{ij}) = p_i - p_j - \mu_{ij}$, dx_{ij}^2 is a simplified form of $(dx_{ij})^2$, and $\beta_{ij} = (\beta_{ij}^a, \beta_{ij}^b, \beta_{ij}^c, \beta_{ij}^d)$ is the model

²For clarity, here we focus on the case where the parts are parametrized by their 2D locations. However, more complex parametrizations of the geometric configuration of the parts can be considered.

³This term can be further extended to enrich the model. For example, a part-type-specific term was adopted in [Yang and Ramanan, 2012] to handle part types.

As we see, 2D part-based models represent objects in the image-domain, by allowing the parts deform around the image-domain anchor positions μ .

5.3.2 Modeling active parts in the 3D scene-domain

It is difficult to model an object’s part configuration in the 2D image-domain, because, for a non-rigid object, the part locations in the image-domain after the 3D-2D projection from different viewpoints can have very different configurations, thus setting anchor positions in a model cannot well-capture the geometric properties of objects. To remove such geometric variations, we introduce the way we model the parts of an object in the 3D scene-domain. To this end, we make the following two assumptions which were often made in the literature: (1) the depth variation of objects are small compared to the distance from the camera, which enables the adoption of the weak-perspective projection model; (2) the 3D configuration of an object’s parts can be written as linear combinations of a few basis shapes.

Under the weak perspective projection model, for an object with $|\mathcal{E}|$ pairs of parts, its inter-part distances in the image-domain, $\mathbf{w}_{(2 \times |\mathcal{E}|)}$, is the projection from the part landmark shape in the 3D scene-domain, $\mathbf{S}_{(3 \times |\mathcal{E}|)}$, to the image-domain, *i.e.*, $\mathbf{w} = \mathbf{R}\mathbf{S} + \mathbf{t}$, where $\mathbf{R}_{(2 \times 3)}$ is the rotation matrix and $\mathbf{t}_{(2 \times |\mathcal{E}|)}$ is the translation matrix [Ma et al., 2004].

Inspired by the non-rigid structure-from-motion techniques [Akhter et al., 2008; Dai et al., 2012], we propose to model an object’s part configuration directly in the 3D scene-domain by characterizing the 3D inter-part shape \mathbf{S} as a subspace, which is represented as weighted combinations of K bases $\{\mathbf{B}_k, k = 1, \dots, K\}$, *i.e.*, $\mathbf{S} = \sum_{k=1}^K c_k \mathbf{B}_k$ where each base \mathbf{B}_k is a $3 \times |\mathcal{E}|$ matrix (note that there are constraints on $\{\mathbf{B}_k\}$ that are imposed to upgrade $\{\mathbf{B}_k\}$ from affine space to Euclidean space. See [Akhter et al., 2008; Dai et al., 2012] for details). Thus, the inter-part configuration in the image-

domain \mathbf{w} can be formulated as follows:

$$\mathbf{w} = \mathbf{R} \sum_{k=1}^K c_k \mathbf{B}_k + \mathbf{t} = \mathbf{R} \mathbf{c}^T \begin{pmatrix} \mathbf{B}_1 \\ \dots \\ \mathbf{B}_K \end{pmatrix} + \mathbf{t}, \quad (5.3)$$

where $\mathbf{c} = (c_1, c_2, \dots, c_K)^T$ are the weight vector.

Let us denote the reconstruction matrix as $\mathbf{m}_{(2 \times 3K)} = \mathbf{R} \mathbf{c}^T$ and the 3D geometric subspace as $\mathbf{B}_{(3K \times |\mathcal{E}|)} = (\mathbf{B}_1; \mathbf{B}_2; \dots; \mathbf{B}_K)$. By translating to the object hypothesis center such that the centroid \mathbf{t} is cancelled out, we have $\mathbf{w} = \mathbf{m} \mathbf{B}$. In this way, we can configure an object's parts in the image-domain from the subspace spanned by \mathbf{B} , and allow all parts to deform in the scene-domain rather than fixing them in anchor positions.

More specifically, in previous part-based models, as shown in Equation 5.2, the inter-part distance vector $\Delta \mathbf{p}$ is constrained to move around a fixed anchor positions $\boldsymbol{\mu}$, with a penalization on the displacement vector $(\Delta \mathbf{p} - \boldsymbol{\mu})$ in Gaussian fashion. However, in our scene-domain active parts model, we do not associate any of the pairwise parts with an fixed anchor position. Instead, we define our part's anchor configuration in the image-domain as a projection from the 3D scene-domain configuration, which is constructed from the subspace \mathbf{B} . Thus, our displacement vector is defined as $\Delta \mathbf{p} - f(\mathbf{w})$, *i.e.*, the difference between the inter-parts distance $\Delta \mathbf{p}$ of the hypothesis and the projected part configuration from the scene-domain.

Here, $f(\mathbf{w})$ is a transformation function from the $2 \times |\mathcal{E}|$ image-domain configuration matrix $\mathbf{w} = (x_1, x_2, \dots, x_{|\mathcal{E}|}; y_1, y_2, \dots, y_{|\mathcal{E}|})$ to a $2|\mathcal{E}| \times 1$ vector form $(x_1, x_2, \dots, x_{|\mathcal{E}|}, y_1, y_2, \dots, y_{|\mathcal{E}|})^T$, namely, $f(\mathbf{w}) = (\mathbf{e}_1 \mathbf{w} \mathbf{A} + \mathbf{e}_2 \mathbf{w} \widehat{\mathbf{A}})^T$ where $\mathbf{e}_1, \mathbf{e}_2, \mathbf{A}, \widehat{\mathbf{A}}$ are constants. $\mathbf{e}_1 = (1, 0)$, $\mathbf{e}_2 = (0, 1)$. \mathbf{A} and $\widehat{\mathbf{A}}$ are $|\mathcal{E}| \times 2|\mathcal{E}|$ matrices. $\mathbf{A} = (\mathbf{I}_{|\mathcal{E}|}, \mathbf{0})$, and $\widehat{\mathbf{A}} = (\mathbf{0}, \mathbf{I}_{|\mathcal{E}|})$ where $\mathbf{I}_{|\mathcal{E}|}$ is the $|\mathcal{E}| \times |\mathcal{E}|$ identity matrix, and $\mathbf{0}$ is the $|\mathcal{E}| \times |\mathcal{E}|$ zero matrix.

Therefore, with $\mathbf{w} = \mathbf{mB}$, the score function of our model is:

$$S(\mathbf{I}, \mathbf{p}, \mathbf{m}) = \sum_{i \in \mathcal{V}} S_i(\mathbf{I}, p_i) + (\Delta \mathbf{p} - f(\mathbf{mB}))^T \boldsymbol{\tau} + (\Delta \mathbf{p} - f(\mathbf{mB}))^T \boldsymbol{\Lambda} (\Delta \mathbf{p} - f(\mathbf{mB})), \quad (5.4)$$

where $f(\mathbf{mB}) = \left(\mathbf{e}_1 \mathbf{mB} \mathbf{A} + \mathbf{e}_2 \mathbf{mB} \widehat{\mathbf{A}} \right)^T$.

5.3.3 Modeling appearance with occlusions

As discussed in Section 5.3.1, appearance information is usually encoded within the unary term⁴ of part-based models (*e.g.*, [Felzenszwalb et al., 2010] models appearance by $\alpha_i \phi(\mathbf{I}, p_i)$). Occlusions frequently occur when a non-rigid object is projected from the 3D scene-domain to the 2D image-domain, either because of self-occlusions or occluded by other objects. In order to handle occlusions, we define a binary occlusion state o_i for each part, and a visible-state template α_i^v when $o_i = 0$ as well as an occluded-state template α_i^o when $o_i = 1$. Accordingly, our unary term in Equation 5.4 is defined as:

$$S_i(\mathbf{I}, p_i) = \max_{o_i} ((1 - o_i) (\alpha_i^v \phi(\mathbf{I}, p_i)), o_i \alpha_i^o \phi(\mathbf{I}, p_i)) + b_i. \quad (5.5)$$

To summarize, we can formally define our model as $(\mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\Lambda}, \{\alpha_i^v\}, \{\alpha_i^o\}, \{b_i\})$, where \mathbf{B} denotes the scene-domain geometric subspace for 3D part configuration, $\boldsymbol{\tau}$ and $\boldsymbol{\Lambda}$ are the deformation parameter matrices, $\{\alpha_i^v\}, \{\alpha_i^o\}$ are the unary term parameters, and $\{b_i\}$ are the bias terms.

⁴Here we present our method based on [Felzenszwalb et al., 2010] in object detection. However, It is similar to apply our occlusion modeling method to models in pose estimation, *e.g.*[Yang and Ramanan, 2012].

5.4 Inference

Our method can obtain rich object representation, including 2D object and parts localization \mathbf{p} , 3D landmark shape \mathbf{S} , and camera viewpoint \mathbf{R} . To infer them, we maximize $S(\mathbf{I}, \mathbf{p}, \mathbf{m})$ in Equation 5.4 over the part hypothesis \mathbf{p} and reconstruction matrix \mathbf{m} using a coordinate descent approach:

- 1). *Optimize over \mathbf{p}* : Fix \mathbf{m} , maximize $S(\mathbf{I}, \mathbf{p}, \mathbf{m})$ over \mathbf{p} , using Dynamic Programming;
- 2). *Optimize over \mathbf{m}* : Fix \mathbf{p} , compute the close-form solution \mathbf{m}^* that maximizes $S(\mathbf{I}, \mathbf{p}, \mathbf{m})$.

Both steps are executed alternatively until convergence.

By fixing \mathbf{m} , step 1 is essentially equivalent to the traditional part-based model inference procedure. We follow [Felzenszwalb et al., 2010] and use dynamic programming to obtain the optimal \mathbf{p}^* . We initialize \mathbf{m} to be the part anchor positions of [Azizpour and Laptev, 2012]. And in step 2, we have a closed-form solution \mathbf{m}^* that maximizes $S(\mathbf{I}, \mathbf{p}, \mathbf{m})$ (see Appendix A for the derivation procedure):

$$\begin{aligned} \mathbf{m}_1^* &= \mathbf{H}_1 (\mathbf{B}\mathbf{A}(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\mathbf{A}^T\mathbf{B}^T)^{-1}, \\ \mathbf{m}_2^* &= \mathbf{H}_2 (\mathbf{B}\hat{\mathbf{A}}(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\hat{\mathbf{A}}^T\mathbf{B}^T)^{-1}, \end{aligned} \quad (5.6)$$

where \mathbf{m}_1^* , \mathbf{m}_2^* are the first and second rows of the optimal solution $\mathbf{m}_{(2 \times 3K)}^*$. \mathbf{B} is the scene-domain geometric subspace, \mathbf{H}_1 and \mathbf{H}_2 are the first and second rows of matrix $\mathbf{H}_{(2 \times 3K)}$ which is defined as follows:

$$\mathbf{H} = \mathbf{e}_1^T ((\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\boldsymbol{\Delta}\mathbf{p} + \boldsymbol{\tau})^T \mathbf{A}^T\mathbf{B}^T + \mathbf{e}_2^T ((\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\boldsymbol{\Delta}\mathbf{p} + \boldsymbol{\tau})^T \hat{\mathbf{A}}^T\mathbf{B}^T,$$

where $\boldsymbol{\tau}$ and $\boldsymbol{\Lambda}$ are the deformation parameter matrices; $\boldsymbol{\Delta}\mathbf{p}$ is the inter-parts distance vector of a hypothesis; \mathbf{e}_1 , \mathbf{e}_2 , \mathbf{A} , $\hat{\mathbf{A}}$ are the constant matrices defined in Section 5.3.2.

5.4.1 3D landmark shape and viewpoint recovery

After we obtain the detected \mathbf{p}^* described above, now we introduce how we recover the 3D landmark shape and viewpoint of the detected object. As discussed in Section 5.3.2, $\mathbf{w} = \mathbf{R}\mathbf{S}$ and the 3D landmark shape $\mathbf{S} = \mathbf{c}^T\mathbf{B}$. Given the detection \mathbf{p}^* in 2D image-domain and the known geometric subspace \mathbf{B} , we recover the 3D landmark shape and viewpoint as follows:

$$\mathbf{R}^*, \mathbf{c}^* = \min_{\mathbf{R}, \mathbf{c}} \|\Delta\mathbf{p}^* - f(\mathbf{R}\mathbf{c}^T\mathbf{B})\|^2 \quad (5.7)$$

where $\Delta\mathbf{p}^*_{(2|\mathcal{E}|\times 1)}$ is the inter-part distance vector obtained from \mathbf{p}^* , and $f(\cdot)$ is the transformation function used in Equation 5.4. Because the squared error is linear in \mathbf{R} and \mathbf{c} , we obtain the optimal $\mathbf{R}^*, \mathbf{c}^*$ of Equation 5.7 with iterative least-squares algorithm [Hejrati and Ramanan, 2012]. Then, we obtain the 3D landmark shape $\mathbf{S}^* = \mathbf{c}^{*T}\mathbf{B}$, and the viewpoint \mathbf{R}^* of the detected object.

5.5 Learning

Our training data consists of a set of positive training examples $\mathbf{x}_n = \{\mathbf{I}_n, \mathbf{p}_n, \mathbf{o}_n\}$, negative training examples $\mathbf{x}_n = \{\mathbf{I}_n\}$, and corresponding example label y_n , $n \in \{1, \dots, N\}$, where N is the total number. \mathbf{I}_n is the image with object bounding boxes, \mathbf{p}_n is the part bounding boxes in \mathbf{I}_n , and \mathbf{o}_n is the parts' occlusion states in \mathbf{I}_n . We learn the model parameter in two steps: firstly we learn the 2D image-domain parameters $\Theta = (\boldsymbol{\tau}, \boldsymbol{\Lambda}, \{\boldsymbol{\alpha}_i^v\}, \{\boldsymbol{\alpha}_i^o\}, \{b_i\})$, then we learn the 3D scene-domain geometric subspace \mathbf{B} .

5.5.1 Learning the 2D image-domain parameter Θ

We learn the image-domain model parameters in a discriminative way by minimizing the loss function $L(\Theta) = \frac{1}{2}\|\Theta\|^2 + C \sum_{n=1}^N \max(0, 1 - y_n S_{\Theta}(\mathbf{x}_n))$.

We follow the strongly supervised learning paradigm in [Azizpour and Laptev, 2012] to learn Θ . In order to reduce the influence of the imprecise part annotation in the training data and the possibly low discriminative power of some annotated parts, we allow our part models to approximately overlap with the training part bounding boxes in positive images. This is achieved by constraining the searching space \mathbf{p} of the score function to be $\mathbf{Z}_{\mathbf{p}}(\mathbf{x}_n)$ that is consistent with the annotation \mathbf{p}_n :

$$S_{\Theta}(\mathbf{x}_n) = \max_{\mathbf{p} \in \mathbf{Z}_{\mathbf{p}}(\mathbf{x}_n)} S(\mathbf{I}, \mathbf{p}), \quad (5.8)$$

where $\mathbf{Z}_{\mathbf{p}}(\mathbf{x}_n) = \begin{cases} \{\mathbf{p} \in \mathbb{P} | O(\mathbf{p}, \mathbf{p}_n) > t_{ovp}\} & \text{if } \mathbf{p}_n \text{ available,} \\ \mathbb{P} & \text{otherwise.} \end{cases}$ \mathbb{P} is the set of all possible part bounding boxes, and $O(\cdot, \cdot)$ is the intersection over union (IoU) measure of two bounding boxes, we set $t_{ovp} = 0.5$ in our experiments.

5.5.2 Learning the 3D geometric subspace \mathbf{B}

Given training data with labeled 2D part locations $\{\mathbf{p}_n\}$, we can learn the scene-domain geometric subspace \mathbf{B} by casting this as non-rigid structure-from-motion (NRSFM) problem. As shown in Equation 5.3, the inter-part distances in the image-domain, $\mathbf{w}_{(2 \times |\mathcal{E}|)}$, is related to the 3D inter-part distances in the scene-domain, $\mathbf{S}_{(3 \times |\mathcal{E}|)}$, via a weak perspective projection model. Given N positive training examples of a certain object category which shares the same 3D part configuration subspace \mathbf{B} , we have $\mathbf{W} = \mathbf{M}\mathbf{B} + \mathbf{T}$, where $\mathbf{W}_{(2N \times |\mathcal{E}|)} = (\mathbf{w}_1; \mathbf{w}_2; \dots; \mathbf{w}_N)$ is the 2D inter-part distance matrix in N images, $\mathbf{B}_{(3K \times |\mathcal{E}|)}$ is the scene-domain geometric subspace shared among the same object category, $\mathbf{M}_{(2N \times 3K)} = (\mathbf{m}_1; \dots; \mathbf{m}_N)$ is the reconstruction coefficient

matrix, and $\mathbf{T}_{(2N \times |\mathcal{E}|)}$ is the translation matrix. Given the 2D part locations \mathbf{W} obtained from $\{\mathbf{p}_n\}$, we use the publically-available NRSFM code [Akhter et al., 2008; Dai et al., 2012] to learn \mathbf{B} .

5.6 Experiments

Since the proposed method provides fine-grained object representation both in 2D and 3D, we conduct two sets of experiments to evaluate it. The first set of experiments evaluates our method on 2D object and parts detection, and compares with 2D part-based models [Felzenszwalb et al., 2010; Azizpour and Laptev, 2012; Bourdev et al., 2010; Parkhi et al., 2011; Wang et al., 2013; Girshick et al., 2014b]. The second set tests on 3D landmark shape and viewpoint estimation, and compares our method with 3D part-based models also learned from 2D data alone [Hejrati and Ramanan, 2012; Arie-Nachimson and Basri, 2009].

5.6.1 Datasets

The experiments are based on three challenging datasets: the PASCAL VOC 2010 dataset [Everingham et al., 2010] for 2D object and parts detection, the Human3.6M dataset [Ionescu et al., 2014] for 2D and 3D landmark shape estimation, and the PASCAL VOC 2007 car dataset [Arie-Nachimson and Basri, 2009] for viewpoint classification. For the PASCAL VOC 2010 dataset, following [Azizpour and Laptev, 2012; Chen et al., 2014], we evaluate on the six animal classes. These animal classes serve as a common testbed for object model evaluation (*e.g.*, in [Azizpour and Laptev, 2012; Chen et al., 2014]), because of the high difficulty in addressing them caused by highly non-rigid deformations, intra-class variations, and different degrees of occlusions. In addition to the part annotation provided in [Azizpour and Laptev, 2012] which is used in our method, we further annotate the locations of occluded parts. We use `trainval`

subset of PASCAL VOC 2010 for training and the `test` subset for testing. Meanwhile, the Human3.6M dataset provides both 2D and 3D landmark annotations, and serves as a suitable testbed to evaluate 2D and 3D pose landmark shape estimation of our method. We use the subject S1 of walking action for training and S7 for testing. Lastly, we test viewpoint estimation on the PASCAL VOC 2007 car dataset [Arie-Nachimson and Basri, 2009], which consists of 200 cars images marked with 40 discrete viewpoint class labels.

5.6.2 Implementation details

Our model is modular w.r.t. the appearance feature $\phi(\mathbf{I}, p_i)$ used in the unary term $S_i(\mathbf{I}, p_i)$. Thus in the experiments on 2D object and parts detection, we construct our method based on the DPM structure as in [Felzenszwalb et al., 2010], but with various types of features: HOG feature as DPM [Felzenszwalb et al., 2010], Segmentation feature as SegDPM [Fidler et al., 2013], and CNN feature as DeepPyramid DPM [Girshick et al., 2014b]. And we apply bounding box regression for object detection. While in the experiments on 2D, 3D pose and viewpoint estimation, we construct our model based on the Mixture-of-Parts structure as [Yang and Ramanan, 2012] of 10 part types, based on HOG feature and CNN feature as [Chen and Yuille, 2014]. We use the Caffe [Jia et al., 2014] to compute the CNN feature. When learning the scene-domain geometric subspace \mathbf{B} , we follow the NRSFM techniques [Dai et al., 2012] to set the geometric subspace bases number $K = 5$ for object and parts detection and $K = 8$ for pose and viewpoint estimation. The part number in our models is set to be consistent with the part annotation $\{\mathbf{p}_n\}$ of the training data.

Table 5.1: Average precision for animal detection on PASCAL VOC 2010. Our method outperforms all baselines of part-based models.

Approach	Bird	Cat	Cow	Dog	Horse	Sheep	mAP
Ours w. HOG	15.3	28.6	28.7	28.2	48.3	30.1	29.9
SSDPM [Azizpour and Laptev, 2012]	11.3	27.2	25.8	23.7	46.1	28.0	27.0
Poselets [Bourdev et al., 2010]	8.5	22.2	20.6	18.5	48.2	28.0	24.3
DPM [Felzenszwalb et al., 2010]	11.0	23.6	23.2	20.5	42.5	29.0	25.0
Ours w. Seg & HOG	26.1	51.2	35.3	41.7	52.8	37.5	40.8
Regionlets [Wang et al., 2013]	25.9	51.2	28.9	35.8	40.2	43.9	37.65
DefPM [Parkhi et al., 2011]	-	45.3	-	36.8	-	-	-
SegDPM [Fidler et al., 2013]	25.3	48.8	30.4	37.7	46.0	35.7	37.3
Ours w. CNN	38.9	48.5	38.8	47.5	55.0	48.3	46.2
DP-DPM [Girshick et al., 2014b]	36.5	48.0	35.0	45.7	50.2	49.1	44.1

5.6.3 Experiments on object and parts detection

5.6.3.1 Object detection

In order to achieve a fair comparison, we compare with three groups of part-based model baselines. The first group uses only HOG as local feature in the unary term $S_i(\mathbf{I}, p_i)$, including the DPM [Felzenszwalb et al., 2010], SSDPM [Azizpour and Laptev, 2012], and Poselets model [Bourdev et al., 2010]. The second group of baselines uses both segmentation and HOG features, including the SegDPM [Fidler et al., 2013], DefPM [Parkhi et al., 2011], and Regionlets [Wang et al., 2013]. And the last group of baseline models uses CNN feature, including DeepPyramid DPM (DP-DPM) [Girshick et al., 2014b], C-DPM [Savalle et al., 2013] and Conv-DPM [Wan et al., 2014]. As claimed before, we show the results of our method using three different features (w.

HOG, w. Seg & HOG, and w. CNN) as the baselines. The detailed quantitative results⁵ are shown in Table 5.1. Overall, our method improves the mean average precision (mAP) of the DPM baseline by 4.9%, the SegDPM baseline by 3.5%, and the DP-DPM baseline by 2.1%. Our method outperforms all baselines in detecting coarse-grained object representation.

It is important to note that although several deep learning approaches, *e.g.* [Girshick et al., 2014a; Zhu et al., 2015], performs better in the task of object detection, our method models fine-grained spatial relationship between parts, thus providing much richer object representation, such as 2D parts localization, 3D landmark shape and camera viewpoint, which is essential for further fine-grained applications.

Figure 5.2 shows representative qualitative results on object detection. Figure 5.2(a) shows example detection results for each of the six animal classes. As we see, our method provides a richer description for objects, *e.g.* the object parts are effectively localized. Figure 5.2(b) shows several cat detections, which demonstrates the robustness of our model under geometric variations. As it shows, we can robustly locate the cat instances with non-rigid deformations, viewpoint changes, and partial occlusions. Figure 5.2(c) shows some typical examples that are correctly localized by our model but missed by DPM.

5.6.3.2 Parts localization:

Our method can localize object parts and provides a richer description of objects. We adopt the widely used measure of PCP (Percentage of Correctly estimated body Parts) [V.Ferrari et al., 2008] to evaluate parts localization by our method. We consider the detection with the highest score that has more than 50% overlap with its bounding box,

⁵Since C-DPM [Savalle et al., 2013] and Conv-DPM [Wan et al., 2014] have not reported results in PASCAL VOC 2010 dataset, we do not list results of [Wan et al., 2014; Savalle et al., 2013]. Note that they show overall comparable results as DP-DPM.



Figure 5.2: *Our method provides a richer object representation and improves the detection results. The blue bounding boxes correspond to the whole object detection, and boxes of other colors correspond to semantic parts respectively, which may indicate different parts across classes. (a) shows one representative result for each of the six animal classes. (b) shows detection results of the cat class to illustrate the ability of our model to robustly represent objects under non-rigid deformations, viewpoint changes, and occlusions. (c) shows typical examples that are correctly localized by our Scene-Domain Active Part Model (SDAPM) but missed by DPM.*

Table 5.2: Part localization performance on PASCAL VOC 2010. The numbers are “PCP of our method” / “PCP of SSDPM [Azizpour and Laptev, 2012]”.

Object Class	Head	Fore legs	Hind legs	Torso/Back	Tail
Bird	50.7 / 28.1	-	15.2 / 12.5	-	35.0 / 20.7
Cat	71.1 / 62.8	18.9 / 11.4	-	50.3 / 37.2	18.1 / 10.1
Cow	72.8 / 56.2	85.7 / 60.9	80.3 / 58.1	77.2 / 69.3	-
Dog	59.0 / 48.7	33.6 / 37.5	-	34.5 / 21.6	30.0 / 9.7
Horse	65.9 / 67.1	82.2 / 53.1	80.6 / 55.7	88.5 / 67.4	68.2 / 42.9
Sheep	58.3 / 41.4	67.4 / 43.8	65.8 / 39.7	83.7 / 71.1	36.1 / 12.7

which factors out the effect of the detection. A part is considered as correctly localized if it has more than 40% overlap with the ground truth annotation. Table 5.2 shows the PCP result using our SDAPM model based on HOG feature and that of SSDPM [Azizpour and Laptev, 2012]. Our model outputs fairly precise locations for parts. As we see, our model offers better part localization than [Azizpour and Laptev, 2012], which validates the effectiveness of our method in locating parts.

5.6.3.3 Diagnostic experiments

Importance of scene-domain modeling: To better justify the contribution of our method by modeling active parts in the scene-domain, we compare our method with its two variants: i) “SDAPM without scene-domain modeling” that uses DPM’s standard pairwise term with fixed anchor positions as in Equation 5.2, instead of the one in Equation 5.4. ii) “SDAPM replaced by image-domain geometric modeling” that replaces the proposed scene-domain geometric modeling by 2D image-domain geometric modeling, where a 2D image-domain geometric subspace $\bar{\mathbf{B}}$ is learned via PCA on the

Table 5.3: Average detection precision of SDAPM and its three variants on the six animal classes of PASCAL VOC 2010 dataset.

Approach	Bird	Cat	Cow	Dog	Horse	Sheep	mAP
i) SDAPM w/o scene-domain modeling	11.2	27.0	26.9	23.1	46.7	28.6	27.3
ii) SDAPM w. image-domain geometric modeling	5.8	13.1	11.9	9.7	29.5	19.2	14.9
iii) SDAPM w/o occlusion modeling	11.9	25.7	24.2	22.1	44.8	29.1	26.3
Our full SDAPM model w. HOG	15.3	28.6	28.7	28.2	48.3	30.1	29.9

normalized 2D inter-part distances \mathbf{w} , and use it to construct $f(\mathbf{w})$ in Equation 5.4, *i.e.*, $f(\mathbf{w}) = \mathbf{c}^T \overline{\mathbf{B}}$ where \mathbf{c} is the weight vector. The comparison results with HOG feature are shown in Table 5.3. The 1st, 2nd, and 4th rows validate the contribution of modeling geometric statistics in the 3D scene-domain. In particular, the second row demonstrates the contribution of modeling parts’ geometric statistics in 3D scene-domain rather than modeling in 2D image-domain.

Importance of occlusion modeling: To validate the importance of our model that explicitly models occlusions, we create another variant to compare with: iii) “SDAPM without occlusion modeling” that discards the occlusion state $\{o_i\}$ and occluded-state templates $\{\alpha_i^o\}$ of our model, but uses DPM’s unary term instead. The last two rows of Table 5.3 justify the importance of explicitly modeling occlusions.

5.6.4 Experiments on pose and viewpoint estimation

In addition to 2D object and parts localization, our method provides 3D landmark shape and viewpoint estimation. In this experiments, we evaluate our method on 3D fine-grained representation.

Table 5.4: 2D pose estimation performance on Human3.6M dataset, comparing with MoP [Yang and Ramanan, 2012] and MH-Car [Hejrati and Ramanan, 2012]. The reported numbers are PCP (Probability of Correct Pose).

Approach	Upper arms	Lower arms	Upper legs	Lower legs	Overall
MoP	60.2	31.1	68.4	62.7	55.6
MH-Car	60.0	31.4	69.0	62.2	55.7
Ours w. HOG	61.0	33.8	69.7	63.9	57.1

5.6.4.1 2D pose and 3D landmark shape estimation

We construct our model following the mixture-of-parts structure [Yang and Ramanan, 2012], with HOG and CNN features respectively.

MH-Car [Hejrati and Ramanan, 2012] is state-of-the-art method that recovers 3D landmark shape and viewpoint by learning from 2D data alone. It detects the 2D pose, then reconstruct it into 3D via two separate steps. Here, we first compare our model with [Hejrati and Ramanan, 2012] on 2D pose estimation. As shown⁶ in Table 5.4, our model improves over [Hejrati and Ramanan, 2012] on average, and especially in the estimation of lower arms. This is mainly because the scene-domain modeling in our model helps excluding those incorrect configurations in the lower arms. Two examples are shown in Figure 5.3(a) to illustrate this, where the right arms are correctly estimated by SDAPM but mis-estimated by [Hejrati and Ramanan, 2012]. Our method outperforms [Yang and Ramanan, 2012] on average⁶, which demonstrates the contribution of modeling parts’ geometric statistics in 3D scene-domain instead of 2D image-domain.

Moreover, 3D landmark shapes are recovered in addition to the 2D poses. As shown in

⁶For fair comparison, we list the result of our model w. HOG feature, the same feature used in [Hejrati and Ramanan, 2012; Yang and Ramanan, 2012].

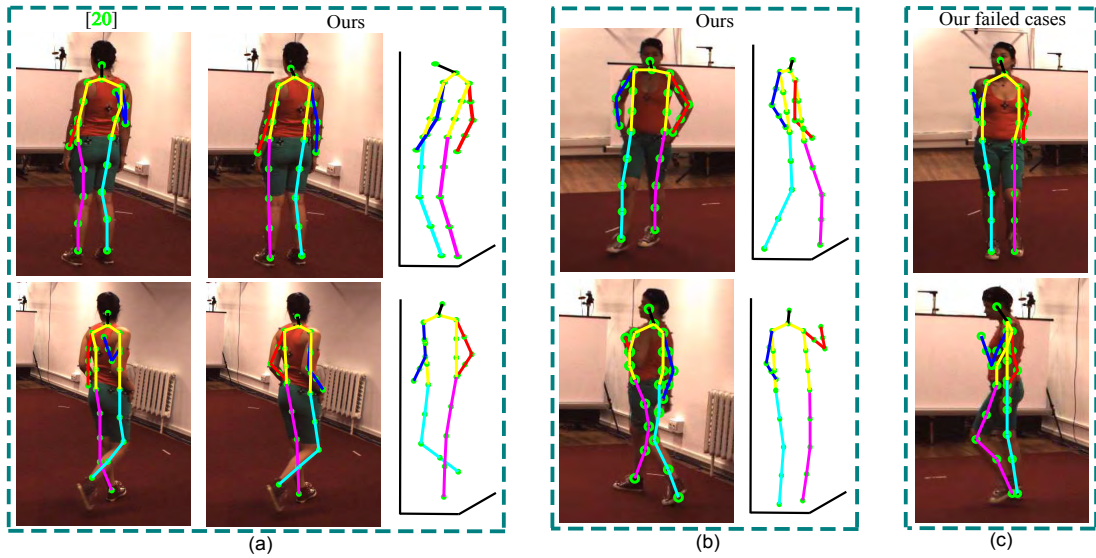


Figure 5.3: *Our method provides a richer object representation including 2D pose, 3D landmark shape, and viewpoint. (a) shows typical poses that are correctly estimated by our method w. HOG but mis-estimated by [Hejrati and Ramanan, 2012] (e.g. the right arm). (b) shows the 2D pose and 3D landmark shape estimations of our method for human in varying viewpoints. (c) gives some failed examples of our method.*

Table 5.5: 3D landmark shape estimation on Human3.6M dataset and camera viewpoint estimation on PASCAL Car 2007 dataset, comparing with MA-N [Arie-Nachimson and Basri, 2009] and MH-Car [Hejrati and Ramanan, 2012].

Metric	MA-N	MH-Car	Ours w. HOG	Ours w. CNN
Average RMS error (mm)	-	298.6	217.2	146.7
Average viewpoint error (°)	27	16	14	8

Figure 5.3(a) (b), the estimated 3D landmark shapes are shown beside the corresponding 2D poses. Although there is inaccuracy in the recovered 3D landmark shapes, e.g., the 3D head position in the upper image of the third column in Figure 5.3(a) is not correct, the results are fairly good.

In order to quantitatively evaluate on 3D landmark shape estimation, we compare with [Hejrati and Ramanan, 2012] using the root mean square error (RMS) metric, which measures the difference of the estimated 3D landmark shape comparing to the 3D landmark ground truth. As shown in the first row of Table 5.5, SDAPM outperforms [Hejrati and Ramanan, 2012], especially with CNN feature, which validates the effectiveness of modeling in the scene-domain via a unified process other than two separated steps.

5.6.4.2 Camera viewpoint estimation

Together with the 3D landmark shape, our SDAPM model estimates projection viewpoint as well. We evaluate viewpoint classification on our car SDAPM model learned from the PASCAL VOC 2007 car dataset [Arie-Nachimson and Basri, 2009]. Given a test instance, we run our car model to estimate the camera projection matrix \mathbf{R}^* as well as 3D landmark shape \mathbf{S}^* as discussed in Section 5.4.1. Then we produce a quantized viewpoint label by matching the reconstructed 2D landmarks generated using the

estimated \mathbf{R}^* and \mathbf{S}^* to the landmark locations of the reference images (provided in the dataset). As shown in the last row of Table 5.5, our method produces an average viewpoint classification error of 8° , which outperforms state-of-the-art viewpoint estimation method MH-Car [Hejrati and Ramanan, 2012] with a mean error of 16° and MA-N [Arie-Nachimson and Basri, 2009] with a mean error of 27° . This suggests that our model can accurately recover the projection viewpoints.

5.6.5 More qualitative results

In Figure 5.4 below, we show more qualitative results of our method in the task of object and parts detection, on a number of representative testing images in PASCAL VOC 2010 dataset.

In Figure 5.5, we show more qualitative results of our method on Human3.6M dataset, demonstrating the performance of our method in 2D pose and 3D landmark shape estimation.

5.7 Summary

In this chapter, we have proposed a novel part-based modeling method in the scenario where the training data are based on 2D images. Our method models object parts in the 3D scene-domain and explicitly models occlusions, and accordingly provides finer-grained object representation, including 2D object, parts localization, 3D landmark shape and camera viewpoint estimation. Our method differs from previous part-based object models in that we explore and model the 3D geometric statistics of object parts. Experimental results on two challenging tasks, *i.e.*, object and parts detection, 3D pose and viewpoint estimation, have demonstrated that the proposed method shows superior performance over existing methods with both better robustness to geometric

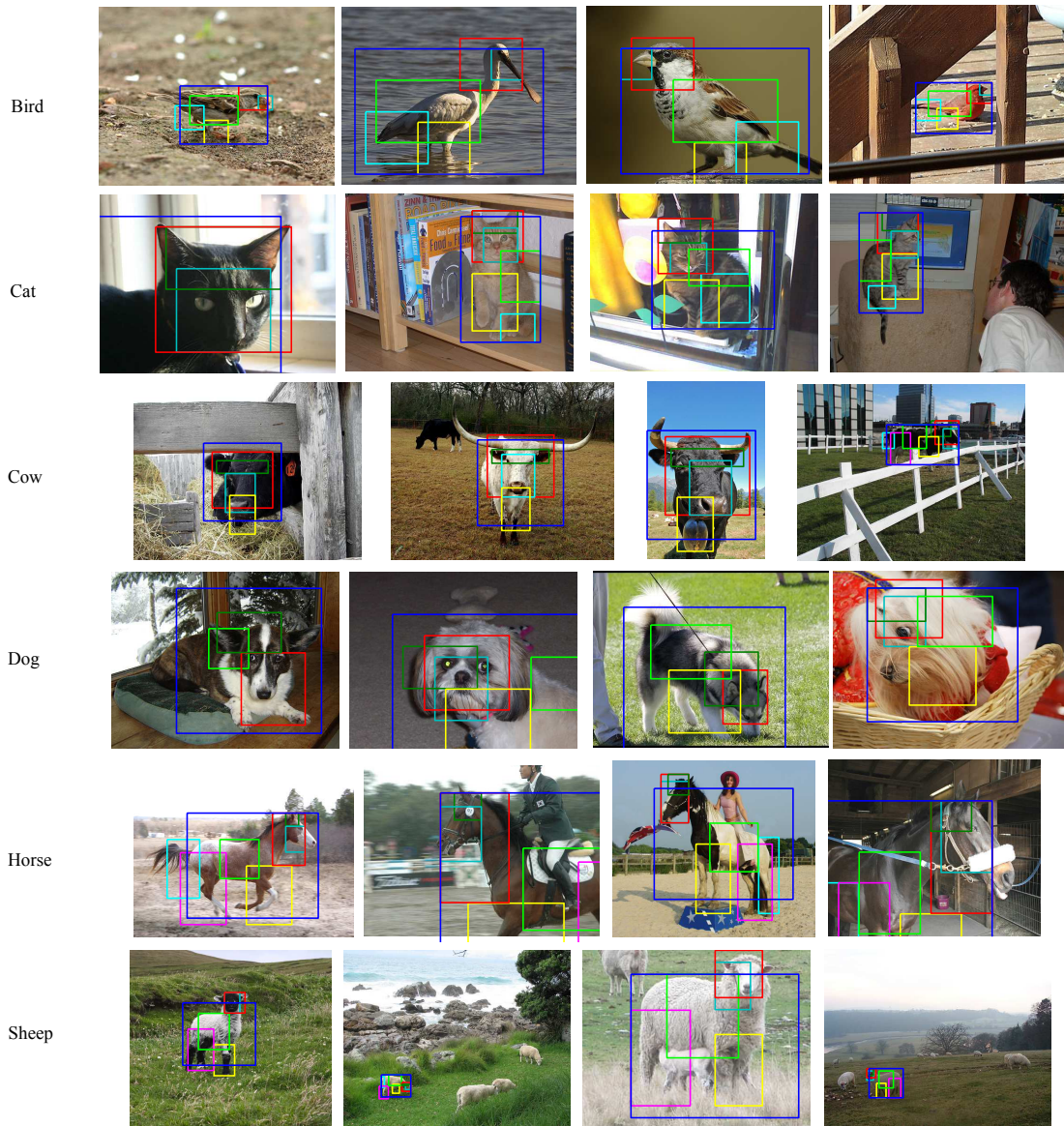


Figure 5.4: *More detection results of our model with HOG feature on the six animal classes. Each row shows one of the six classes respectively. The blue bounding box indicates the whole object detection result, and the boxes of other colors indicate the parts. As we see, our method can robustly represent the objects under non-rigid deformations, viewpoint changes, and occlusions, with parts localization.*

variations and richer object descriptions.

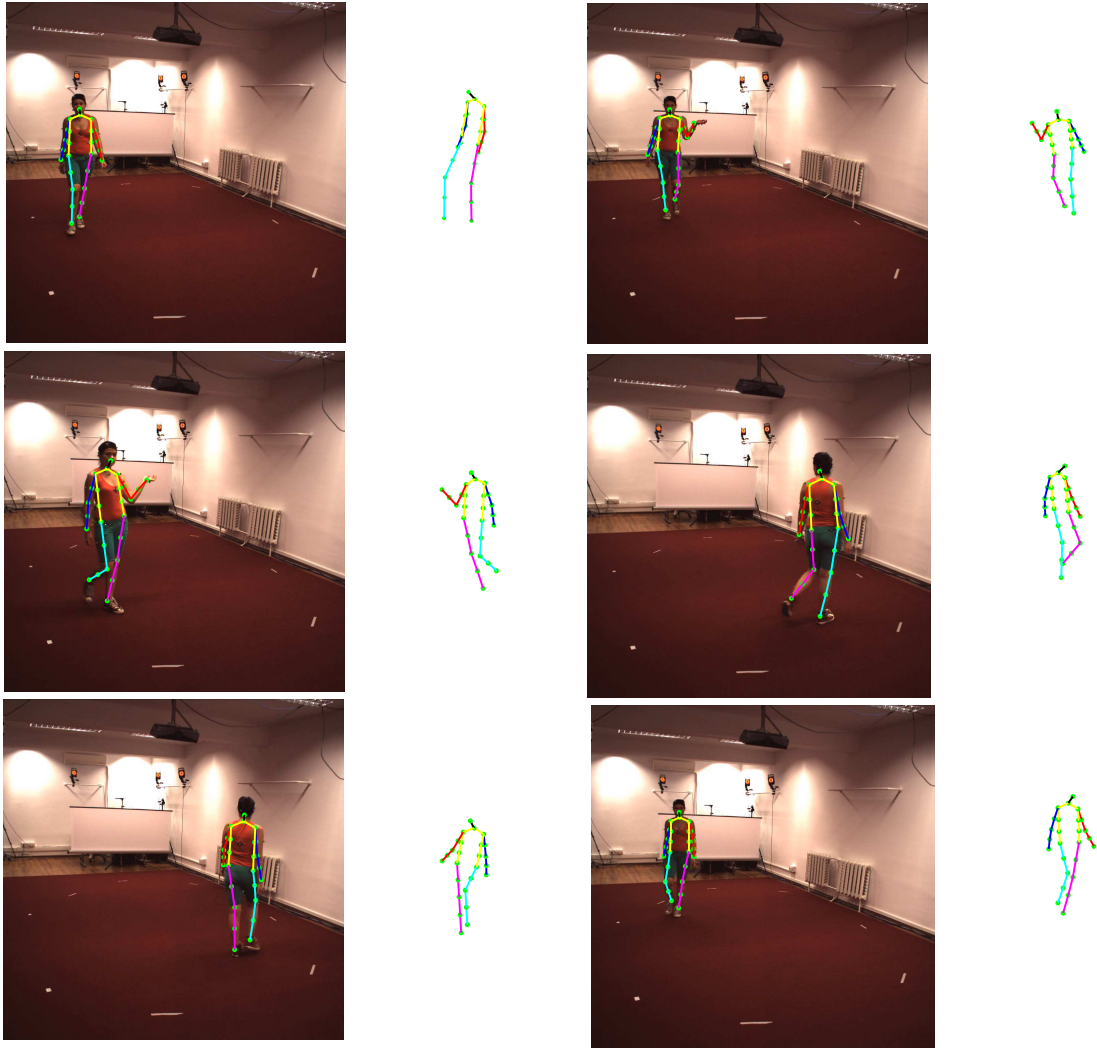


Figure 5.5: *Our method offers a richer object representation in 3D, in addition to the 2D pose. For each test image, we show, in a sub-figure on the right side of the image, the estimated 3D landmark shape using a different viewpoint from that of the test image, so as to illustrate the 3D configuration of the estimated landmark shapes.*

CHAPTER 6

Advanced: Joint Image-Sentence Representation & Joint Video-Sentence Representation

In the previous chapters, we introduced three different levels of modeling algorithms for joint image-text representation, *i.e.*, text concept modeling, image-level modeling, and object-level modeling. They have shown impressive performance in various multi-modal tasks, which outperforms the existing approaches. However, the three proposed models are for images and short texts only, such as words and phrases. In the real world, other visual and textual modalities, such as videos, long texts like sentences and documents, are also important and require to be processed.

Therefore, in this chapter, we explore joint representation of advanced visual and textual modalities, including joint image-sentence representation and joint video-sentence representation. And we validate its effectiveness in three multi-modal tasks, *i.e.*, image captioning, video annotation, and text-based video retrieval.

6.1 Joint Image-Sentence Representation

6.1.1 Image-sentence embedding

A lot of efforts have been devoted to image and sentence representation respectively. Recently, the learnt representation of images with deep convolutional neural networks

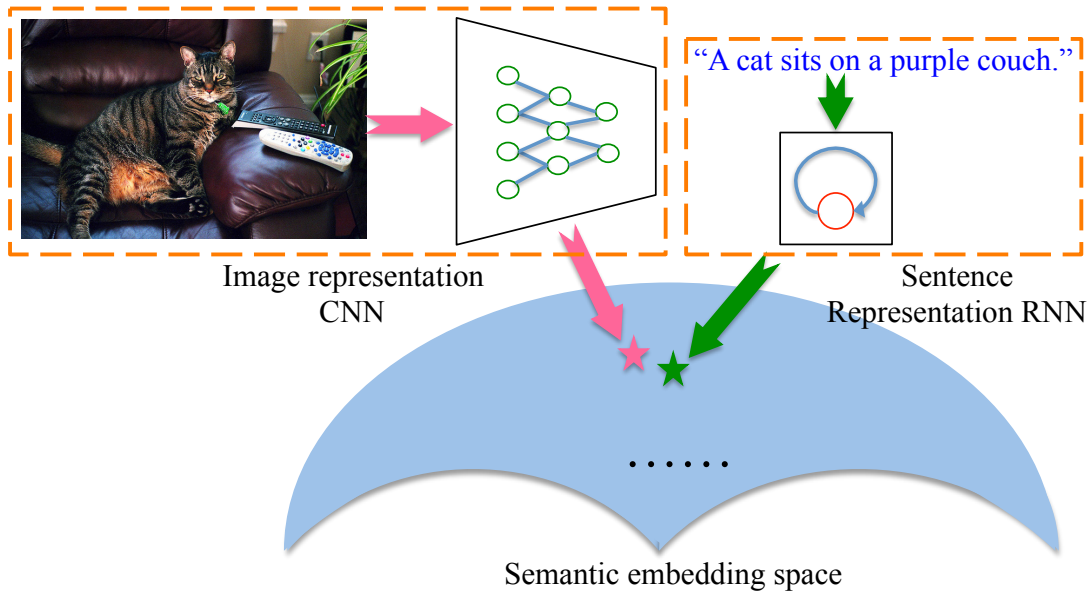


Figure 6.1: *The general framework of joint image-sentence embedding.*

(CNN) structures has shown superior expressivity power than traditional hand-crafted visual features, such as AlexNet [Krizhevsky et al., 2012], VGGNet [Simonyan and Zisserman, 2015], GoogleNet [Szegedy et al., 2015], C3D [Tran et al., 2015], etc.

On the other hand, various sentence representation models have been proposed, from the traditional modeling techniques based on Hidden Markov Models (HMM) [Baum and Petrie, 1966], to the recently proposed Recurrent Neural Network (RNN) models, such as Long-Short Term Memory (LSTM) model [Hochreiter and Schmidhuber, 1997] and Gated Recurrent Unit (GRU) model [Chung et al., 2014].

Therefore, the general joint image-sentence representation models first capture the image and sentence feature respectively, using the state-of-the-art approaches above such as CNN models and RNN models, and then maps both of them to the latent semantic embedding space, as shown in Figure 6.1. The semantic embedding space is constructed such that the projections of a certain image and its corresponding sentences are close to each other.

Let D be the dimensionality of an image feature vector (e.g., AlexNet [Krizhevsky

et al., 2012] has 4096 as D), E the dimensionality of the embedding space. Let $\mathbf{v} \in \mathbb{R}^D$ denote the feature vector of image \mathbf{I} , and its corresponding sentence description is S . Let w_1, \dots, w_T be the words in a sentence S , where T is the number of words in S .

In each time step t , given the embedded feature vector \mathbf{x}_t for a word w_t , the GRU encoder [Chung et al., 2014] updates its hidden state at time t , denoted by \mathbf{h}_t , using the following equations:

$$\mathbf{r}_t = \sigma(\mathbf{W}_r \mathbf{x}_t + \mathbf{U}_r \mathbf{h}_{t-1}), \quad (6.1)$$

$$\mathbf{z}_t = \sigma(\mathbf{W}_z \mathbf{x}_t + \mathbf{U}_z \mathbf{h}_{t-1}), \quad (6.2)$$

$$\bar{\mathbf{h}}_t = \tanh(\mathbf{W}_h \mathbf{x}_t + \mathbf{U}_h (\mathbf{r}_t \odot \mathbf{h}_{t-1})), \quad (6.3)$$

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \bar{\mathbf{h}}_t, \quad (6.4)$$

where \mathbf{r}_t and \mathbf{z}_t respectively denote the reset and update gates at time t , and $\bar{\mathbf{h}}_t$ is candidate activation at time t . In addition, \odot indicates element-wise multiplication operator and $\sigma(\cdot)$ is a sigmoid function. By applying this encoder to the sentence S through a series of GRU cells, we obtain the final embedding vector $\mathbf{h}_T \in \mathbb{R}^E$ of the sentence, which constructs the semantic space.

Therefore, the embedding function $f(\cdot)$ that maps images to the semantic space can be learnt by bi-directional ranking loss function as below:

$$\begin{aligned} L_{ISE} = & \sum_{\mathbf{v}} \sum_k \max(0, m + \|\mathbf{f}(\mathbf{v}) - \mathbf{t}\|_{L_2} - \|\mathbf{f}(\mathbf{v}) - \mathbf{t}_k\|_{L_2}) \\ & + \sum_{\mathbf{t}} \sum_j \max(0, m + \|\mathbf{t} - \mathbf{f}(\mathbf{v})\|_{L_2} - \|\mathbf{t} - \mathbf{f}(\mathbf{v}_j)\|_{L_2}), \end{aligned} \quad (6.5)$$

where m is the margin, \mathbf{v} denotes all the images, \mathbf{t}_k is a contrastive (non-descriptive, negative) sentence for image embedding \mathbf{v} , \mathbf{t} is the ground truth (positive) sentence for image embedding \mathbf{v} captured by GRU's last hidden state \mathbf{h}_T , and vice-versa with \mathbf{v}_j .

After we learn the semantic embedding space for joint image-sentence representation, given an input pair of image \mathbf{T} and sentence Q , we can represent \mathbf{T} as a CNN feature

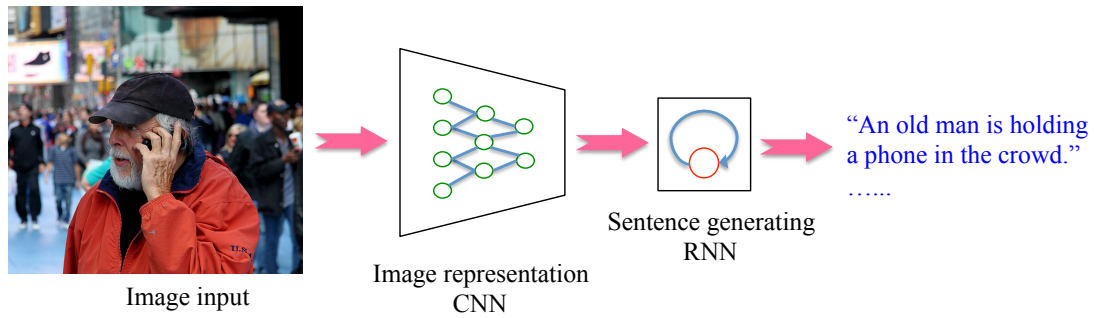


Figure 6.2: *The general framework of image captioning.*

vector \mathbf{u} and represent Q as a GRU feature \mathbf{h} . Thus, with the learnt embedding function $f(\cdot)$, the embedding distance between \mathbf{T} and Q is computed as $\|f(\mathbf{u}) - \mathbf{h}\|_{L_2}$. Now we show how to utilize such embedding function to help image-sentence tasks, such as image captioning.

6.1.2 Image captioning

Image captioning, whose goal is to generate a sentence caption for the query image, is an interesting problem involving both image and sentence modalities. Recently, various models have been proposed to handle this problem, *e.g.*, Google NIC [Vinyals et al., 2015], m-RNN [Mao et al., 2015], LRCN [Donahue et al., 2015], MSR/CMU [Chen and Zitnick, 2015], Spatial_ATT [Xu et al., 2015], Semantic_ATT [You et al., 2016], UnifyVSE [Kiros et al., 2015], MSR [Fang et al., 2015], denseCap [Johnson et al., 2016], *etc.*

Even a lot of models have been proposed, the state-of-the-art image captioning approaches are based on a common general framework, as shown in Figure 6.2. In particular, given an input image, we will use CNN models to capture its image feature; and then, we use such feature as the initial hidden state of a RNN model which is used to generate sentences. After keeping feeding the current word output as the input of the next time step, the RNN model will hit the final END token, which means a full

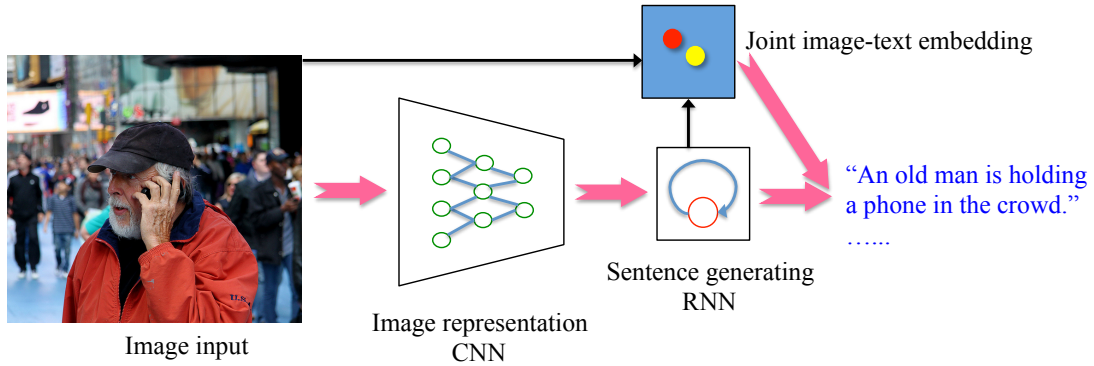


Figure 6.3: The framework of the proposed Embedding-Guided Image Captioning.

sentence has been generated, where the first one with the highest score will output to the user.

In order to learn an effective image captioning model, generally the cost function is defined to be the perplexity of a generated sentence $O = w_{1:T}$ given the image \mathbf{I} . Thus the score of each generated sentence is defined as:

$$S = \sum_{n=1}^T \log P(w_n | w_{1:n-1}, \mathbf{I}). \quad (6.6)$$

6.1.3 Embedding-Guided Image Captioning

Now we present a novel Embedding-Guided Image Captioning framework, which utilize the learned joint image-sentence embedding representation to help image captioning.

As shown in Figure 6.3, beyond the traditional sequential framework, we compute the embedding scores between each candidate caption in the beam search pool and the input image, and use that to improve image captioning. Mathematically, during the beam search of traditional image captioning, each candidate sentence has a original score computed as in Equation 6.6, which is defined as prior score S_{prior} . Besides, we compute the embedding score as defined in Section 6.1.1, as S_{embed} . And the final score

Table 6.1: The image captioning results in MSCOCO dataset [Lin et al., 2014] of the proposed Embedding-Guided Image Captioning approach, of different values of λ , with beam size set to be 50.

λ	Bleu_1	Bleu_2	Bleu_3	Bleu_4	Rouge_L	CIDEr	METEOR
0	0.693	0.511	0.370	0.266	0.506	0.861	0.238
0.1	0.700	0.523	0.385	0.285	0.513	0.898	0.241
0.2	0.703	0.529	0.392	0.294	0.517	0.911	0.243
0.3	0.702	0.528	0.391	0.293	0.517	0.906	0.242
0.4	0.697	0.523	0.387	0.291	0.514	0.895	0.240
0.5	0.692	0.517	0.383	0.287	0.511	0.884	0.238
0.6	0.689	0.513	0.379	0.285	0.509	0.876	0.237
0.7	0.686	0.511	0.377	0.283	0.507	0.869	0.235
0.8	0.684	0.509	0.376	0.283	0.506	0.864	0.235
0.9	0.682	0.507	0.374	0.281	0.504	0.856	0.234
1	0.680	0.505	0.372	0.279	0.503	0.850	0.233

is a combination of these two:

$$S_{final} = \lambda \times S_{prior} + (1 - \lambda) \times S_{embed}, \quad (6.7)$$

where λ is a weight parameter between 0 and 1.

6.1.4 Experiments on image captioning

We evaluate our Embedding-Guide Image Captioning framework on one of the largest image captioning dataset, MSCOCO dataset [Lin et al., 2014], which has over 400,000 training image-caption pairs, over 200,000 validation image-caption pairs, and over 40,000 testing images. We use all the training images for training, and randomly select

Table 6.2: The image captioning results in MSCOCO dataset [Lin et al., 2014] of the proposed Embedding-Guided Image Captioning approach, of different beam sizes. The last two rows are the results of the Google NIC baseline [Vinyals et al., 2015].

#beam	Bleu_1	Bleu_2	Bleu_3	Bleu_4	Rouge_L	CIDEr	METEOR
100	0.702	0.527	0.391	0.293	0.517	0.904	0.242
50	0.703	0.529	0.392	0.294	0.517	0.911	0.243
25	0.704	0.530	0.393	0.295	0.518	0.910	0.243
10	0.706	0.532	0.395	0.296	0.520	0.911	0.243
2	0.699	0.524	0.385	0.283	0.513	0.875	0.237
1	0.688	0.511	0.366	0.261	0.506	0.839	0.232

5000 images for validation and 5000 for testing, following Karpathy *et al.* [Karpathy and Fei-Fei, 2015]. We choose the Google NIC model [Vinyals et al., 2015] as our baseline model and implement based upon it. We employ the common image captioning metrics, Bleu_1, Bleu_2, Bleu_3, Bleu_4 [Papineni et al., 2002], Rouge_L [Lin and Och, 2004], CIDEr [Vedantam et al., 2015], METEOR [Denkowski and Lavie, 2014], to evaluate the performance comprehensively. For all metrics, the higher value means the better performance.

We first evaluate the effect of using different values of parameter λ in Equation 6.7. The results is shown in Table 6.1 using beam size of 50. As we see, when $\lambda = 0$, it is equivalent to that only uses embedding score to rank, and $\lambda = 1$ is equivalent to the traditional image captioning models. The result in Table 6.1 shows that there is a local maximum of λ at value equals to 0.2, and the performance goes down in both directions to $\lambda = 0$ and $\lambda = 1$. This validates the effectiveness of utilizing embedding into the image captioning framework.

Table 6.3: *The image captioning results in MSCOCO dataset [Lin et al., 2014] of the proposed Embedding-Guided Image Captioning approach, comparing with several existing approaches including Google NIC [Vinyals et al., 2015], M-RNN [Mao et al., 2015], LRCN [Donahue et al., 2015], MSR/CMU [Chen and Zitnick, 2015], Spatial_ATT [Xu et al., 2015], Semantic_ATT [You et al., 2016].*

Approach	Bleu_1	Bleu_2	Bleu_3	Bleu_4	Rouge_L	CIDEr	METEOR
Google NIC	0.666	0.451	0.304	0.203	-	-	-
M-RNN	0.67	0.49	0.35	0.25	-	-	-
LRCN	0.628	0.442	0.304	0.21	-	-	-
MSR/CMU	-	-	-	0.19	-	-	0.204
Spatial_ATT	0.718	0.504	0.357	0.250	-	-	0.230
Semantic_ATT	0.709	0.537	0.402	0.304	-	-	0.243
Ours	0.706	0.532	0.395	0.296	0.520	0.911	0.243

Then, we evaluate our Embedding-Guided Image Captioning framework with different beam sizes, and compare them to the baseline model Google NIC [Vinyals et al., 2015]. The results are shown in Table 6.2. As we see, with various beam sizes from 10 to 100, the performance of them are comparable, which implies that embedding can consistently improve the image captioning performance and push it to a very close range regardless of the beam size parameter. As we know, beam size is a difficult parameter to set in traditional approaches. Researchers generally set it empirically and many of them claim that 2 is the best beam size number. Thus, we also list the results of Google NIC baseline method, with beam size equals to 1 and 2, as shown in the last two rows of Table 6.2. As we see, the proposed Embedding-Guided Image Captioning consistently outperforms the best baseline performance, by 1.24% on average across all metrics.

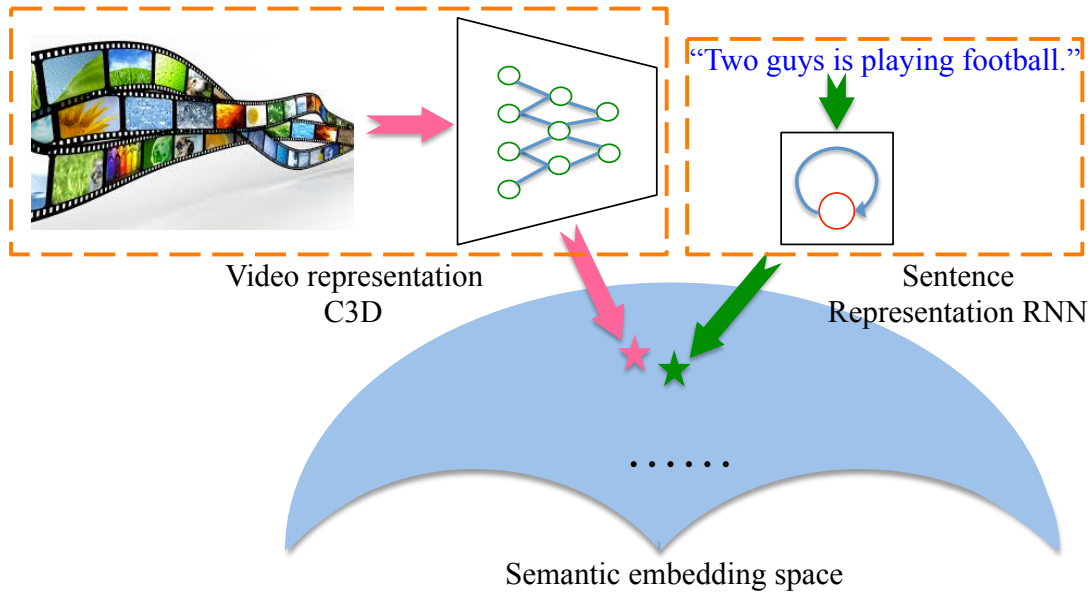


Figure 6.4: *The framework of the proposed video-sentence embedding.*

Finally we compare to state-of-the-art approaches. The results are shown in Table 6.3. As we see, our method outperforms the Google NIC baseline by a large margin. And we obtain comparable results as the best results so far. Semantic_ATT [You et al., 2016] reported the best performance in 4 metrics and we perform the best in 3 metrics. Note that the Semantic_ATT model utilizes semantic information online with complicated modeling process. And our method is based on the simply baseline Google NIC. This validates the advantage of utilizing joint image-text representation to improve multi-modal task such image captioning. We believe it can further boost the performance if we build upon the state-of-the-art model Semantic_ATT and utilize embedding to improve it.

6.2 Joint Video-Sentence Representation

Tasks involving videos and texts have been explored by many researchers [Venugopalan et al., 2015; Yao et al., 2015; Ballas et al., 2016; Pan et al., 2016]. Many of them

Table 6.4: *The video annotation results of joint video-sentence embedding in the MSR-VTT dataset [Xu et al., 2016], using different video features.*

Feature	recall@1	recall5	recall@10	med rank
fc6-mean	22.3	50.7	63.7	5
fc6-max	23.7	51.7	60.0	5
fc7-mean	19.3	44.3	58.7	7
fc7-max	23.3	48.0	61.7	7

follow the general image captioning framework as shown in Figure 6.2, and target on generating a description for a given video input.

We propose to learn a joint video-sentence representation and it can be utilized in various tasks involving both video and sentence inputs. As shown in Figure 6.4, we represent the input video by deep neural networks, and the corresponding sentence description is modeled by RNN. Finally the semantic embedding space of video and sentence is learned by pushing corresponding video and sentence pairs to be close to each other.

More specifically, we utilize the C3D [Tran et al., 2015] model to capture the video feature. And the bi-directional ranking loss function is employed to learn the embedding function, as shown in Equation 6.5.

With the learned joint video-sentence representation, we have applied it to two applications, *i.e.*, video annotation and text-based video retrieval.

6.2.1 Experiments on video annotation

We run experiments of video annotation in a large scale video description dataset, MSR-VTT dataset [Xu et al., 2016], which has around 140,000 video-description pairs. Video

Table 6.5: *The text-based video retrieval results of joint video-sentence embedding in the MSR-VTT dataset [Xu et al., 2016], using different video features.*

Feature	recall@1	recall@5	recall@10	med rank
fc6-mean	12.8	34.5	46.2	13
fc6-max	9.9	31.8	44.9	13
fc7-mean	12.7	32.8	45.7	13
fc7-max	11.2	31.2	44.4	14

annotation means given a query video and retrieve the sentence to annotate it. The results is shown in Table 6.4. We design four different video features based on the C3D network [Tran et al., 2015]. The first feature is obtained by extracting the fc6 layer feature vectors of all the 12-frame clips and do a mean pooling over them, denoted as fc6-mean. The second option is to extract the fc6 layer feature vectors of all the 12-frame clips and do a max pooling over them, denoted as fc6-max. The third one is to extract the fc7 layer feature vectors and do mean pooling, denoted as fc7-mean. The last one is to extract the fc7 layer feature vectors and do max pooling, denoted as fc7-max. And we employ four metrics to comprehensively evaluate the performance, `recall@1`, `recall@5`, `recall@10`, and `med rank`, where `recall@k` metric is the recall of the top- k annotations, `med rank` denotes the medium rank of ground truth descriptions of all the video queries.

As we can see in Table 6.4, the video features based on fc6 layer feature vectors perform better than fc7-mean and fc7-max in the video annotation task.

6.2.2 Experiments on text-based video retrieval

In the task of text-based video retrieval, we test on MSR-VTT dataset [Xu et al., 2016] and use the same four video features as above, *i.e.*, fc6-mean, fc6-max, fc7-mean, and fc7-max. The goal of this task is to retrieve related videos according to the text inputs.

The results are shown in Table 6.5. As we see, the fc6-mean feature outperforms all other features. Together with the result in Table 6.4, we find that fc6-mean is a good option to compute video feature.

6.3 Summary

In this chapter, we investigated joint image-sentence representation and joint video-sentence representation, for advanced visual and textual modalities such as videos and long texts. Given a learned joint image-sentence embedding representation, we have proposed a novel Embedding-Guide Image Captioning framework, which uses the joint embedding space to improve image captioning performance. And we have obtained very promising results and validated its effectiveness in MSCOCO dataset. Besides, we have proposed a basic framework for joint video-sentence representation learning. And we have evaluated four different video feature computing methods in the tasks of video annotation and text-based video retrieval.

CHAPTER 7

Conclusion and Future Research

7.1 Summary

Like humans who are intelligent enough to process information of multiple modalities, such as video, text, audio, *etc.*, teaching computers to jointly understand multi-modal information is a necessary and essential step towards artificial intelligence. And the joint representation of multi-modal information is critical to such step.

However, we are still far from this goal. The main challenges lie in the richness of visual and textual inputs as well as the large variations among them. In order to learn an effective joint representation, in this dissertation, we addressed such problem by different levels of modeling, including text concept modeling, image-level modeling, and object-level modeling.

Specifically, we first introduced a novel Gaussian Visual-Semantic Embedding (GVSE) model for text concept modeling by leveraging the visual information to model text concepts as density distributions other than single points in semantic space. Then, we proposed Multiple Instance Visual-Semantic Embedding (MIVSE) via image-level modeling, which discovers and maps the semantically meaningful image sub-regions to their corresponding text labels. Next, we presented a fine-grained object-level representation of images, Scene-Domain Active Part Models (SDAPM), that reconstructs and characterizes the 3D geometric statistics between objects parts in 3D scene-domain.

Finally, we explored advanced joint representations for video and long sentences, including joint image-sentence representation and joint video-sentence representation. The effectiveness of the proposed joint representations have been validated in various multi-modal tasks, *e.g.*, image classification, image annotation, text-based image retrieval, image captioning, video annotation, text-based video retrieval, *etc.*

7.2 Future Research Directions

The research of jointly understand multi-modal information is still in its infancy. More research effort should be made in this area, not only for the theoretical foundation, but also for more practical applications. Many interesting and important issues in joint multi-modal understanding should be investigated further.

In Chapter 3, we proposed a novel text concept modeling algorithm for joint image-text representation. We explored text concept modeling as Gaussian. And for effective training, we constrained the densities to be diagonal Gaussians. We noticed that there are things to improve. A straightforward improvement of our method is to model text concepts as more sophisticated density distributions, such as Gaussians with arbitrary covariances and Gaussian Mixture Models. Another future improvement over the proposed method is in the training process. We learned our model parameters by end-to-end training. However, with more complicated text modelings, such as GMM, iterative training can benefit the convergence.

A novel fine-grained object representation model, Scene-Domain Active Part Models, was presented in Chapter 5. We validated the effectiveness of our model in representing object in images. It is an interesting future problem to study the functionality of utilizing the object modeling in joint image-text representation. Such model by utilizing object models in joint representation can be time consuming, but it should be more precise and accurate in understanding detailed parts of images. Therefore, more

subtle tasks that require detailed understanding of images could be benefited from such approach.

Finally, we can apply different levels of modeling, as proposed in this dissertation, to joint representation of advanced visual and textual inputs, such as video and long sentences. And we can apply the embedding representation to improve many other multi-modal tasks, similar to the image captioning.

The techniques developed in this dissertation are general enough to be extended to many other research topics, such as the joint representation of other modalities such as audio, depth data, *etc.* The author hopes that this study could motive increasingly more investigation of joint multi-modal understanding and serve as an important step toward realizing the dream of making computers intelligent.

Appendix A

Derivation of Inference Solution of Scene-Domain

Active Part Models

This appendix present the derivation of the closed-form optimal solution \mathbf{m}^* of Scene-Domain Active Part Models Inference shown in Equation 5.6. As discussed in Section 5.4 of Chapter 5, the inference corresponds to maximizing the score function $S(\mathbf{I}, \mathbf{p}, \mathbf{m})$ shown as below:

$$\begin{aligned} S(\mathbf{I}) &= \max_{\mathbf{p}, \mathbf{m}} S(\mathbf{I}, \mathbf{p}, \mathbf{m}) \\ &= \max_{\mathbf{p}, \mathbf{m}} \sum_{i \in \mathcal{V}} S_i(\mathbf{I}, p_i) + (\Delta \mathbf{p} - f(\mathbf{mB}))^T \boldsymbol{\tau} + (\Delta \mathbf{p} - f(\mathbf{mB}))^T \boldsymbol{\Lambda} (\Delta \mathbf{p} - f(\mathbf{mB})), \end{aligned} \quad (\text{A.1})$$

where $f(\mathbf{mB}) = (\mathbf{e}_1 \mathbf{mB} \mathbf{A} + \mathbf{e}_2 \mathbf{mB} \hat{\mathbf{A}})^T$.

Section 5.4 of this dissertation introduces a coordinate descent approach that is used to compute $S(\mathbf{I})$ above. In the second step of our coordinate descent approach, when fixing \mathbf{p} , the optimal \mathbf{m}^* that maximizes $S(\mathbf{I}, \mathbf{p}, \mathbf{m})$ satisfies:

$$\left. \frac{\partial S(\mathbf{I}, \mathbf{p}, \mathbf{m})}{\partial \mathbf{m}} \right|_{\mathbf{m}=\mathbf{m}^*} = 0. \quad (\text{A.2})$$

Since $\sum_{i \in \mathcal{V}} S_i(\mathbf{I}, p_i)$ is constant to \mathbf{m} , we have:

$$\frac{\partial S(\mathbf{I}, \mathbf{p}, \mathbf{m})}{\partial \mathbf{m}} = \frac{\partial g(\mathbf{m})}{\partial \mathbf{m}}, \quad (\text{A.3})$$

where $g(\mathbf{m})$ is defined as follows:

$$\begin{aligned}
g(\mathbf{m}) &= (\Delta\mathbf{p} - f(\mathbf{m}\mathbf{B}))^T \boldsymbol{\tau} + (\Delta\mathbf{p} - f(\mathbf{m}\mathbf{B}))^T \boldsymbol{\Lambda} (\Delta\mathbf{p} - f(\mathbf{m}\mathbf{B})) \\
&= \left(\Delta\mathbf{p} - (\mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A} + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}}) \right)^T \boldsymbol{\tau} \\
&\quad + \left(\Delta\mathbf{p} - (\mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A} + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}}) \right)^T \boldsymbol{\Lambda} \left(\Delta\mathbf{p} - (\mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A} + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}}) \right)^T \\
&= (\mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A} + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}}) \boldsymbol{\Lambda} (\mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A} + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}})^T + (\Delta\mathbf{p}^T \boldsymbol{\tau} + \Delta\mathbf{p}^T \boldsymbol{\Lambda} \Delta\mathbf{p}) \\
&\quad - (\mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A} + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}}) (\boldsymbol{\Lambda} \Delta\mathbf{p} + \boldsymbol{\tau}) - \Delta\mathbf{p}^T \boldsymbol{\Lambda} (\mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A} + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}})^T \\
&= \mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A}\boldsymbol{\Lambda}\boldsymbol{\Lambda}^T\mathbf{B}^T\mathbf{m}^T\mathbf{e}_1^T + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}}\boldsymbol{\Lambda}\widehat{\mathbf{A}}^T\mathbf{B}^T\mathbf{m}^T\mathbf{e}_2^T \\
&\quad + \mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A}\boldsymbol{\Lambda}\widehat{\mathbf{A}}^T\mathbf{B}^T\mathbf{m}^T\mathbf{e}_2^T + \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}}\boldsymbol{\Lambda}\mathbf{A}^T\mathbf{B}^T\mathbf{m}^T\mathbf{e}_1^T \\
&\quad - \mathbf{e}_1\mathbf{m}\mathbf{B}\mathbf{A} (\boldsymbol{\Lambda}\Delta\mathbf{p} + \boldsymbol{\tau}) - \mathbf{e}_2\mathbf{m}\mathbf{B}\widehat{\mathbf{A}} (\boldsymbol{\Lambda}\Delta\mathbf{p} + \boldsymbol{\tau}) - \Delta\mathbf{p}^T \boldsymbol{\Lambda}\mathbf{A}^T\mathbf{B}^T\mathbf{m}^T\mathbf{e}_1^T \\
&\quad - \Delta\mathbf{p}^T \boldsymbol{\Lambda}\widehat{\mathbf{A}}^T\mathbf{B}^T\mathbf{m}^T\mathbf{e}_2^T + \Delta\mathbf{p}^T \boldsymbol{\tau} + \Delta\mathbf{p}^T \boldsymbol{\Lambda} \Delta\mathbf{p}
\end{aligned}$$

According to the matrix calculus, for a matrix variable \mathbf{X} and a scalar function $f(\mathbf{X})$, we have the theorem for matrix derivative which is as follows:

$$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \begin{cases} \mathbf{A}^T \mathbf{B}^T & , \quad \text{if } f(\mathbf{X}) = \mathbf{A}\mathbf{X}\mathbf{B} \\ \mathbf{B}\mathbf{A} & , \quad \text{if } f(\mathbf{X}) = \mathbf{A}\mathbf{X}^T\mathbf{B} \\ \mathbf{C}\mathbf{A}\mathbf{X}\mathbf{B} + \mathbf{A}^T\mathbf{C}^T\mathbf{X}\mathbf{B}^T & , \quad \text{if } f(\mathbf{X}) = \mathbf{A}\mathbf{X}\mathbf{B}\mathbf{X}^T\mathbf{C} \end{cases} \quad (\text{A.4})$$

Thus according to the Theorem in Equation A.4, for the scalar function $g(\mathbf{m})$ above, its derivative to the $2 \times 3K$ matrix \mathbf{m} is:

$$\begin{aligned}
\frac{\partial g(\mathbf{m})}{\partial \mathbf{m}} &= \mathbf{e}_1^T \mathbf{e}_1 \mathbf{m} \mathbf{B} \mathbf{A} (\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T) \mathbf{A}^T \mathbf{B}^T + \mathbf{e}_2^T \mathbf{e}_2 \mathbf{m} \mathbf{B} \widehat{\mathbf{A}} (\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T) \widehat{\mathbf{A}}^T \mathbf{B}^T \\
&\quad + \mathbf{e}_1^T \mathbf{e}_2 \mathbf{m} \mathbf{B} \widehat{\mathbf{A}} (\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T) \mathbf{A}^T \mathbf{B}^T + \mathbf{e}_2^T \mathbf{e}_1 \mathbf{m} \mathbf{B} \mathbf{A} (\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T) \widehat{\mathbf{A}}^T \mathbf{B}^T \\
&\quad - \mathbf{e}_1^T ((\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T) \Delta\mathbf{p} + \boldsymbol{\tau})^T \mathbf{A}^T \mathbf{B}^T - \mathbf{e}_2^T ((\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T) \Delta\mathbf{p} + \boldsymbol{\tau})^T \widehat{\mathbf{A}}^T \mathbf{B}^T
\end{aligned} \quad (\text{A.5})$$

As we defined in Section 5.3.1 of Chapter 5, $\boldsymbol{\Lambda}$ is a $2|\mathcal{E}| \times 2|\mathcal{E}|$ diagonal matrix, and \mathbf{A} , $\widehat{\mathbf{A}}$ are $|\mathcal{E}| \times 2|\mathcal{E}|$ constant matrices, which concatenate the $|\mathcal{E}| \times |\mathcal{E}|$ zero matrix with the $|\mathcal{E}| \times |\mathcal{E}|$ identity matrix. Therefore, $\mathbf{A}(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\widehat{\mathbf{A}}^T = \widehat{\mathbf{A}}(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\mathbf{A}^T = \mathbf{0}$. As a

result, Equation A.5 is simplified to:

$$\begin{aligned} \frac{\partial g(\mathbf{m})}{\partial \mathbf{m}} &= \mathbf{e}_1^T \mathbf{e}_1 \mathbf{m} \mathbf{B} \mathbf{A} (\mathbf{\Lambda} + \mathbf{\Lambda}^T) \mathbf{A}^T \mathbf{B}^T + \mathbf{e}_2^T \mathbf{e}_2 \mathbf{m} \mathbf{B} \widehat{\mathbf{A}} (\mathbf{\Lambda} + \mathbf{\Lambda}^T) \widehat{\mathbf{A}}^T \mathbf{B}^T \\ &\quad - \mathbf{e}_1^T ((\mathbf{\Lambda} + \mathbf{\Lambda}^T) \Delta \mathbf{p} + \boldsymbol{\tau})^T \mathbf{A}^T \mathbf{B}^T - \mathbf{e}_2^T ((\mathbf{\Lambda} + \mathbf{\Lambda}^T) \Delta \mathbf{p} + \boldsymbol{\tau})^T \widehat{\mathbf{A}}^T \mathbf{B}^T \end{aligned} \quad (\text{A.6})$$

And according to Equation A.2 and Equation A.3, the optimal solution \mathbf{m}^* must satisfy the condition that Equation A.6 = 0, *i.e.*:

$$\begin{aligned} &\mathbf{e}_1^T \mathbf{e}_1 \mathbf{m} \mathbf{B} \mathbf{A} (\mathbf{\Lambda} + \mathbf{\Lambda}^T) \mathbf{A}^T \mathbf{B}^T + \mathbf{e}_2^T \mathbf{e}_2 \mathbf{m} \mathbf{B} \widehat{\mathbf{A}} (\mathbf{\Lambda} + \mathbf{\Lambda}^T) \widehat{\mathbf{A}}^T \mathbf{B}^T \\ &= \mathbf{e}_1^T ((\mathbf{\Lambda} + \mathbf{\Lambda}^T) \Delta \mathbf{p} + \boldsymbol{\tau})^T \mathbf{A}^T \mathbf{B}^T + \mathbf{e}_2^T ((\mathbf{\Lambda} + \mathbf{\Lambda}^T) \Delta \mathbf{p} + \boldsymbol{\tau})^T \widehat{\mathbf{A}}^T \mathbf{B}^T \end{aligned} \quad (\text{A.7})$$

Let us denote $\mathbf{m}_{(2 \times 3K)}^* = \begin{pmatrix} \mathbf{m}_1^* \\ \mathbf{m}_2^* \end{pmatrix}$ where \mathbf{m}_1^* , \mathbf{m}_2^* are the two rows of \mathbf{m}^* , and as de-

fined in Section 5.3.2 of Chapter 5, $\mathbf{e}_1 = (1, 0)$, $\mathbf{e}_2 = (0, 1)$, thus $\mathbf{e}_1^T \mathbf{e}_1 \mathbf{m}^* = \begin{pmatrix} \mathbf{m}_1^* \\ \mathbf{0} \end{pmatrix}$,

$\mathbf{e}_2^T \mathbf{e}_2 \mathbf{m}^* = \begin{pmatrix} \mathbf{0} \\ \mathbf{m}_2^* \end{pmatrix}$. Besides, let us denote the right hand side of Equation A.7,

i.e., $\mathbf{e}_1^T ((\mathbf{\Lambda} + \mathbf{\Lambda}^T) \Delta \mathbf{p} + \boldsymbol{\tau})^T \mathbf{A}^T \mathbf{B}^T + \mathbf{e}_2^T ((\mathbf{\Lambda} + \mathbf{\Lambda}^T) \Delta \mathbf{p} + \boldsymbol{\tau})^T \widehat{\mathbf{A}}^T \mathbf{B}^T$, as a matrix \mathbf{H} , and suppose $\mathbf{H}_{(2 \times 3K)} = \begin{pmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \end{pmatrix}$ where \mathbf{H}_1 , \mathbf{H}_2 are the two rows of \mathbf{H} . Thus

Equation A.7 is equivalent to:

$$\begin{pmatrix} \mathbf{m}_1^* \mathbf{B} \mathbf{A} (\mathbf{\Lambda} + \mathbf{\Lambda}^T) \mathbf{A}^T \mathbf{B}^T \\ \mathbf{m}_2^* \mathbf{B} \widehat{\mathbf{A}} (\mathbf{\Lambda} + \mathbf{\Lambda}^T) \widehat{\mathbf{A}}^T \mathbf{B}^T \end{pmatrix} = \begin{pmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \end{pmatrix} \quad (\text{A.8})$$

Thus, we have:

$$\begin{cases} \mathbf{m}_1^* \mathbf{B} \mathbf{A} (\mathbf{\Lambda} + \mathbf{\Lambda}^T) \mathbf{A}^T \mathbf{B}^T = \mathbf{H}_1 \\ \mathbf{m}_2^* \mathbf{B} \widehat{\mathbf{A}} (\mathbf{\Lambda} + \mathbf{\Lambda}^T) \widehat{\mathbf{A}}^T \mathbf{B}^T = \mathbf{H}_2 \end{cases} \quad (\text{A.9})$$

Therefore, the optimal solution \mathbf{m}^* is as follows:

$$\begin{cases} \mathbf{m}_1^* = \mathbf{H}_1 (\mathbf{B}\mathbf{A}(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\mathbf{A}^T\mathbf{B}^T)^{-1} \\ \mathbf{m}_2^* = \mathbf{H}_2 (\mathbf{B}\widehat{\mathbf{A}}(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\widehat{\mathbf{A}}^T\mathbf{B}^T)^{-1} \end{cases} \quad (\text{A.10})$$

where $\mathbf{H}_1, \mathbf{H}_2$ are the two rows of matrix $\mathbf{H}_{(2 \times 3K)} = \mathbf{e}_1^T ((\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\boldsymbol{\Delta}\mathbf{p} + \boldsymbol{\tau})^T \mathbf{A}^T\mathbf{B}^T + \mathbf{e}_2^T ((\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^T)\boldsymbol{\Delta}\mathbf{p} + \boldsymbol{\tau})^T \widehat{\mathbf{A}}^T\mathbf{B}^T$.

BIBLIOGRAPHY

- Akata, Z., Perronnin, F., Harchaoui, Z., and Schmid, C. (2013). Label-embedding for attribute-based classification. In *CVPR*. 28
- Akhter, I., Sheikh, Y. A., Khan, S., and Kanade, T. (2008). Nonrigid structure from motion in trajectory space. In *NIPS*. 49, 55, 61
- Arie-Nachimson, M. and Basri, R. (2009). Constructing implicit 3d shape models for pose estimation. In *ICCV*. xiii, 50, 52, 61, 62, 70, 71
- Azizpour, H. and Laptev, I. (2012). Object detection using strongly-supervised deformable part models. In *ECCV*. xiii, 47, 48, 50, 51, 58, 60, 61, 63, 66
- Ballas, N., Yao, L., Pal, C., and Courville, A. (2016). Delving deeper into convolutional networks for learning video representations. In *ICLR*. 82
- Barnard, K. and Forsyth (2001). Learning the semantics of words and pictures. In *ICCV*. 3
- Baum, L. E. and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state markov chains. *Bulletin of the American Mathematical Society*. 3, 75
- Bengio, Y., Ducharme, R., and Vincent, P. (2003a). A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155. 3
- Bengio, Y., Ducharme, R., and Vincent, P. (2003b). A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155. 28
- Blei, D. and Jordan, M. (2003). Modeling annotated data. In *ACM SIGIR*. 3
- Bourdev, L., Maji, S., Brox, T., and Malik, J. (2010). Detecting people using mutually consistent poselet activations. In *ECCV*. 50, 61, 63

- Breitenstein, M. D., Reichlin, F., Leibe, B., Koller-Meier, E., and Gool, L. V. (2009). Robust tracking-by-detection using a detector confidence particle filter. In *ICCV*. 47
- Chen, Q., Song, Z., Hua, Y., Huang, Z., and Yan, S. (2012). Hierarchical matching with side information for image classification. In *CVPR*. 2, 27
- Chen, X., Mottaghi, R., Liu, X., Cho, N.-G., Lee, S.-W., Fidler, S., Urtasun, R., and Yuille, A. (2014). Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*. 48, 50, 51, 61
- Chen, X. and Yuille, A. (2014). Articulated pose estimation by a graphical model with image dependent pairwise relations. In *NIPS*. 62
- Chen, X. and Zitnick, C. L. (2015). Mind’s eye: A recurrent visual representation for image caption generation. In *CVPR*. xiv, 77, 81
- Chen, Y., Zhu, L., and Yuille, A. (2010). Active mask hierarchies for object detection. In *ECCV*. 50
- Chua, T.-S., Tang, J., Hong, R., Li, H., Luo, Z., and Zheng, Y.-T. (2009). Nus-wide: a real-world web image database from national university of singapore. In *ACM International Conference on Image and Video Retrieval*. 36, 43
- Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *Arxiv report 1412.3555*. 3, 75, 76
- Coates, A. and Ng, A. (2011). The importance of encoding versus training with sparse coding and vector quantization. In *ICML*. 28
- Dai, Y., Li, H., and He, M. (2012). A simple prior-free method for non-rigid structure-from-motion factorization. In *CVPR*. 49, 55, 61, 62

- Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*. 2, 50
- Deng, J., Ding, N., Jia, Y., Frome, A., Murphy, K., Bengio, S., Li, Y., Neven, H., and Adam, H. (2014). Large-scale object classification using label relation graphs. In *ECCV*. 24
- Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Fei-Fei, L. (2009). Imagenet: a large-scale hierarchical image database. In *CVPR*. 11, 24
- Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *EACL Workshop on Statistical Machine Translation*. 80
- Dietterich, T. G., Lathrop, R. H., and Lozano-Perez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89:31–71. 31
- Donahue, J., Hendricks, L. A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*. xiv, 77, 81
- Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338. 61
- Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Dollar, P., Gao, J., He, X., Mitchell, M., Platt, J. C., Zitnick, C. L., and Zweig, G. (2015). From captions to visual concepts and back. In *CVPR*. 77
- Farhadi, A., Endres, I., Hoiem, D., and Forsyth, D. (2009). Describing objects by their attributes. In *CVPR*. 28
- Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object detec-

- tion with discriminatively trained part based models. *TPAMI*, 32:1627–1645. 47, 50, 53, 57, 58, 61, 62, 63
- Felzenszwalb, P. F. and Huttenlocher, D. P. (2005). Pictorial structure for object recognition. *IJCV*, 61(1):55–79. 48
- Fidler, S., Dickinson, S., and Urtasun, R. (2012). 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In *NIPS*. 48, 51
- Fidler, S., Mottaghi, R., Yuille, A., and Urtasun, R. (2013). Bottom-up segmentation for top-down detection. In *CVPR*. 50, 62, 63
- Fischler, M. and Elschlager, R. (1973). The representation and matching of pictorial structures. *IEEE Trans. on Computers*, C-22(1):67 – 92. 52
- Frome, A., Corrado, G., Shlens, J., Bengio, S., Dean, J., Ranzato, M., and Mikolov, T. (2013). Devise: A deep visual-semantic embedding model. In *NIPS*. 4, 8, 9, 11, 13, 14, 17, 18, 24, 26, 28, 29, 36, 42, 43, 44
- Girshick, R. (2015). Fast r-cnn. In *ICCV*. 2, 27, 34
- Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014a). Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*. 34, 47, 51, 64
- Girshick, R., Felzenszwalb, P., and Mcallester, D. (2011). Object detection with grammar models. In *NIPS*. 51
- Girshick, R., Iandola, F., Darrell, T., and Malik, J. (2014b). Deformable part models are convolutional neural networks. *CoRR*, abs/1409.5403. 51, 61, 62, 63
- Gong, Y., Jia, Y., Leung, T. K., Toshev, A., and Ioffe, S. (2014a). Deep convolutional ranking for multilabel image annotation. In *ICLR*. 8, 11, 26, 27, 29, 36, 37, 39, 42
- Gong, Y., Ke, Q., Isard, M., and Lazebnik, S. (2014b). A multi-view embedding space

- for modeling internet images, tags, and their semantics. *International Journal of Computer Vision*. 3
- Guillaumin, M., Mensink, T., Verbeek, J., and Schmid, C. (2009). Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *ICCV*. 8, 27
- Hardoon, D., Szedmak, S., and Shawe-Taylor, J. (2004). Canonical correlation analysis: an overview with application to learning methods. *Neural Computation*. 3
- Hejrati, M. and Ramanan, D. (2012). Analyzing 3d objects in cluttered images. In *NIPS*. xi, xiii, 50, 52, 59, 61, 68, 69, 70, 71
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780. 3, 75
- Hoiem, D., Efros, A. A., and Hebert, M. (2008). Closing the loop in scene interpretation. In *CVPR*. 47
- Hotelling, H. (1936). Relations between two sets of variables. *Biometrika*. 3
- Hu, H., Zhou, G.-T., Deng, Z., Liao, Z., and Mori, G. (2016). Learning structured inference neural networks with label relations. In *CVPR*. 27
- Hwang, S. J. and Grauman, K. (2011). Learning the relative importance of objects from tagged images for retrieval and cross-modal search. *International Journal of Computer Vision*. 3
- Ionescu, C., Papava, D., Olaru, V., and Sminchisescu, C. (2014). Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 36(7):1325–1339. 61
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S.,

- and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv:1408.5093*. 17, 36, 62
- Johnson, J., Ballan, L., and Fei-Fei, L. (2015). Love thy neighbors: Image annotation by exploiting image metadata. In *ICCV*. 27
- Johnson, J., Karpathy, A., and Fei-Fei, L. (2016). Denscap: Fully convolutional localization networks for dense captioning. In *CVPR*. 77
- Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *CVPR*. 4, 13, 27, 42, 80
- Karpathy, A., Joulin, A., and Fei-Fei, L. (2014). Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*. 27, 42
- Kiros, R., Salakhutdinov, R., and Zemel, R. S. (2015). Unifying visual-semantic embeddings with multimodal neural language models. In *TACL*. 4, 8, 9, 11, 13, 77
- Krähenbühl, P. and Koltun, V. (2014). Geodesic object proposals. In *ECCV*. 26, 34, 35, 40
- Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS*. 2, 24, 27, 37, 51, 75
- Lai, Z., Zhu, J., Ren, Z., Liu, W., and Yan, B. (2010). Arbitrary directional edge encoding schemes for the operational rate-distortion optimal shape coding framework. In *DCC*. 47
- Lampert, C. H., Nickisch, H., and Harmeling, S. (2009). Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*. 28
- Laptev, I. and Lindeberg, T. (2003). Space-time interest points. In *ICCV*. 2
- LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel,

- L. (1990). Handwritten digit recognition with a back-propagation network. In *NIPS*. 2, 27
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*. 51
- Li, X., Guo, Y., and Schuurmans, D. (2015). Semi-supervised zero-shot classification with label representation learning. In *ICCV*. 28
- Lim, J. J., Khosla, A., and Torralba, A. (2014). Fpm: Fine pose parts-based model with 3d cad models. In *ECCV*. 48, 51
- Lin, C.-Y. and Och, F. J. (2004). Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *ACL*. 80
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollar, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *ECCV*. xiv, 79, 80, 81
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110. 2
- Ma, Y., Soatto, S., Kosecka, J., and Sastry, S. (2004). *An invitation to 3-d vision: from images to geometric models*. Springer Verlag. 55
- Makadia, A., Pavlovic, V., and Kumar, S. (2008). A new baseline for image annotation. In *ECCV*. 8, 27
- Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., and Yuille, A. L. (2015). Deep captioning with multimodal recurrent neural networks (m-rnn). In *ICLR*. xiv, 11, 77, 81

- Marr, D. (1981). *Artificial intelligence: a personal view*. MIT Press. [1](#)
- Mensink, T., Verbeek, J., Perronnin, F., and Csurka, G. (2012). Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. In *ECCV*. [28](#)
- Mikolov, T., Deoras, A., Kombrink, S., Burget, L., and Cernock, J. (2011). Empirical evaluation and combination of advanced language modeling techniques. In *Inter-speech*. [3](#)
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *NIPS*. [3](#), [9](#), [13](#), [14](#), [29](#)
- Norouzi, M., Mikolov, T., Bengio, S., Singer, Y., Shlens, J., Frome, A., Corrado, G., and Dean, J. (2014). Zero-shot learning by convex combination of semantic embeddings. In *ICLR*. [4](#), [8](#), [9](#), [11](#), [13](#), [14](#), [24](#), [28](#), [29](#)
- Palatucci, M., Pomerleau, D., Hinton, G. E., and Mitchell, T. M. (2009). Zero-shot learning with semantic output codes. In *NIPS*. [28](#)
- Pan, Y., Mei, T., Yao, T., Li, H., and Rui, Y. (2016). Jointly modeling embedding and translation to bridge video and language. In *CVPR*. [82](#)
- Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *ACL*. [80](#)
- Parikh, D. and Grauman, K. (2011). Relative attributes. In *ICCV*. [28](#)
- Parkhi, O., Vedaldi, A., Jawahar, C., and Zisserman, A. (2011). The truth about cats and dogs. In *ICCV*. [50](#), [61](#), [63](#)

- Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*. 3, 9, 13, 14, 29, 33
- Pepik, B., Gehler, P., Stark, M., and Schiele, B. (2012). 3d²pm - 3d deformable part models. In *ECCV*. 51
- Perronnin, F. and Dance, C. (2007). Fisher kernels on visual vocabularies for image categorization. In *CVPR*. 2, 27
- Rajaraman, A. and Ullman, J. D. (2011). *Mining of Massive Datasets*. Cambridge University Press. 2
- Rasiwasia, N., Pereira, J. C., Coviello, E., Doyle, G., Lanckriet, G., Levy, R., and Vasconcelos, N. (2010). A new approach to cross-modal multimedia retrieval. In *ACM Multimedia*. 3
- Ren, Z., Jin, H., Lin, Z., Fang, C., and Yuille, A. (2015a). Multi-instance visual-semantic embedding. In *arXiv preprint arXiv:1512.06963*. 4, 8, 9, 11, 13, 14, 21
- Ren, Z., Jin, H., Lin, Z., Fang, C., and Yuille, A. (2016). Joint image-text representation by gaussian visual-semantic embedding. In *ACM Multimedia*. 4
- Ren, Z., Meng, J., and Yuan, J. (2011a). Depth camera based hand gesture recognition and its applications in human-computer-interaction. In *ICICS*. 47
- Ren, Z., Meng, J., Yuan, J., and Zhang, Z. (2011b). Robust hand gesture recognition with kinect sensor. In *ACM Multimedia*. 47
- Ren, Z., Wang, C., and Yuille, A. (2015b). Scene-domain active part models for object representation. In *ICCV*. 2, 27
- Ren, Z., Yuan, J., Li, C., and Liu, W. (2011c). Minimum near-convex decomposition for robust shape representation. In *ICCV*. 47

- Ren, Z., Yuan, J., and Liu, W. (2013a). Minimum near-convex shape decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 47
- Ren, Z., Yuan, J., Meng, J., and Zhang, Z. (2013b). Robust part-based hand gesture recognition using kinect sensor. *IEEE Trans. on Multimedia*, 15(5):1110–1120. 1, 47
- Ren, Z., Yuan, J., and Zhang, Z. (2011d). Robust hand gesture recognition based on finger-earth mover’s distance with a commodity depth camera. In *ACM Multimedia*. 47
- Rohrbach, M., Stark, M., and Schiele, B. (2011). Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *CVPR*. 28
- Rui, Y., Huang, T. S., and Chang, S.-F. (1999). Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation*. 11
- Savalle, P.-A., Tsogkas, S., Papandreou, G., and Kokkinos, I. (2013). Deformable part models with cnn features. In *3rd Parts and Attributes Workshop, ECCV*. 51, 63, 64
- Schwenk, H. (2007). Continuous space language models. *Computer Speech and Language*. 3
- Shrivastava, A. and Gupta, A. (2013). Building part-based object detectors via 3d geometry. In *ICCV*. 51
- Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *ICLR*. 2, 24, 27, 75
- Socher, R., Ganjoo, M., Sridhar, H., Bastani, O., Manning, C. D., and Ng, A. Y. (2013). Zero-shot learning through cross-modal transfer. In *NIPS*. 28

- Sun, M. and Savarese, S. (2011). Articulated part-based model for joint object detection and pose estimation. In *ICCV*. 48, 51
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *CVPR*. 2, 16, 24, 27, 33, 37, 75
- Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. (2015). Learning spatiotemporal features with 3d convolutional networks. In *ICCV*. 2, 75, 83, 84
- Trulls, E., Tsogkas, S., Kokkinos, I., Sanfeliu, A., and Moreno, F. (2014). Segmentation-aware deformable part models. In *CVPR*. 50
- Vedantam, R., Zitnick, C. L., and Parikh, D. (2015). Consensus-based image description evaluation. In *CVPR*. 80
- Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., and Saenko, K. (2015). Sequence to sequence - video to text. In *ICCV*. 82
- V.Ferrari, M.Marin-Jimenez, and A.Zisserman (2008). Progressive search space reduction for human pose estimation. In *CVPR*. 64
- Vilnis, L. and McCallum, A. (2015). Word representations via gaussian embedding. In *ICLR*. 4, 13
- Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *CVPR*. xiv, 11, 77, 80, 81
- Wan, L., Eigen, D., and Fergus, R. (2014). End-to-end integration of a convolutional network, deformable parts model and non-maximum suppression. *CoRR*, abs/1411.5309. 51, 63, 64

- Wang, X., Yang, M., Zhu, S., and Lin, Y. (2013). Regionlets for generic object detection. In *ICCV*. 50, 61, 63
- Weston, J., Benjio, S., and Usunier, N. (2011). Wsabie: scaling up to large vocabulary image annotation. In *IJCAI*. 27, 32, 42
- Xu, J., Mei, T., Yao, T., and Rui, Y. (2016). Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*. xiv, 83, 84, 85
- Xu, K., Ba, J., Kiros, R., Courville, A., Salakhutdinov, R., Zemel, R., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *ICML*. xiv, 77, 81
- Yang, J., Yu, K., Gong, Y., and Huang, T. (2009). Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*. 2, 27
- Yang, W., Wang, Y., and Mori, G. (2010). Recognizing human actions from still images with latent poses. In *CVPR*. 47
- Yang, Y. and Ramanan, D. (2012). Articulated human detection with flexible mixtures of parts. *TPAMI*, 35:2878 – 2890. xiii, 47, 48, 50, 51, 53, 57, 62, 68
- Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., and Courville, A. (2015). Describing videos by exploiting temporal structure. In *ICCV*. 82
- You, Q., Jin, H., Wang, Z., Fang, C., and Luo, J. (2016). Image captioning with semantic attention. In *CVPR*. xiv, 77, 81, 82
- Zeiler, M. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *ECCV*. 2, 27
- Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. (2014). Learning deep features for scene recognition using places database. In *NIPS*. 17, 18, 43

Zhu, Y., Urtasun, R., Salakhutdinov, R., and Fidler, S. (2015). segdeepm: Exploiting segmentation and context in deep neural networks for object detection. In *CVPR*. 47, 51, 64