Faculty of Engineering and Information Technology
University of Technology Sydney

Towards Automatic Construction of Diverse, High-quality Image Dataset

A thesis submitted in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

by

Yazhou Yao

September 2018

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate

Production Note:
Signature removed prior to publication.

Acknowledgments

Foremost, I would like to express my sincere gratitude to my supervisors Prof. Jian Zhang for the continuous support of my Ph.D study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis.

I also would like to appreciate my co-supervisor Prof. Zhenming Tang and Fumin Shen for providing me with continuous support throughout my Ph.D study and research. Without their professional guidance and persistent help, this thesis would not have been possible.

I am very grateful to Senior Researcher Xian-sheng Hua at Alibaba Research, and I have benefited greatly from every high-quality group meeting. Their cutting-edge research perspectives and solid theoretical foundation make me open-minded and inspire me to work harder.

I thank my fellow labmates in Global Big Data Technologies Center: Yucheng Wang, shangrong Huang for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last three years.

Last but not the least, I would like to thank my family: my wife and my parents, for their unconditional support, both financially and emotionally throughout the whole Ph.D studying.

Yazhou Yao
March 2018 @ UTS

Contents

List of Figures

List of Tables

List of Publications

Papers Published

- **Yazhou Yao**, Fumin Shen, Jian Zhang, Li Liu, Zhenmin Tang, and Ling Shao (2018), Extracting Privileged Information for Enhancing Classifier Learning, *in* 'Proceedings of IEEE Transactions on Image Processing (**TIP**)', doi=10.1109/TIP.2018.2869721.

- **Yazhou Yao**, Fumin Shen, Jian Zhang, Li Liu, Zhenmin Tang, and Ling Shao (2018), Discovering and Distinguishing Multiple Visual Senses for Web Learning, *in* 'Proceedings of IEEE Transactions on Multimedia (**TMM**)', doi=10.1109/TMM.2018.2847248.

- **Yazhou Yao**, Jian Zhang, Fumin Shen, Xian-Sheng Hua, Wankou Yang and Zhenmin Tang (2018), Extracting Privileged Information from Untagged Corpora for Classifier Learning, *in* 'Proceedings of the International Joint Conference on Artificial Intelligence (**IJCAI 18**)', pp, 1085-1091.

- **Yazhou Yao**, Jian Zhang, Fumin Shen, Wankou Yang, Pu Huang and Zhenmin Tang (2018), Discovering and Distinguishing Multiple Visual Senses for Polysemous Words. *in* 'Proceedings of the AAAI Conference on Artificial Intelligence (**AAAI 18**)', pp, 523-530.

- **Yazhou Yao**, Jian Zhang, Fumin Shen, Xian-Sheng Hua, Jingsong Xu and Zhenmin Tang (2017), Exploiting Web Images for Dataset

Construction: A Domain Robust Approach. *in* 'Proceedings of IEEE Transactions on Multimedia (**TMM**)', pp. 1771-1784.

- **Yazhou Yao**, Xiansheng Hua, Fumin Shen, Jian Zhang and Zhenmin Tang (2016), A Domain Robust Approach for Image Dataset Construction. *in* 'Proceedings of the ACM International Conference on Multimedia (**ACM MM 16**)', pp, 212-216.

- **Yazhou Yao**, Jian Zhang, Fumin Shen, Xian-Sheng Hua, Jingsong Xu and Zhenmin Tang (2016), Automatic Image Dataset Construction with Multiple Textual Metadata. *in* 'Proceedings of the IEEE International Conference on Multimedia and Expo (**ICME 16**)', pp, 1-6.

- **Yazhou Yao**, Jian Zhang, Fumin Shen, Xian-Sheng Hua, Jingsong Xu and Zhenmin Tang (2017), A New Web-supervised Method for Image Dataset Construction. *in* 'Proceedings of Neurocomputing', pp. 23-31.

- **Yazhou Yao**, Wankou Yang, Pu Huang, Qiong Wang, Yunfei Cai and Zhenmin Tang (2018), Exploiting Textual and Visual Features for Image Categorization. *in* 'Proceedings of Pattern Recognition Letters', doi=https://doi.org/10.1016/j.patrec.2018.05.028.

- **Yazhou Yao**, Jian Zhang, Xian-Sheng Hua, Fumin Shen and Zhenmin Tang (2016), Extracting Visual Knowledge from the Internet: Making Sense of Image Data. *in* 'Proceedings of the International Conference on Multimedia Modelling', pp, 862-873.

Papers Under Review

- **Yazhou Yao**, Jian Zhang, Fumin Shen, Dongxiang Zhang, Zhenmin Tang, and Heng Tao Shen (2017), Towards Automatic Construction of Diverse, High-quality Image Dataset, IEEE Transactions on Knowledge and Data Engineering (**TKDE**) (**major revision**).

Abstract

The availability of labeled image datasets has been shown critical for high-level image understanding, which continuously drives the progress of feature designing and models developing. However, the process of manual labeling is both time-consuming and labor-intensive. To reduce the cost of manual annotation, there has been increased research interest in automatically constructing image datasets by exploiting web images. Datasets constructed by existing methods tend to suffer from the disadvantage of low accuracy and low diversity. These datasets tend to have a weak domain adaptation ability, which is known as the "dataset bias problem".

This research aims at automatically collect accurate and diverse images for given queries from the Web, and construct a domain robust image dataset. Thus, within this thesis, various methods are developed and presented to address the following research challenges. The first is the retrieved web images are usually noisy, how to remove noise and construct a relatively high accuracy dataset. The second is the collected web images are often associated with low diversity, how to address the dataset bias problem and construct a domain robust dataset.

In Chapter 3, a framework is presented to address the problem of polysemy in the process of constructing a high accuracy dataset. Visual polysemy means that a word has several semantic (text) senses that are visually (image) distinct. Solving polysemy can help to choose appropriate visual senses for sense-specific images collection, thereby improving the accuracy of the collected images. Unlike previous methods which leveraged the human-developed knowledge such as Wikipedia or dictionaries to handle polysemy, we propose to automate the process of discovering and distinguishing multiple visual senses from untagged corpora to solve the problem of polysemy.

In Chapter 4, a domain robust framework is presented for image dataset construction. To address the dataset bias problem, our framework mainly consists of three stages. Specifically, we first obtain the candidate query expansions by searching in the Google Books Ngram Corpus. Then, by treating word-word (semantic) and visual-visual distance (visual) as features from two different views, we formulate noisy query expansions pruning as a multi-view learning problem. Finally, by treating each selected query expansion as a "bag" and the images therein as "instances", we formulate image selection and noise removal as a multi-instance learning problem. In this way, images from different distributions can be kept while noise is filtered out.

Chapter 5 details a method for noisy images removing and accurate images selecting. The accuracy of selected images is limited by two issues: the noisy query expansions which are not filtered out and the error index of image search engine. To deal with the noisy query expansions, we divide them into two types and propose to remove noise from visual consistency and relevancy respectively. To handle noise induced by error index, we classify the noisy images into three categories and filter out noise by different mechanisms separately.

Chapter 6 proposes an approach for enhancing classifier learning by using the collected web images. Different from previous works, our approach, while improving the accuracy and robustness of the classifier, greatly reduces the time and labor dependence. Specifically, we proposed a new instance-level MIL model to select a subset of training images from each selected privileged information and simultaneously learn the optimal classifiers based on the selected images.

Chapter 7 concludes the thesis and outlines the scope of future work.

Chapter 1

Introduction

1.1 Background

In the past few years, labeled image datasets have played a critical role in high-level image understanding. For example, ImageNet (Deng, Dong, Socher, Li, Li & Fei-Fei 2009) has acted as one of the most important factors in the recent advance of developing and deploying visual representation learning models (e.g., deep CNN (Krizhevsky, Sutskever & Hinton 2012)). However, as the computer vision community considers more visual categories and greater intra-class variations, it is clear that larger and more exhaustive datasets are needed. Due to the process of constructing such datasets is time-consuming and labor-intensive. It is unlikely that the manual annotation can keep pace with the growing need for annotated datasets.

To reduce the time and labor costs of manual annotation, some works focused on active learning. For example, a method in (Collins, Deng, Li & Fei-Fei 2008) proposed to label some seed images to train the initial classifiers. Then these classifiers were used to do image categorization on other unlabeled images, to find low confidence images for manual labeling. The process is iterated until sufficient classification accuracy is achieved. In (Vijayanarasimhan 2014), a system for online learning of object detectors was proposed. This system refines its models by actively requesting annotations on images. Active learning methods require pre-existing annotations, which is one of the most significant limitations to overcome the scalability.

With the development of the Internet, we have entered the era of big data. It is consequently a natural idea to leverage the large scale yet noisy data on the web for image dataset construction. Methods of exploiting web images for automatic image dataset construction have recently become a hot topic (Hua & Li 2015, Schroff, Criminisi & Zisserman 2011, Yao, Zhang, Shen, Hua, Xu & Tang 2016, Li & Fei-Fei 2010) in the field of multimedia processing. Compared to manually labeled datasets, web images are a richer and larger resource. For arbitrary categories, the possible training data can be easily obtained from an image search engine. Unfortunately, due to the error index of image search engine, retrieved images are limited by the poor precision and restrictions on the total numbers. For example, Schroff *et al.* in (Schroff et al. 2011) reported the average precision of Google Image Search engine on 18 categories is only 32%, and downloads are restricted to 1000 images for each query.

One of the most important reasons for the noisy results is the inherent ambiguity in the user query. In addition, the retrieved images from image search engine usually have the overlapping problem which results in a reduced diversity. In general, there are three main challenges: visual polysemy, limited diversity, and low accuracy.

Some existing unsupervised approaches attempt to reduce the influence of visual polysemy by filtering out irrelevant images (Fergus, Fei-Fei, Perona & Zisserman 2005, Berg & Forsyth 2006, Li & Fei-Fei 2010, Schroff et al. 2011, Hua & Li 2015). For example, one approach in (Li & Fei-Fei 2010) utilized the few top-ranked images returned from an image search engine to learn the initial classifier. The classifier refines its model through incremental learning strategy. With the increase in the number of positive images accepted by the classifier, the learned classifier will reach a robust level. The method in (Hua & Li 2015) leveraged the clustering based strategy to remove "group" noisy images and propagation based strategy to filter individual noisy images.

Since the semantic and visual senses of a given query are highly related, recent works also concentrated on jointly leveraging text and images (Loeff, Alm & Forsyth 2006, Wan, Tan, Lim, Chia & Roy 2009, Saenko & Trevor 2009). Most of these methods assume that there exists a one-to-one mapping between semantic and visual sense towards to the given query. However, this assumption is not always true in practice (Chen, Ritter, Gupta & Mitchell 2015). To deal with the multiple visual senses, Chen *et al.* in (Chen et al. 2015) adopt a one-to-many mapping between semantic and visual spaces. This approach can help us to find multiple visual senses from the web but overly depends on the collected web pages. If we can not collect web pages that contain multiple semantic and visual senses for the given query, the effect of this method will be greatly reduced.

To ensure the diversity of the collected images, methods (Vijayanarasimhan & Grauman 2008, Duan, Li, Tsang & Xu 2011) partitioned candidate images into a set of clusters, treated each cluster as a "bag" and the images therein as "instances", and proposed MIL based methods to prune noisy images. However, the yield for both of two methods mentioned above is limited by the poor diversity of the initial candidate images which were obtained through one single query. To obtain lots of candidate images in a richer diversity, Divvala *et al.* (Divvala, Farhadi & Guestrin 2014) proposed to use multiple query expansions instead of one single query to collect images. However, the yield for (Divvala et al. 2014) is restricted by the iterative mechanism in the process of noises removing and images selection.

To improve the overall accuracy, some authors proposed to re-rank the images returned from the image search engine (Lin, Jin & Hauptmann 2003, Fergus, Perona & Zisserman 2004, Fergus et al. 2005, Vijayanarasimhan & Grauman 2008, Li & Fei-Fei 2010). Fergus *et al.* in (Fergus et al. 2004) and (Fergus et al. 2005) proposed to use visual clustering of the images over a visual vocabulary while method (Vijayanarasimhan & Grauman 2008) adopted multiple instances learning to learn the visual classifiers for images re-ranking. Li *et al.* in (Li & Fei-Fei 2010) leveraged the first few images

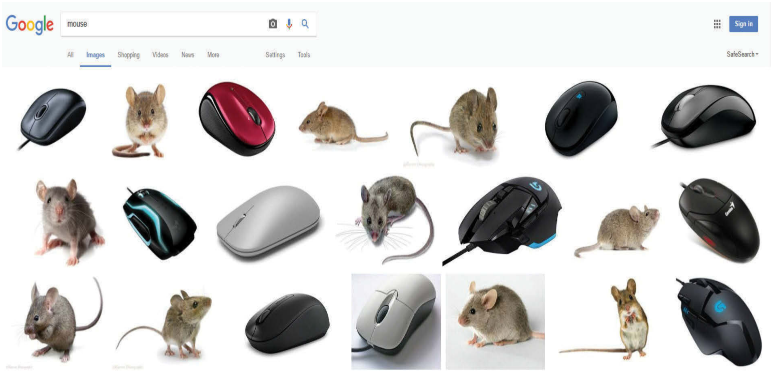

Figure 1.1: Visual polysemy. For example, the query "mouse" returns multiple visual senses on the first page of results. The retrieved web images suffer from the low precision of any particular visual sense.

returned from an image search engine to train the image classifier, classifying images as positive or negative. When the image is classified as a positive sample, the classifier uses incremental learning strategy to refine its model and collect more positive images.

The goal of our thesis is how to quickly build a diverse and accurate dataset. We expect that our method will not only be applicable in the process of large-scale data collection but also hope to apply it in building small-scale datasets. Deep learning methods have been applied to many problems and have achieved good results. But deep models are computationally intensive and require a large number of data. There are some limitations in building small-scale datasets with deep models. However, our proposed approach can efficiently solve this problem.

1.2 Research Issues

Although automatically construct image dataset technology has recently garnered more attention from many scholars, there are still some unsolved and partially solved problems that can be further explored and discussed:

- Visual polysemy. Visual polysemy means that a word has several se-

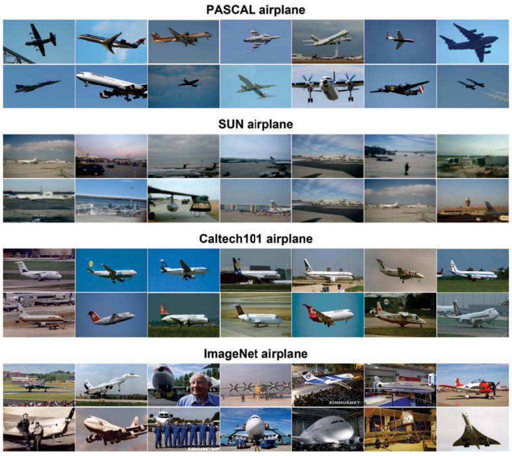

Figure 1.2: Most discriminative images for "airplane" from four different datasets. Each dataset has their preference for image selection.

mantic senses that are visually distinct. One of the most important reasons for the noisy results is the inherent ambiguity in the user query. As shown in Fig. 1.1, when we submit the query "mouse" into the Google Image Search engine, the returned results can refer to the animal "mouse", or the electronic product "mouse". The retrieved web images suffer from the low precision of any particular visual sense. Therefore, handling polysemy is a useful and challenging problem in the process of automatically construct image datasets.

- Diversity. Existing methods usually use an iterative mechanism in the process of image selection. However, due to the visual feature distribution of images selected in this way, these datasets tend to have the dataset bias problem. Fig. 1.2 shows the "airplane" images from four different image datasets. We can observe some significant differences in these datasets: PASCAL shows "airplanes" from the flying viewpoint, while SUN tends to show distant views at the airport; Caltech has a strong preference for side views and ImageNet is rich in diversity, but mainly contains close-range views (Torralba & Efros 2011). Classifiers learned from these datasets usually perform poorly in domain adaptation tasks. To obtain a domain robust image dataset, further exploration of the dataset bias problem is worthwhile.

- Accuracy. Due to the error index of image search engine, even with the top few images, noisy images may still be included. Existing methods tend to solve this problem by re-ranking the returned images from image search engine. However, the performance of these methods is still unsatisfactory. It is also essential to design an algorithm for improving the accuracy of the collected web images.

1.3 Research Contributions

After researching the above issues, the author has developed corresponding solutions, presented in this thesis. These study contributions follow.

- Proposed a novel approach for discovering and distinguishing multiple visual senses for polysemous words without explicit supervision. (chapter 3);
- Released one domain robust image dataset DRID-20 on website. We hope the diversity of DRID-20 can offer unparalleled opportunities to researchers in the multi-instance learning, transfer learning, image dataset construction and other related fields. (chapter 4).
- Proposed a general image dataset construction framework that ensures the scalability and accuracy of the image collections while with no need of manual annotation. (chapter 5);
- Proposed three different filtering mechanisms for different types of noisy images in the process of image dataset construction. (chapter 5);
- Released one dataset WSID-100 on website, we hope the scalability and accuracy of WSID-100 can help researchers further their study in the computer vision and other related fields. (chapter 5);

- Provided a benchmark platform for evaluating the performance of various algorithms in the task of pruning noise and selecting useful data. (chapter 5);

- Proposed an approach for enhancing classifier learning by using the collected web images. (chapter 6);

- Proposed a new instance-level MIL model to select a subset of training images from each selected privileged information and simultaneously learn the optimal classifiers based on the selected images (chapter 6).

1.4 Thesis Structure

The thesis is structured as follow:

Chapter 2 provides a literature review of image dataset construction. Specifically, we first gave the background of image dataset construction, as well as the existing foundation. Then we discuss the existing methods of constructing image datasets from three aspects.

Chapter 3 presents a framework for solving the visual polysemy in the process of image dataset construction. Unlike previous works which leveraged the human-developed knowledge to handle polysemy, we propose to automate the process and leverage untagged corpora to solve the problem of polysemy. Specifically, we first discover a list of possible semantic senses to retrieve sense-specific images. Then we merge visual similar semantic senses and prune noise by using the retrieved images. Finally, we train one visual classifier for each selected semantic sense and use the learned sense-specific classifiers to distinguish multiple visual senses. Relevant experiments are designed to verify the effectiveness and accuracy of the method.

Chapter 4 proposes a domain-robust image dataset construction framework that can be generalized well to unseen target domains. Specifically, the given queries are first expanded by searching in the Google Books Ngrams Corpus to obtain a rich semantic description, from which the visually non-salient and less relevant expansions are filtered out. By treating each selected expansion as a bag and the retrieved images therein as instances, we formulate image selection as a multi-instance learning (MIL) problem with constrained positive bags. We propose to solve the employed problems by the cutting-plane and concave-convex procedure (CCCP) algorithm. To verify the effectiveness of our proposed approach, we build an image dataset with 20 categories. Extensive experiments on image classification, cross-dataset generalization, diversity comparison and object detection demonstrate the domain robustness of our dataset.

Chapter 5 presents a novel image dataset construction framework which aims at collecting accurate images for given queries from the Web. Specifically, we formulate noisy textual metadata removing and noisy images filtering as a multi-view and multi-instance learning problem separately. To verify the effectiveness of our proposed approach, we construct an image dataset with 100 categories. The experiments show significant performance gains by using the generated data of our approach on several tasks, such as image classification, cross-dataset generalization and object detection.

Chapter 6 presents a new approach for enhancing classifier learning by using the collected web images. Specifically, we proposed a new instance-level MIL model to select a subset of training images from each selected privileged information and simultaneously learn the optimal classifiers based on the selected images. Extensive experimental results demonstrated the superiority of our proposed approach.

Chapter 7 concludes the thesis and outlines the scope of future work.

Figure 1.3 shows the research profile of this thesis.

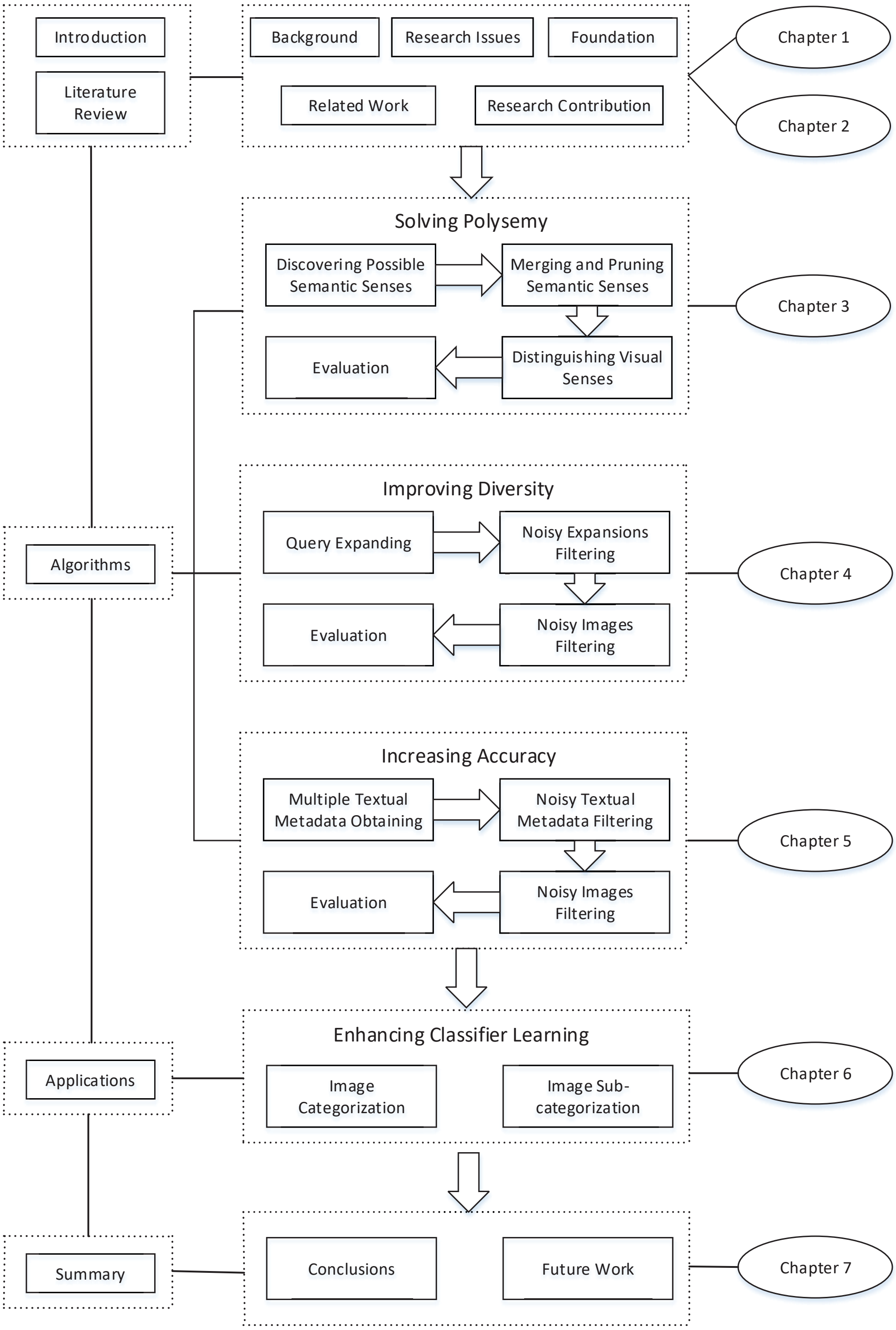

Figure 1.3: The profile of work in this thesis.

Chapter 2

Literature Review and Foundation

Automatic image dataset construction is a hot research field in computer vision, multimedia processing, and other fields. Image dataset construction has also attracted much research attention in many institutions. It is a popular topic in important academic journals and conferences, such as the IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, IEEE Transactions on Knowledge and Data Engineering, ACM Conference on Multimedia. IEEE Conference on Multimedia and Expo, AAAI Conference on Artificial Intelligence, International Joint Conference on Artificial Intelligence, and the IEEE Conference on Computer Vision and Pattern Recognition.

Although much progress has been made in automating the construction of image datasets, the collected data by these methods still have some drawbacks. The first is that the collected data typically contain large amounts of polysemy noise. This is mainly because of the inherent ambiguity in the user query. The second is that the collected data are of limited diversity. The reason is that existing methods usually leverage a single query to collect images and use an iterative mechanism for filtering noisy images. The third is that the accuracy of the collected data has yet to be improved.

Given the importance of labelled image datasets in the area of high-level image understanding, many efforts have been directed toward image dataset construction. In general, these efforts can be divided into three principal categories: polysemy-based methods, diversity-based methods, and accuracy-based methods.

2.1 Polysemy-oriented Methods

Visual polysemy means that a word has several semantic senses that are visually distinct. Automatically discovering and distinguishing multiple visual senses for polysemous words is an extremely difficult problem.

Several authors proposed to clean the retrieved images and learn visual classifiers, although none have specifically addressed the problem of polysemy (Fergus et al. 2004, Fergus et al. 2005, Berg & Forsyth 2006, Li & Fei-Fei 2010, Schroff et al. 2011, Hua & Li 2015). Fergus *et al.* in (Fergus et al. 2004) proposed the use of visual classifiers learned from Google Image Search engine to re-rank the images based on the visual consistency. Subsequent methods (Fergus et al. 2005, Li & Fei-Fei 2010) have employed similar removing mechanisms to automatically construct clean image datasets for training visual classifiers. Berg *et al.* in (Berg & Forsyth 2006) discovered topics using LDA in the text domain, and then use them to cluster the images. This approach requires manual intervention by the user to sort the topics into positive and negative for each category. Schroff *et al.* in (Schroff et al. 2011) adopted text information to rank images retrieved from a web search and used these top-ranked images to learn visual models to re-rank images once again. The method in (Hua & Li 2015) leveraged the clustering based strategy to remove “group” noisy images and propagation based strategy to filter individual noisy images. These methods have the advantage of eliminating manual intervention. However, these methods are category-independent and do not learn which words are predictive of a specific sense.

The traditional way to handle polysemy is text-based methods (Pantel &

Lin 2002, Chatterjee & Mohan 2008, Snow, Prakash, Jurafsky & Ng 2007). Pantel *et al.* in (Pantel & Lin 2002) presented a clustering algorithm called Clustering By Committee (CBC) that automatically discovers word senses from text. It firstly discovers a set of tight clusters called committees that are well scattered in the similarity space. Then proceed by assigning words to their most similar clusters. It allows CBC to discover the less frequent senses of a word and to avoid discovering duplicate senses. Each cluster that a word belongs to represents one of its senses. Two subsequent methods in (Chatterjee & Mohan 2008, Snow et al. 2007) have also employed similar Clustering by Committee algorithm to congregate similar words.

Some works also leveraged the human-developed knowledge to handle polysemy (Veronis & Ide 1990, Yarowsky 1995, Yarowsky 1992, Mihalcea 2007). Yarowsky in (Yarowsky 1992) proposed to disambiguate word senses in unrestricted corpora using statistical models of the major Roget's Thesaurus categories. Roget's categories serve as approximations of conceptual classes. The categories listed for a word in Roget's index tend to correspond to sense distinctions; thus selecting the most likely category provides a useful level of sense disambiguation. The selection of categories is accomplished by identifying and weighing words that are indicative of each category when seen in context, using a Bayesian theoretical framework. Then Yarowsky in (Yarowsky 1995) proposed an unsupervised word senses disambiguation method but relied on the use of dictionary definition as an initial seed. Mihalcea *et al.* in (Mihalcea 2007) and Veronis *et al.* in (Veronis & Ide 1990) proposed to use Wikipedia and dictionary for disambiguating word senses.

Since the semantic and visual senses of a given query are highly related, recent works also concentrated on jointly leveraging text and images (Loeff et al. 2006, Wan et al. 2009, Saenko & Trevor 2009, Chen et al. 2015). The method in (Loeff et al. 2006) involves two major steps: (1) extracting and weighting text features from the web pages, visual features from the retrieved images, (2) running spectral clustering on both of the text features and visual features to derive the multiple semantic senses. Wan *et al.* in (Wan et al.

2009) and Saenko *et al.* in (Saenko & Trevor 2009) proposed a latent model to learn multiple visual senses from a large collection of unlabeled web data, but rely on Wikipedia and WordNet's sense inventory respectively. Chen *et al.* in (Chen et al. 2015) proposed a one-to-many mapping between the text-based feature space and image-based visual space to discover multiple semantic and visual senses of a Noun Phrase. However, clustering presents a scalability issue for this problem. The reason is that our images are sourced directly from the web and have no bounding boxes, every image creates millions of data points, the majority of which are outliers. In addition, this approach overly depends on the quality of the collected web pages, and the effect will be greatly reduced when we can not collect web pages that contain enough useful semantic and visual senses.

2.2 Diversity-oriented Methods

Most of the existing methods (Schroff et al. 2011, Li & Fei-Fei 2010, Hua & Li 2015) leverage one single query to collect images. However, due to the limitation of one single query, the diversity of the collected images has been greatly reduced. WordNet (Miller 1995) and ConceptNet (Speer & Havasi 2013) are often used to obtain synonyms to improve the diversity as well as to overcome the download restriction of image search engines. The advantage of WordNet (Miller 1995) and ConceptNet (Speer & Havasi 2013) is that synonyms are usually relevant to the given query and almost do not need to be purified. The disadvantage of WordNet (Miller 1995) and ConceptNet (Speer & Havasi 2013) is that both of them are usually not comprehensive enough for query expanding. Worse, the images returned from image search engine using synonyms tend to experience the homogeneous problem, which results in poor performance on dataset diversity.

Recent works (Yao, Zhang, Shen, Hua, Xu & Tang 2016) and (Divvala et al. 2014) proposed the use of Google Books Ngram Corpus (GBNC) (Lin, Michel & Petrov 2012) instead of WordNet and ConceptNet to obtain query expansions for candidate images collection, then using an iterative mechanism to filter noisy images. The Google Books Ngrams Corpus covers almost all related queries at the text level. It is much more general and richer than WordNet and ConceptNet. The disadvantage of using GBNC for query expanding is that it may also bring noisy query expansions. Recently, word embedding (Cilibrasi & Vitanyi 2007) provides a learning-based method for computing the word-word similarity distance which can be used to filter noisy query expansions.

Due to the iterative mechanism used in the process of images selection, the diversity of the collected images is still of limited. To efficiently ease the dataset diversity problem, several authors have developed domain-robust approaches (MIL) for various vision tasks. MIL is in the sense that we partition the training samples into clusters and use the bag to denote each cluster. A set of MIL approaches were developed in (Li, Duan, Xu & Tsang 2011, Andrews, Tsochantaridis & Hofmann 2003, Li, Kwok, Tsang & Zhou 2009). In multi-instance (mi-SVM) (Andrews et al. 2003), the support vector machine (SVM) classifier is trained at each iteration based on the inferred instance labels from the previous iteration. In key-instance (KI-SVM) (Li, Kwok, Tsang & Zhou 2009), the key instances inside each bag are used as the representatives of the bag. Nevertheless, these methods were proposed without taking the data distribution mismatch between two domains into consideration, so that the learned classifiers may not generalize well to the arbitrary target domain.

Duan *et al.* in (Duan et al. 2011) clustered relevant images using both textual and visual features. By treating each cluster as a "bag" and the images in the bag as "instances", the authors formulated this problem as a multi-instance learning problem (MIL) which learns a target decision function for image re-ranking. Xu *et al.* in (Xu, Li, Niu & Xu 2014) exploited the low-rank structure of source latent domains based on exemplar classifiers. When we have target domain data in the training process, domain adaptation approaches can be used to reduce the domain distribution mismatch.

The recently developed domain adaptation approaches can be classified into classifier-based methods (Duan, Xu & Tsang 2012, Duan et al. 2012, Bruzzone & Marconcini 2010), instance-reweighting methods (Huang, Gretton, Borgwardt, Schölkopf & Smola 2007), and feature-based methods (Gopalan, Li & Chellappa 2011, Kulis, Saenko & Darrell 2011, Gong, Shi, Sha & Grauman 2012, Baktashmotlagh, Harandi, Lovell & Salzmann 2013, Fernando, Habrard, Sebban & Tuytelaars 2013). Some works (Ding, Shao & Fu 2014, Jhuo, Liu, Lee & Chang 2012, Shao, Kit & Fu 2014, Ding, Shao & Fu 2015) applied low-rank techniques for domain adaptation. In particular, the transformed source domain samples are expected to be linearly constructed by the target domain samples in (Jhuo et al. 2012). In (Shao et al. 2014), both the source and target domain data are projected to the common subspace, where each target domain sample can be linearly constructed by the source domain samples. Ding *et al.* in (Ding et al. 2015) proposed an iterative approach, in which the transformed source domain is treated as the dictionary to reconstruct the transformed data from both domains at each iteration. Ding *et al.* in (Ding et al. 2014) proposed to recover the missing modality in the target domain under a transfer learning framework.

2.3 Accuracy-oriented Methods

According to the process of image collection, the accuracy-based methods can be divided into three types: manual annotation, active learning methods, and automatic methods.

2.3.1 Manual Annotation Methods

In the early years, manual annotation was the most important way to construct diverse image datasets (e.g., STL-10 (Coates, Ng & Lee 2011), CIFAR-10 (Krizhevsky & Hinton 2009), PASCAL VOC (Everingham, Van Gool, Williams, Winn & Zisserman 2010), ImageNet (Deng et al. 2009), LabelMe (Russell, Torralba, Murphy & Freeman 2008), SUN (Xiao, Hays, Ehinger, Oliva & Torralba 2010), and Caltech-101 (Griffin, Holub & Perona 2007)). Most of these datasets are built by sending category names to image search engines and aggregating returned images as candidate images, then cleaning candidate images by human judgement. Here, we briefly discuss these works along the steps involved in image dataset construction:

Generating Category List. The generation of category list depends on specific tasks. For example, SUN (Xiao et al. 2010) targets on scene recognition task by defining 899 scene categories. Borth *et al.* in (Borth, Ji, Chen, Breuel & Chang 2013) proposed to detect visual sentiment by constructing a dataset around a category list with strong sentiment. Datasets such as Tiny-Image (Torralba, Fergus & Freeman 2008) and ImageNet (Deng et al. 2009) directly adopt nouns of WordNet (Miller 1995) as category list, which cover a large amount of objects but are still far from complete.

Query formation. Since most image search engines restrict the number of images returned for each query (in the order of hundreds to one thousand) and only top-ranked images are with acceptable precision. To overcome the restriction, synonyms are often used to expand a category into a query set. Moreover, methods such as appending category with popular adjectives and words from its parent category, even translating category to different languages are further used to enrich the query set. All expanded queries will submit to several popular image search engines to collect candidate images from the Internet. The method only works for categories defined from existing ontology such as WordNet (Miller 1995), and cannot generalize to categories that have not been compiled into existing ontology. Recently, word embedding (Collobert & Weston 2008, Pennington, Socher & Manning 2014) provides a learning-based method to compute the similarity between words and can be used to bypass the manual compilation of ontology.

Noisy image removal. The candidate images contain lots noisy images with average accuracy around 10% (Deng et al. 2009). Human efforts are involved to remove noisy images by checking candidate images one by one. As this step is quite time-consuming and labor-intensive, NUS-WIDE only partially labelled the whole dataset (Chua, Tang, Hong, Li, Luo & Zheng 2009), while TinyImage (Torralba et al. 2008) and visual sentiment dataset (Borth et al. 2013) keep all raw candidate images without manual labelling. Manual labeling has high accuracy but is limited in scalability and diversity.

2.3.2 Active Learning Methods

To reduce the cost of manual annotation, a large number of works have focused on active learning (a special case of the semi-supervised method). Li *et al.* in (Collins et al. 2008) randomly labelled some seed images to learn visual classifiers. The learned visual classifiers were then implemented to conduct image classification on unlabelled images, to find low confidence images for manual labelling. Here low confidence images are those whose probability is classified into positive and negative close to 0.5. The process is iterated until sufficient classification accuracy is achieved. Siddiquie *et al.* in (Siddiquie & Gupta 2010) presented an active learning framework to simultaneously learn contextual models for scene understanding tasks (multi-class classification). Grauman *et al.* in (Vijayanarasimhan 2014) presented an approach for online learning of object detectors, in which the system automatically refines its models by actively requesting crowd-sourced annotations on images crawled from the web. However, active learning requires pre-existing annotations, which often results in one of the most significant limitations to construct a large-scale image dataset.

2.3.3 Automatic Methods

To further reduce the cost of manual annotation, automatic methods have attracted more and more people's attention. Schroff *et al.* in (Schroff et al. 2011) adopted text information to rank images retrieved from a web search and used these top-ranked images to learn visual models to re-rank images once again. Li *et al.* in (Li & Fei-Fei 2010) leveraged the first few images returned from an image search engine to train the image classifier, classifying

Table 2.1: The publicly available automatic datasets.

Ours	WSID-100	DRID-20	AutoImgSet-10
Others	Webvision	MIT-ISD	CMU-Poly-30

images as positive or negative. When the image is classified as a positive sample, the classifier uses incremental learning to refine its model. With the increase in the number of positive images accepted by the classifier, the trained classifier will reach a robust level for this query. Hua *et al.* in (Hua & Li 2015) proposed to use a clustering-based method to filter "group" noisy images and propagation based method to filter individual noisy images. We summaries the publicly available automatic datasets in Table 2.1:

2.4 Privileged Information

Data-driven approaches become very brittle and prone to over-fitting when the training data is inadequate either in quantity or quality. Unfortunately, this is often the case in many real-world applications. A natural solution to alleviate this limitation is incorporating additional privileged information (Wang & Ji 2015, Li, Niu & Xu 2014, Niu, Li, Xu & Cai 2017, Divvala et al. 2014). For example, in object recognition, in addition to the image features and labels (e.g., , "horse"), the learner may also leverage object attributes (e.g., , "walking" and "jumping") in the training process. In human action recognition, besides the RGB features and human action labels, human joint positions can be incorporated into the classifier training. In practice, the privileged information can be tags, properties, attributes, positions or the context of the web images.

However, learning classifier with privileged information is a challenging problem. The difficulty lies in three aspects. Firstly, the process of manually labeling privileged information is very expensive. Secondly, it is only available during training and unseen during testing. We cannot combine the privileged information with input features to predict the category label. Thirdly, learning classifiers with PI overly depends on the quality of the collected PI.

Li *et al.* in (Li et al. 2014) proposed an image categorization method by incorporating the textual features (extracted from the surrounding textual descriptions) and simultaneously coping with noise in the loose labels of training images. Similarly, method (Wang & Ji 2015) and (Niu et al. 2017) adopt different types of PI to improve the classifier learning. All of the methods in (Li et al. 2014, Wang & Ji 2015) and (Niu et al. 2017) encode privileged information into the parameters of the classifier during training. The disadvantage is that these methods overly depend on the quality of the collected privileged information. Due to the complexity of the Internet, it is difficult to select useful privileged information from the surrounding textual descriptions which contain a large amount of noise. The performance of the learned classifier will be largely reduced when we failed to filter out the noisy privileged information during training.

Chapter 3

Polysemy

3.1 Introduction

Web images are a rich and free resource. For arbitrary categories, the potential training data can be easily obtained from the image search engines like Google or Bing. Unfortunately, due to the error index of image search engine, the precision of returned images from image search engine is still unsatisfactory. For example, Schroff *et al.* in (Schroff et al. 2011) reported that the average precision of the top 1000 images for 18 categories from Google Image Search engine is only 32%. One of the most important reasons for the noisy results is the inherent ambiguity in the user query.

Visual polysemy means that a word has several semantic senses that are visually distinct. Some existing unsupervised approaches attempt to reduce the influence of visual polysemy by filtering out irrelevant images. For example, one approach in (Li & Fei-Fei 2010) utilized the few top-ranked images returned from an image search engine to learn the initial classifier. The classifier refines its model through incremental learning strategy. With the increase in the number of positive images accepted by the classifier, the learned classifier will reach a robust level. The method in (Hua & Li 2015) leveraged the clustering based strategy to remove "group" noisy images and propagation based strategy to filter individual noisy images. These methods have the advantage of eliminating manual intervention. However, all of these methods do not directly address the problem of polysemy.

The traditional way to handle polysemy is text-based methods (Pantel & Lin 2002, Chatterjee & Mohan 2008). However, all of these methods have no information about the visual senses and still need manual annotation to bridge the semantic and visual senses. Some works also leveraged the human-developed knowledge such as Wikipedia (Mihalcea 2007) or dictionaries (Veronis & Ide 1990, Yarowsky 1995) to handle polysemy. However, this human-developed knowledge still suffers from the problem of missing information. For example, the machine-readable dictionary has a large coverage of NOUN category, but it contains very few entities (e.g., organizations, locations). Wikipedia can help to bridge this gap, but a great deal of information is still missing (Chen et al. 2015).

Since the semantic and visual senses of a given query are highly related, recent works also concentrated on jointly leveraging text and images (Loeff et al. 2006, Wan et al. 2009, Saenko & Trevor 2009). Most of these methods assume that there exists a one-to-one mapping between semantic and visual sense towards to the given query. This assumption is not always true in practice. To deal with the multiple visual senses, method (Chen et al. 2015) adopt a one-to-many mapping between semantic and visual spaces. This approach can help us to find multiple visual senses from the web but overly depends on the collected web pages (Torralba & Efros 2011). If we can not collect web pages that contain multiple semantic and visual senses for the given query, the effect of this method will be greatly reduced (Yao, Zhang, Shen, Hua, Xu & Tang 2017).

Inspired by the situation described above, we seek to automate the process of discovering and distinguishing multiple visual senses for polysemous words. We propose an unsupervised method that resolves visual polysemy by allowing sense-specific diversity in search results. We take a three-step approach. Firstly, we discover a list of possible semantic senses to retrieve sense-specific images. Secondly, we merge visual similar semantic senses and

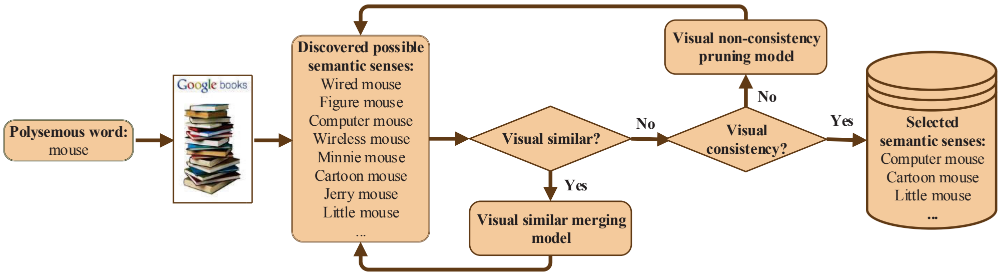

Figure 3.1: Illustration of the process for obtaining selected semantic senses.

prune noise by using the retrieved sense-specific images. Thirdly, we learn one visual classifier for each selected semantic sense and use the learned sense-specific classifiers to group and re-rank the polysemous images into its specific senses. To verify the effectiveness of our approach, we conducted experiments on the tasks of classifying images into sense-specific categories and re-ranking search results. The experimental results demonstrate the superiority of our proposed approach.

3.2 Framework and Methods

The inspiration for our work stems from the fact that web images indexed by a polysemous word are often rich in diversity. Our main idea of solving the problem of polysemy is allowing sense specific diversity in search results. Specifically, our proposed framework consists of three major steps: 1) discovering a list of possible semantic senses, to retrieve sense-specific images, 2) merging and pruning semantic senses, 3) distinguishing multiple visual senses for polysemous words.

3.2.1 Discovering Possible Semantic Senses

Inventories of manually compiled dictionaries (e.g., WordNet (Miller 1995), ConceptNet (Speer & Havasi 2013)) usually serve as a source for word senses. However, they often include many rare senses while missing corpus/domain-specific senses. In addition, the process of constructing manually compiled dictionaries is time-consuming and labor-intensive. To ease the limitations of missing information, as well as to reduce the dependence on manually labeled data, method (Pantel & Lin 2002) and (Chatterjee & Mohan 2008) proposed to discover semantic senses from text via clustering. The disadvantage is that these methods overly depend on the quality of the collected text. The performance of these methods will be greatly reduced when we failed to collect enough useful text.

Inspired by recent works (Divvala et al. 2014, Michel, Shen et al. 2011), we can use untagged Google Books Ngram Corpus to discover an exhaustive vocabulary explaining all the appearance variations for the given query. Compared to manually labeled WordNet (Miller 1995) and ConceptNet (Speer & Havasi 2013), it is not only much richer but also more general and exhaustive. Following (Lin et al. 2012) (see section 4.3), we specifically use the dependency gram data with parts-of-speech (POS) for possible semantic senses discovering. For example, given a word (e.g., "mouse") and its corresponding POS tag (e.g., 'mighty, ADJ'), we find all its occurrences annotated with POS tag within the dependency gram data. Of all the ngram dependencies retrieved for the given word, we choose those whose modifiers are tagged as NOUN, VERB, ADJECTIVE, and ADVERB as the possible semantic senses. Our motivation is to find all the possible semantic senses the human race has ever written down in books. We use these discovered semantic senses to retrieve sense-specific web images from the image search engine.

3.2.2 Merging and Pruning Semantic Senses

Among the list of possible semantic senses, some of them are sharing visually similar distributions (e.g., "jerry mouse", "Minnie mouse" and "cartoon mouse"). To avoid training separate models for visually similar semantic senses, and to pool valuable training data across them, we need to merge and sample these visually similar semantic senses. In addition, not all the discovered semantic senses are useful, some noise may also be included (e.g., “figure mouse” and “flying mouse”). To avoid training meaningless visual models and to better distinguish multiple visual senses, we need to prune these noisy semantic senses.

Merging visual similar semantic senses

The traditional way to merge senses is calculating the semantic similarity of texts (Snow et al. 2007, Cilibrasi & Vitanyi 2007). These methods usually calculate the semantic similarity by calculating the frequency of their simultaneous appearance. Semantically similar senses usually have a smaller semantic distance. However, this assumption is not always true from the perspective of computer vision. For example, the semantic distance (Normalized Google Distance (Cilibrasi & Vitanyi 2007)) between “hot dog” and “dog” is relatively smaller (0.213). But visually speaking, they are two completely different objects that should not be merged. To this end, different from previous works which merge semantic senses from the textual semantics view, but from a visual point of view.

For each possible semantic sense, we use the top N images from image search engine to represent its visual distribution. We denote the visual similarity space of all discovered semantic senses by a graph $G = \{V, W\}$, where each node represents a semantic sense and each edge represents the visual similarity between two nodes. Each node has a score S_i which corresponds to the quality of its classifier. Specifically, we assume the top N images are positive instances, then these images were randomly split into a training set and validation set $I_i = \{I_i^t, I_i^v\}$. A random pool of negative images was collected and split into a training set and validation set $\overline{I} = \{\overline{I}^t, \overline{I}^v\}$. We learn the linear SVM classifier f_i with I_i^t and $\overline{I}^t$ using the 4096 dimensional deep features (based on AlexNet (Krizhevsky et al. 2012)). We then use $\{I_i^v, \overline{I}^v\}$ as validation images to calculate the classification results. We set the score S_i equal to the classification results on its own validation set $\{I_i^v, \overline{I}^v\}$. The edge weights $W_{i,j}$ correspond to the visual similarity between two nodes, and is measured by the score of the ith node classifier f_i on the jth node validation set $\{I_j^v, \overline{I}^v\}$.

Then the problem of merging visually similar semantic senses can be formulated as sampling a representative subset of space $v \subseteq V$ which maximizes the quality of the subset:

$$\begin{aligned} \max_v \quad & \sum_{i \in V} S_i \cdot \phi(i, v) \\ \text{s.t.} \quad & |v| \leqslant k \end{aligned} \tag{3.1}$$

where k is the number of semantic senses for the given word and ϕ is a soft coverage function that implicitly ensure the diversity of representative subset:

$$\phi(i, v) = \begin{cases} 1 & i \in v \\ 1 - \prod_{j \in v}(1 - W_{i,j}) & i \notin v \end{cases} \tag{3.2}$$

Similar to recent work (Batra, Yadollahpour, Guzman-Rivera & Shakhnarovich 2012), our formulation is to find a subset of representative space v which can cover the space of variance within the space V. Since our objective function is sub-modular, we can get a constant approximation of the optimal solution. We use an iterative mechanism for discovering the most representative subset. Particularly, we add one semantic sense i at each iteration by maximizing the current space:

$$\arg\max_i S(v \cup i) - S(v). \tag{3.3}$$

By setting the cost of adding semantic sense in v to a large value, each new semantic sense can be merged to its closest member in v.

Pruning noisy semantic senses

After we merge the visual similar semantic senses, we get a relatively few discrete senses. Among these discrete senses, some noise may also be included. To avoid training meaningless visual models and to better distinguish multiple visual senses, we prune these noisy semantic senses. As shown in Fig 3.2, our basic idea is that noisy semantic senses have no specific visual patterns

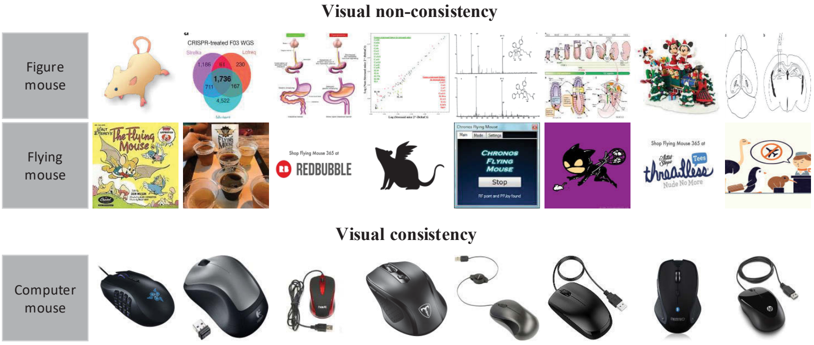

Figure 3.2: A snapshot of the retrieved images for visual consistency and non-consistency semantic senses.

(e.g., "figure mouse", "flying mouse"). Thus, we can prune noise from the perspective of visual consistency.

We represent each discrete semantic sense as a "bag" and the retrieved images therein as "instances". In particular, we represent each semantic sense G_I with the compound feature $\delta_{f,k}$ of its top k positive images:

$$\delta_{f,k}(G_I) = \frac{1}{k} \sum_{x_i \in \Phi^*_{f,k}(G_I)} x_i \tag{3.4}$$

with

$$\Phi^*_{f,k}(G_I) =^{\arg\max}_{\Phi \subseteq G_I, |\Phi|=k} \sum_{x_i \in \Phi} f(x_i). \tag{3.5}$$

The images in $\Phi^*_{f,k}(G_I)$ are referred to the top k positive instances of G_I according to the SVM classifier f_i (obtained in previous step). The closer of images to the center of the bag, the higher probability to be associated with the bag. The assignment of relatively heavier weights to these images would increase the accuracy of classifying semantic sense G_I to be positive or negative, then increase the efficiency of pruning noisy semantic senses. Following (Carneiro, Chan, Moreno & Vasconcelos 2007), the form of weighting function is assumed as

$$\rho_i = [1 + \exp(\alpha \log d(x_i) + \beta)]^{-1}. \tag{3.6}$$

$d(x_i)$ is the visual distance of image x_i to the bag center, $\alpha \in \mathbb{R}_{++}$ and β are scaling and offset parameters. Then the representation of (3.4) for semantic sense G_I can be represented as a weighted compound feature:

$$\delta_{f,k}(G_I) = \delta(X, h^*) = \frac{Xh^*}{\rho^\top h^*} \tag{3.7}$$

with

$$\begin{aligned} h^* = & \underset{h \in \mathrm{H}}{\arg\max} \; f(\frac{Xh}{\rho^\top h}) \\ \text{s.t.} & \sum_i h_i = k. \end{aligned} \tag{3.8}$$

$X = [x_1, x_2, x_3.., x_i] \in \mathbb{R}^{D \times i}$ is a matrix whose columns are the instances of bag G_I, h is a vector of latent variables and H is the hypothesis space $\{0,1\}^i \setminus \{0\}$. $h^* \in \mathrm{H} = \{0,1\}^i \setminus \{0\}$ ($\sum_i h_i = k$) is an indicator function for the top k positive instances of bag G_I. $\rho = [\rho_1, \rho_2, \rho_3 ... \rho_i]^\top \in \mathbb{R}^i_{++}$ are the vectors of weights. Then the decision rule of semantic sense G_I to be selected or pruned is:

$$\begin{aligned} f_{\mathrm{w}}(X) = & \max_{h \in \mathrm{H}} \mathrm{w}^\top \delta(X, h) \\ & \sum_i h_i = k \end{aligned} \tag{3.9}$$

where $\mathrm{w} \in \mathbb{R}^D$ is the vector of classifying coefficients, $\delta(X, h) \in \mathbb{R}^D$ is the feature vector of (3.7). In order to solve the classifying rule of (3.9), we need to solve the below following problem:

$$\begin{aligned} & \max_{h \in \mathrm{H}} \frac{\mathrm{w}^\top X h}{\rho^T h} \\ \text{s.t.} & \sum_i h_i = k. \end{aligned} \tag{3.10}$$

This is an integer linear-fractional programming problem. Since $\rho \in \mathbb{R}^i_{++}$, (3.10) is identical to the relaxed problem:

$$\begin{aligned} & \max_{h \in \lambda^i} \frac{\mathrm{w}^\top X h}{\rho^\top h} \\ \text{s.t.} & \sum_i h_i = k. \end{aligned} \tag{3.11}$$

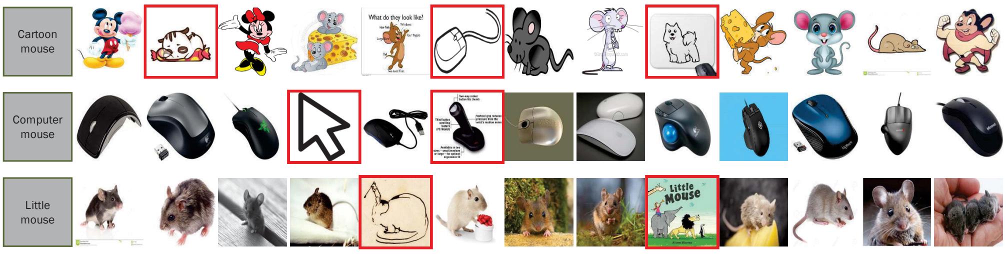

Figure 3.3: A snapshot of the retrieved images for selected semantic senses. Due to the error index of image search engine, even we retrieve the sense-specific images, some instance-level noise may also be included. The noisy images are marked with red bounding boxes.

where $\lambda^i = [0,1]^i$ is a unit box in $\mathbb{R}^i$. (3.11) is a linear-fractional programming problem. We can reduce it to be a linear programming problem with $i+1$ variables and $i+2$ constraints (Boyd & Vandenberghe 2004). Given a training set $\{G_I, Y_I\}_{I=1}^N$, the learning problem is to determine the parameter vector w in (3.9). This is a latent SVM problem:

$$\min_{\mathrm{w}} \frac{1}{2}\left\|\mathrm{w}\right\|^2 + C\sum_{I=1}^{N} \max\left(0, 1 - Y_I f_{\mathrm{w}}\left(X_{G_I}\right)\right). \tag{3.12}$$

The objective of (3.12) can be rewrited as two convex functions:

$$\begin{aligned} \min_{\mathrm{w}} & \left[\frac{1}{2}\left\|\mathrm{w}\right\|^2 + C\sum_{I\in D_N} \max\left(0, 1 + f_{\mathrm{w}}\left(X_{G_I}\right)\right) + \right. \\ & \left. C\sum_{I\in D_P} \max\left(f_{\mathrm{w}}\left(X_{G_I}\right), 1\right)\right] - \left[C\sum_{I\in D_P} f_{\mathrm{w}}\left(X_{G_I}\right)\right] \end{aligned} \tag{3.13}$$

where D_P and D_N are positive and negative training sets respectively. Here we leverage the concave-convex procedure (CCCP) algorithm (Yuille & Rangarajan 2003) to address (3.13). Finally, we obtain the pruning rule as (3.9) to remove noisy semantic senses which have no specific visual senses.

3.2.3 Distinguishing Visual Senses

After pruning the noisy semantic senses, we set the rest as the final selected semantic senses. As shown in Fig 3.3, due to the error index of image search engine, even we retrieve the sense-specific images, some instance-level noise may also be included. The last step of our approach is to prune these instance-level noisy images and train visual classifiers for distinguishing multiple visual senses. Particularly, we train one optimal classifier for each semantic sense based on the selected images.

By treating each selected semantic sense as a "bag" and the retrieved images therein as "instances", we formulate noisy images pruning and classifiers learning as an instance-level multi-instance learning problem. Our objective is to select a subset of images from each bag to learn the optimal classifier for the selected semantic sense. As the accuracy of images retrieved from an image search engine is relatively high, we define each positive bag has a portion of δ positive instances.

Each instance was denoted as x_i with its label $y_i \in \{\pm 1\}$, where $i = 1,...,n$. The label of each bag was denoted as $Y_I \in \{\pm 1\}$. The decision function is assumed in the form of $f(x) = \mathrm{w}^\top \varphi(x) + b$ and it will be used to prune instance-level noisy images. We apply the formulation of Lagrangian SVM. Then the decision function can be learned by minimizing the following structural risk functional:

$$\begin{aligned}
\min_{\mathbf{y},\mathrm{w},b,\rho,\varepsilon_i} \frac{1}{2} & \left(\|\mathrm{w}\|^2 + b^2 + C \sum_{i=1}^{n} \varepsilon_i^2 \right) - \rho \\
\text{s.t. } y_i(\mathrm{w}^\top \varphi\left(x_i\right) + b) & \geq \rho - \varepsilon_i, i = 1, ...n, \\
y_i & = -1 \qquad for \quad Y_I = -1, \\
\sum_{i:x_i \in G_I} \frac{y_i + 1}{2} & \geq \delta \left| G_I \right| \quad for \quad Y_I = 1,
\end{aligned} \tag{3.14}$$

where φ is a mapping function that maps x from the original space into a high dimensional space $\varphi(x)$, $C > 0$ is a regularization parameter and ε_i values are slack variables. The margin separation is defined as $\rho / \|\mathrm{w}\|$.

$\mathbf{y} = [y_1 ... y_n]^\top$ means the vector of instance labels, $\lambda = \{\mathbf{y} | y_i \in \{\pm 1\}\}$ and $\mathbf{y}$ satisfies constraint in (3.14).

We employ the cutting-plane algorithm (Kelley 1960) to solve the optimization problem (3.14). Finally, we can derive the decision function for the selected semantic sense as:

$$f(x) = \sum_{i:\alpha_i \neq 0} \alpha_i \widetilde{y}_i \widetilde{k}(x, x_i) \tag{3.15}$$

where $\widetilde{y}_i = \sum_{t:\mathbf{y}^t \in \lambda} u_t y_i^t$ and $\widetilde{k}(x, x_i) = k(x, x_i) + 1$. The decision function will be used to prune instance-level noisy images in each selected semantic sense. In addition, it will also be leveraged to distinguish different visual senses.

3.3 Experiments

To verify the effectiveness of our proposed approach, in this section, we first conduct experiments on the task of classifying images into sense-specific categories. Then we compare the search results re-ranking ability of our approach with baseline methods.

3.3.1 Classifying Sense-specific Images

Experimental setting

We follow the setting of baseline methods (Loeff et al. 2006, Wan et al. 2009) and exploit web images as the training set, human-labeled images as the testing set. Instead of using co-clustering on web text and images, we use general corpus information and web images to discover and distinguish multiple visual senses for polysemous words. Particularly, we evaluate the performance on following datasets:

• CMU-Poly-30 (Chen et al. 2015). The CMU-Poly-30 dataset consists of 30 polysemy categories. Each category contains a varying number of images.

• MIT-ISD (Saenko & Trevor 2009). The MIT-ISD dataset contains 5 categories. Each of which has three sizes. We are concerned with the "keyword" based size as it has the ground truth.

For each category, we first discover the possible semantic senses by searching in the Google Books Ngram Corpus. Then we retrieve the top $N = 100$ images from the Google Image Search engine for each discovered semantic sense. We assume the retrieved images as the positive instances (in spite of the fact that noisy images might be included). We randomly split the retrieved 100 images for each semantic sense into a training set and validation set $I_i = \{I_i^t = 50, I_i^v = 50\}$. We gather a random pool of negative images and split them into a training set and validation set $\overline{I} = \{\overline{I}^t = 50, \overline{I}^v = 50\}$. We train the SVM classifier f_i and calculate the score S_i using the validation set. The edge weights $W_{i,j}$ are obtained by calculating the score of the ith node classifier f_i on the jth node validation set $\{I_j^v, \overline{I}^v\}$. We merge the visually similar semantic senses and sample the representative subset of space by setting the cost to be 0.3.

To prune noisy semantic senses, we retrieve the top 500 images for each semantic sense. We then use the previously trained classifier f_i to select the most positive $k = 200$ images from the rest 450 images (the training data and testing data have no duplicates). We represent the selected semantic sense G_I with the compound feature $\delta_{f,k}$ of the most positive 200 images. There are multiple methods for learning the weighting function (e.g., cross-validation or logistic regression), we follow (Carneiro et al. 2007) and take cross-validation to learn the weighting function. To this end, we label $D_P = 500$ positive bags and $D_N = 500$ negative bags. This labelling is only for the bag, we do not label every image in the bag. Labelling work only needs to be done once to learn the weighting function and the bag classification rule (3.9). The learned classification rule (3.9) will also be used to prune noisy bags (corresponding to noisy semantic senses) which have no specific visual senses.

After pruning the noisy semantic senses, we set the rest as the final selected semantic senses. For each selected semantic sense, we collect the training data (500 images) from the image search engine. We take the MIL based method to handle instance-level noisy images and select the positive training data, to train the visual classifier. The negative training data is drawn from a "background" category, which in our case is the union of all other categories that we are asked to classify. The visual feature in our experiment is 4096 dimensional deep features (based on AlexNet (Krizhevsky et al. 2012)).

Baselines

To quantify the performance of our proposed approach, we compare the sense-specific image classification ability of our approach with two sets of baseline methods. For all the baseline methods, we adopt the same parameter configuration as described in their original works. Our baselines include:

• Knowledge-based methods. These baselines consist of Wikipedia method Wiki-MD (Mihalcea 2007), dictionary method Dict-MD (Veronis & Ide 1990) and corpora method Copr-MD (Yarowsky 1992). For all of these three methods, we obtain the multiple semantic senses from human-developed knowledge. We directly retrieve the images from image search engine to learn the visual classifier for each semantic sense (without noisy images removing).

• Combination of text and images based methods. This set of baselines include ISD (Loeff et al. 2006), VSD (Wan et al. 2009), ULVSM (Saenko & Trevor 2009), SDCIT (Chen et al. 2015) and LEAN (Divvala et al. 2014). The ISD approach and SDCIT approach involve two major steps: (1) extracting and weighting text features from the web pages, visual features from the retrieved images, (2) running spectral clustering or co-clustering mechanism on both of the text features and visual features to derive the multiple semantic senses. The VSD approach and ULVSM approach consist of three steps: (1) discovering multiple semantic senses and using the discovered semantic senses to retrieve images, (2) learning probabilistic models for discovered semantic senses, (3) using the probabilistic models to construct visual classifiers. The LEAN approach contains three steps: (1) using Google Books Ngram Corpus to discover multiple semantic senses, (2) using the iterative mechanism to filter noisy semantic senses and images, (3) learning visual classifiers.

Experimental results

Table 3.1 shows the average performance comparison of classification accuracy on the CMU-Poly-30 and MIT-ISD dataset. Fig. 3.5 presents the examples of multiple visual senses discovered by our proposed approach on the CMU-Poly-30 dataset. Fig. 3.6 and Fig. 3.4 demonstrate the detailed performance comparison of classification accuracy on the CMU-Poly-30 and MIT-ISD dataset respectively.

It is interesting to note in Fig. 3.5, our proposed approach not only discovers and distinguishes the sense of “notes” for “$Note$”, but also “galaxy note”, “note tablet” and “music note”. For “$Bass$”, in addition to “bass fish” and “bass guitar”, our approach also discovers and distinguishes the sense of “Mr./Mrs. Bass”.

Compared to knowledge-based methods which discover possible semantic senses through Wikipedia or WordNet, our proposed approach that adopts untagged Google Books Ngram Corpus to discover possible semantic senses is much more exhaustive and general. Method ISD (Loeff et al. 2006) and SDCIT (Chen et al. 2015) which uses webpages can discover multiple semantic senses but overly depends on the collected data. For example, method ISD (Loeff et al. 2006) fails to collect webpages that contain enough semantic senses and visual senses for the given query, it can be seen that in Table 3.1, the performance of is greatly reduced.

From Fig. 3.6 and Fig. 3.4, we achieved the best results in 26 categories on the CMU-Poly-30 dataset. In the 5 categories of dataset MIT-ISD, we obtained the best results in all 5 categories. By observing Table 3.1, the best average performance is achieved by our approach, which produces significant improvements over two sets of baseline methods. The explanation is that the automatically generated sense-specific terms by our approach could return relatively high-precision web images. Meanwhile, the MIL based method can handle the few noises in the training data and train a robust classifier.

From Fig. 3.6, we found that all methods showed higher accuracy in both of the “AK47” and “Motorbike” categories. The explanation is perhaps that

Table 3.1: The average performance comparison of classification accuracy on the CMU-Poly-30 and MIT-ISD dataset.

Method	**Dataset**	
	CMU-Poly-30	MIT-ISD
Wiki-MD	0.498	0.487
Dict-MD	0.529	0.522
Copr-MD	0.549	0.593
ISD	0.555	0.634
VSD	0.728	0.786
ULVSM	0.772	0.803
SDCIT	0.839	0.853
LEAN	0.827	0.814
Ours	**0.884**	**0.897**

the visual patterns of polysemous words “AK47” and “Motorbike” are relatively simpler than other polysemous words. That is to say, the samples are densely distributed in the feature space, and the distribution of the training data and testing data overlaps much more easily.

3.3.2 Re-ranking Search Results

Experimental setting

We collect the top 500 images from Google Image Search engine for semantically ambiguous words: “bass” and “mouse”. We perform a cleanup step for broken links, webpages, end up with 349 and 251 images for “bass” and “mouse” respectively. These images were annotated with one of the several semantic senses by one of the authors. The annotator tried to resist name influence, and make judgments based just on the image. For each query, 2 core semantic senses were distinguished from inspecting the data. The detailed information for these retrieved images is summarized in Table 3.2.

We now evaluate how well the two sets of baseline methods and our

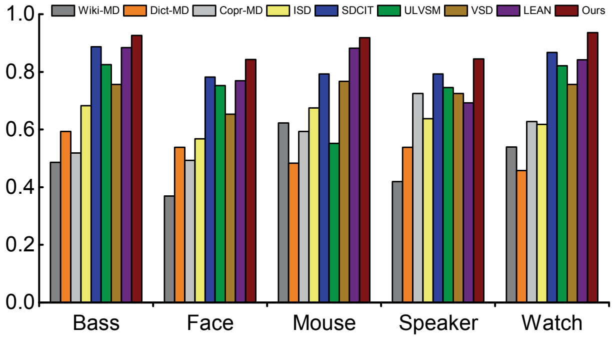

Figure 3.4: The detailed performance comparison over 5 categories on the MIT-ISD dataset.

method can re-rank the retrieved images. For each query, the sense-specific classifiers are trained on the sense-specific web images. Particularly, we use the previously trained sense-specific classifiers in the previous experiment. Retrieved images are then re-ranked by moving the negatively-classified images down to the last rank. For an image d, we compute the probability $P(S_i|d)$ of image d belonging to the ith sense S_i and rank the corresponding images according to the probability of each sense S. $P(S_i|d)$ provides a way to re-rank the images in the original polysemous order. Images belonging to some sibling sense are given lower probabilities and pushed to the back of the rank list.

Baselines

We compare the search results re-ranking ability of our approach with two sets of baseline methods which include knowledge-based methods and the combination of text and images based methods. The knowledge-based methods consist of Wiki-MD (Mihalcea 2007), Dict-MD (Veronis & Ide 1990) and Copr-MD (Yarowsky 1992). The combination of text and images based methods contain ISD (Loeff et al. 2006), VSD (Wan et al. 2009), ULVSM (Saenko & Trevor 2009), LEAN (Divvala et al. 2014), and SDCIT (Chen et al. 2015).

Experimental results

Following (Wan et al. 2009), we evaluate the re-ranking performance by computing the Area Under Curve (AUC) of all senses for "bass" and "mouse". The results are shown in Table 3.3.

From Table 3.2, we observe that there are only 4.6% and 7.5% true noise in the retrieved images for "bass" and "mouse" respectively. Most of the retrieved images are different forms of visual senses for the given query. This indicates that we should first discover the multiple visual senses for the given query. So that we can choose appropriate visual senses as needed to carry out sense-specific images collection. By doing this, we can greatly improve the efficiency of collecting web images, thereby improving the efficiency of learning from the web images.

We observe that the combination of text and images based methods ISD (Loeff et al. 2006), VSD (Wan et al. 2009), ULVSM (Saenko & Trevor 2009), SDCIT (Chen et al. 2015), LEAN (Divvala et al. 2014) and our method are generally better than knowledge-based methods Wiki-MD (Mihalcea 2007), Dict-MD (Veronis & Ide 1990) and Copr-MD (Yarowsky 1992) in Table 3.3. In specific, methods SDCIT (Chen et al. 2015), LEAN (Divvala et al. 2014) and our method achieve better results than other methods. The explanation is that it is necessary to remove noisy images from the training set during the process of classifier learning. Learning directly from the web images without noise removing may affect the performance of the classifier due to the presence of noisy images.

By observing Table 3.3, we achieve the best average performance which consistent with the results of sense-specific image classification. The reason can be explained by the generated sense-specific terms and filtered images of our approach. Compared to knowledge-based methods Wiki-MD (Mihalcea 2007), Dict-MD (Veronis & Ide 1990) and Copr-MD (Yarowsky 1992), our approach does not directly use web images for classifier learning. Instead, we filter the retrieved images to select useful data and then use the selected images to learn classifiers. By doing this, our approach can effectively overcome the impact of noise on the classifiers due to the error index of image search engine. Compared to the combination of text and images based methods ISD (Loeff et al. 2006), VSD (Wan et al. 2009), ULVSM (Saenko & Trevor 2009), LEAN (Divvala et al. 2014) and SDCIT (Chen et al. 2015), the sense-specific terms generated by our approach are more accurate and exhaustive, using our sense-specific terms to retrieve images can return high precision web images, thereby can help us to train sense-specific classifiers to re-rank the search results.

3.4 Conclusions

In this chapter, we focused on one important yet often ignored problem: we argue that the current poor performance of existing methods for image dataset construction is due to the visual polysemy. We solved the problem by allowing sense-specific diversity in search results. Specifically, we presented a new framework for discovering and distinguishing multiple visual senses for polysemous words. Compared to existing methods, our proposed method can not only figure out the right sense but also generates the right mapping between semantic and visual senses. We verified the effectiveness of our approach on the tasks of sense-specific image classification and search results re-ranking. The experimental results demonstrated the superiority of our proposed approach over existing weakly supervised state-of-the-art approaches.

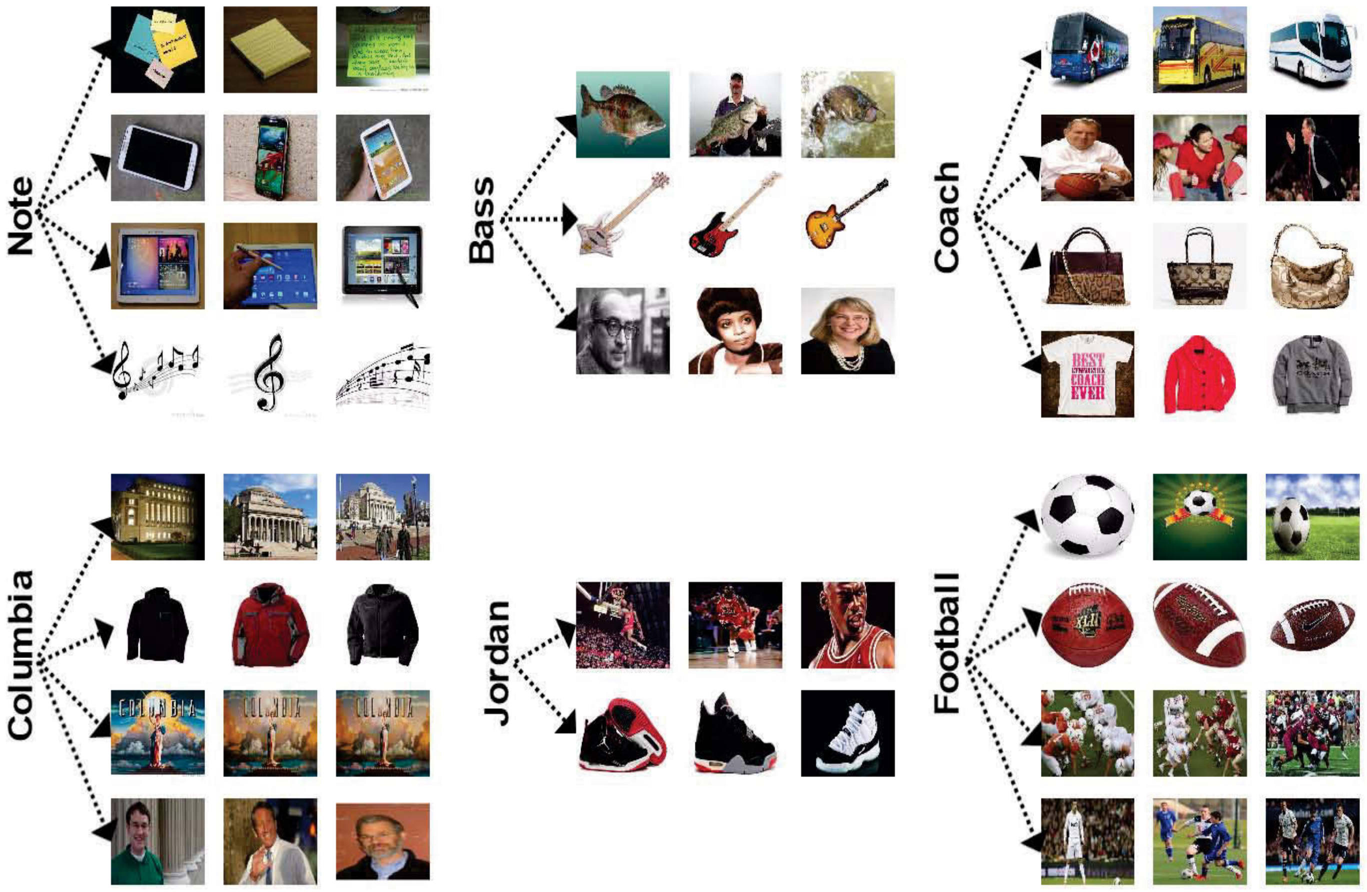

Figure 3.5: Examples of multiple visual senses discovered by our proposed approach. For example, our approach automatically discovers and distinguishes four senses for "*Note*": notes, galaxy note, note tablet and music note. For "*Bass*", it discovers multiple visual senses of: bass fish, bass guitar and Mr./Mrs. Bass *etc.*

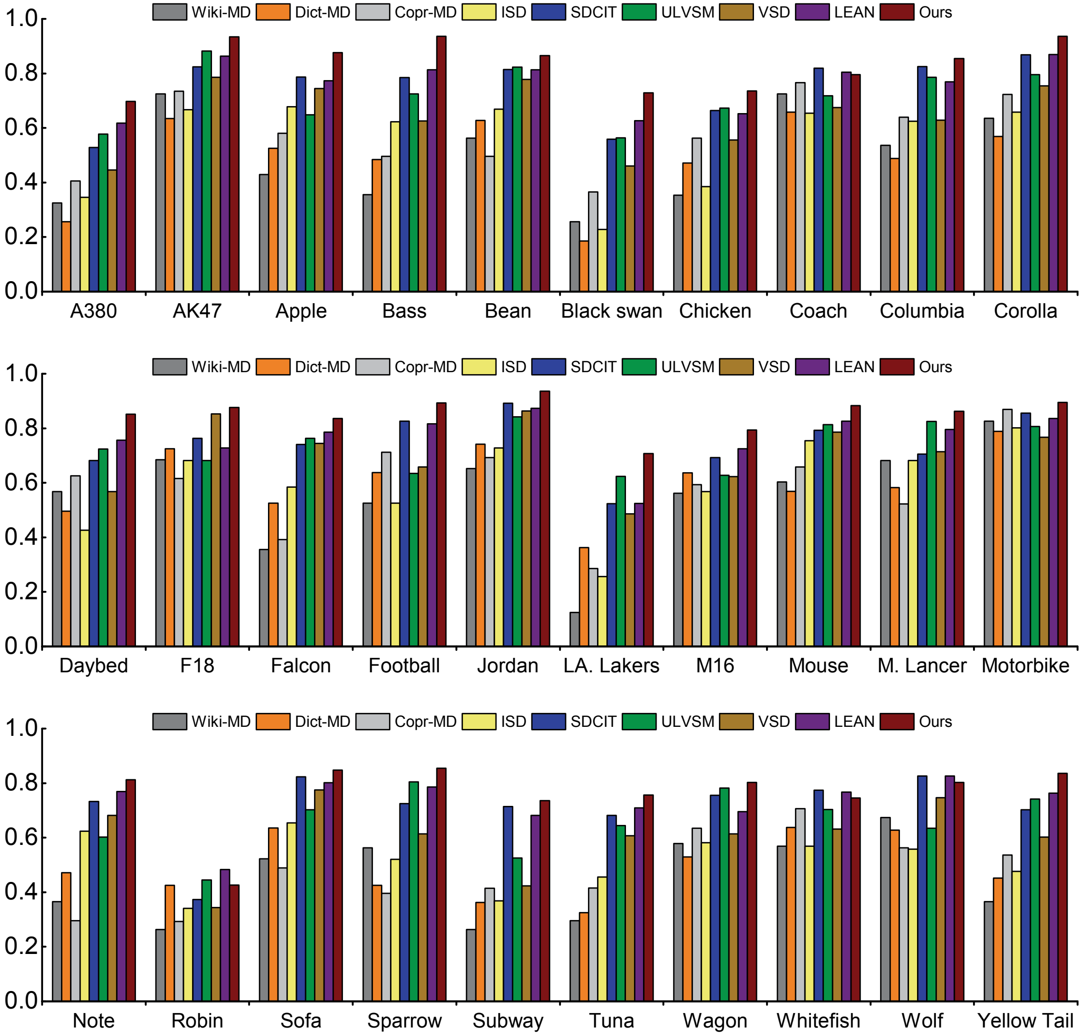

Figure 3.6: The detailed performance comparison over 30 categories on the CMU-Poly-30 dataset.

Table 3.2: Web images for polysemy terms were annotated manually. For each term, the number of annotated images, the semantic senses, the visual senses and their distributions are provided, with core semantic senses marked in boldface.

Query (#Annot. images)	Semantic senses	Visual senses	Numbers of images	Coverage
Bass (349)	1. **bass fish**	fish	159	45.6%
	2. **bass guitar**	musical instrument	154	44.1%
	3. Mr./ Mrs. Bass	people	20	5.7%
	Noise	unrelated	16	4.6%
Mouse (251)	1. **computer mouse**	electronic product	125	49.8%
	2. **little mouse**	animal	81	32.3%
	3. carton mouse	cartoon role	26	10.4%
	Noise	unrelated	19	7.5%

Table 3.3: Area Under Curve (AUC) of all senses for “bass” and “mouse”.

Method	**Semantic senses**						**Average**
	bass fish	bass guitar	M. Bass	Computer mouse	little mouse	carton mouse	
Wiki-MD	0.364	0.429	0.132	0.536	0.623	0.114	0.366
Dict-MD	0.443	0.635	0.205	0.464	0.573	0.186	0.418
Copr-MD	0.504	0.486	0.305	0.624	0.675	0.263	0.476
ISD	0.453	0.526	0.243	0.614	0.536	0.218	0.432
VSD	0.547	0.538	0.239	0.684	0.652	0.226	0.481
ULVSM	0.526	0.615	0.326	0.732	0.735	0.314	0.541
LEAN	0.623	0.658	0.413	0.753	0.785	0.336	0.595
SDCIT	0.658	**0.773**	0.386	0.815	0.845	0.337	0.636
Ours	**0.713**	0.736	**0.572**	**0.835**	**0.873**	**0.436**	**0.694**

Chapter 4

Diversity

4.1 Introduction

Existing methods (Hua & Li 2015, Schroff et al. 2011, Li & Fei-Fei 2010) usually use an iterative mechanism in the process of image selection. However, due to the visual feature distribution of images selected in this way, these datasets tend to have the dataset bias problem (Niu, Li & Xu 2015, Torralba & Efros 2011, Yao, Hua, Shen, Zhang & Tang 2016).

To address the dataset bias problem, a large number of domain-robust approaches have been proposed (Vijayanarasimhan & Grauman 2008, Duan et al. 2011). The images in these methods are partitioned into a set of clusters; each cluster is treated as a "bag" and the images in each bag as "instances". As a result, these tasks can be formulated as multi-instance learning (MIL) problem. Different MIL methods have been proposed in (Vijayanarasimhan & Grauman 2008, Duan et al. 2011). However, the yield for all of these methods is limited by the diversity of training data which was collected with a single query.

To obtain highly diverse candidate images, as well as to overcome the download restrictions of the image search engine, (Divvala et al. 2014) and (Yao, Hua, Shen, Zhang & Tang 2016) proposed the use of multiple query expansions instead of a single query to collect candidate images from the image search engine. The issue remains that these methods still use iterative mechanisms in the process of image selection, which leads to the dataset bias problem (Niu et al. 2015, Torralba & Efros 2011, Yao, Hua, Shen, Zhang & Tang 2016).

Motivated by the situation described above, we target the construction of an image dataset in a scalable way while ensuring the diversity and robustness. The basic idea is to leverage multiple query expansions for initial candidate images collection and to use MIL methods for selecting images from different distributions. We first expand each query to a set of query expansions, from which the visually non-salient and less relevant expansions are filtered out. Then we set the rest as selected query expansions and construct the raw image dataset with these selected query expansions. By treating each selected query expansion as a “bag” and the images therein as “instances”, we formulate image selection and noise removal as a multi-instance learning problem. In this way, images from different distributions will be kept while noise is filtered out.

To verify the effectiveness of our proposed approach, we build an image dataset with 20 categories, which we refer to as DRID-20. We compare the image classification ability, cross-dataset generalization ability and diversity of our dataset with three manually labelled datasets and three automated datasets, to demonstrate the domain robustness of our dataset. We also report the results of object detection on PASCAL VOC 2007, and then compare the object detection ability of our method with weakly supervised and web-supervised methods.

4.2 Domain robust dataset construction

We seek to construct a domain-robust image dataset that can generalize to unseen target domains. As shown in Fig. 4.1, we propose our web-supervised image dataset construction framework by three major steps: query expanding, noisy query expansions filtering and noisy images filtering. We expand

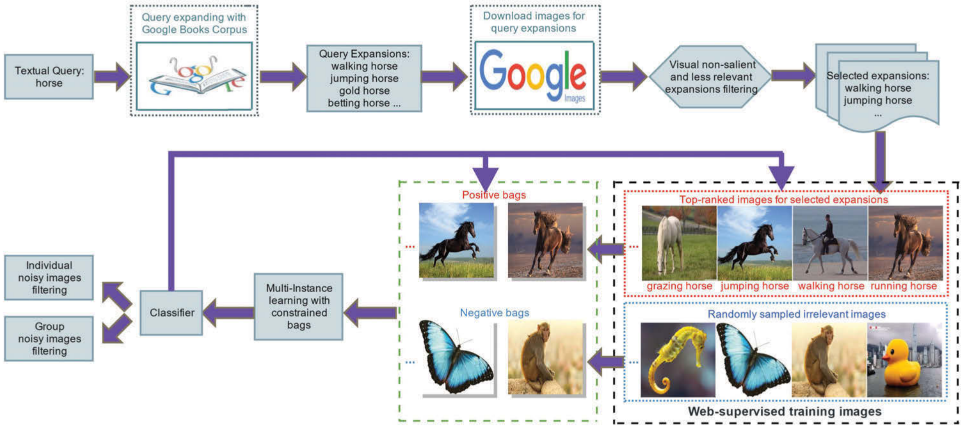
Textual Query: horse
Query expanding with Google Books Corpus
Query Expansions: walking horse jumping horse gold horse betting horse ...
Download images for query expansions
Google
Visual non-salient and less relevant expansions filtering
Selected expansions: walking horse jumping horse ...
Positive bags
Top-ranked images for selected expansions
grazing horse
jumping horse
walking horse
running horse
Negative bags
Randomly sampled irrelevant images
Web-supervised training images
Multi-Instance learning with constrained bags
Classifier
Individual noisy images filtering
Group noisy images filtering

Figure 4.1: Domain robust image dataset construction framework. The input is text query that we would like to build a image dataset for. The outputs are a set of selected images corresponding to the given query.

the query to a set of semantically rich expansions by searching Google Books Ngram Corpus, from which the visually non-salient and less relevant expansions are filtered out. After obtaining the candidate images by retrieving the selected expansions with an image search engine, we treat each selected expansion as a "bag" and the images in each bag as "instances". We then formulate image selection and noisy images filtration as a MIL problem with constrained positive bags. In partifcular, the learned classifiers are used to filter individual noisy images (corresponding to the top-ranked images for selected expansions) and group noisy images (corresponding to the positive bags). Using this approach, images from different distributions will be kept while noisy images are filtered out, and a domain-robust image dataset will be constructed.

4.2.1 Query Expanding

Image datasets constructed by existing methods tend to be highly accurate but usually have weak domain adaptation ability (Niu et al. 2015, Torralba & Efros 2011, Yao, Hua, Shen, Zhang & Tang 2016). To construct a domain-robust image dataset, we expand given query (e.g., “horse”) to a set of query expansions (e.g., “jumping horse, walking horse, roaring horse”) and then use these different query expansions (corresponding images) to reflect the different “visual distributions” of the query.

4.2.2 Noisy Expansions Filtering

Through query expanding, we obtain a comprehensive semantic description for the given query. However, query expanding not only provides all the useful query expansions, but also some noise. These noisy query expansions can be roughly divided into two types: (1) visually non-salient (e.g., “betting horse”) and (2) less relevant (e.g., “sea horse”). Using these noisy query expansions to retrieve images will have a negative effect on dataset accuracy and robustness.

Visual non-salient expansions filtering

From the visual perspective, we aim to identify visually salient and eliminate non-salient query expansions in this step. The intuition is that visually salient expansions should exhibit predictable visual patterns. Hence, we can use the image classifier-based filtering method. For each query expansion, we directly download the top N images from the Google image search engine as positive images (based on the fact that the top few images returned from image search engine tend to be positive), then randomly split these images into a training set and validation set $I_i = \{I_i^t, I_i^v\}$. We gather a random pool of negative images and split them into a training set and validation set $\overline{I} = \{\overline{I}^t, \overline{I}^v\}$. We train a linear support vector machine (SVM) classifier C_i with I_i^t and $\overline{I}^t$ using dense histogram of oriented gradients (HOG) features (Dalal & Triggs 2005). We then use $\{I_i^v, \overline{I}^v\}$ as validation images to calculate the classification results. We declare a query expansion i to be visually salient if the classification results S_i give a relatively high score.

Less relevant expansions filtering

From the relevance perspective, we want to identify both semantically and visually relevant expansions for the given query. The intuition is that relevant expansions should have a relatively small semantic and visual distance; therefore, we use a combined word-word and visual-visual similarity distance-based filtering method. Words and phrases acquire meaning from the way they are used in society. For computers, the equivalent of "society" is "database", and the equivalent of "use" is "a way to search the database" (Cilibrasi & Vitanyi 2007). Normalized Google Distance (NGD) constructs a method to extract semantic similarity distance from the World Wide Web (WWW) using Google page counts. For a search term x and search term y, NGD is defined by:

$$\mathrm{NGD}(x, y) = \frac{\max\{\log f(x), \log f(y)\} - \log f(x, y)}{\log N - \min\{\log f(x), \log f(y)\}} \tag{4.1}$$

where $f(x)$ denotes the number of pages containing x, $f(x, y)$ denotes the number of pages containing both x and y and N is the total number of web pages searched by Google.

We denote the semantic distance of all query expansions by a graph $G_{semantic} = \{N, D\}$ in which each node represents a query expansion and its edge represents the NGD between two nodes. We set the target query as center y and other expansions have a score D_{xy} which corresponds the NGD to the target query. Similarly, we represent the visual distance of query and expansions by a graph $G_{visual} = \{C, E\}$ in which each node represents a query expansion and each edge represents the visual distance between the query and the expansions. We denote the visual distribution of each query expansion by the compound feature $\phi_k = \frac{1}{k} \sum_{i=1}^{k} x_i$ of its first k images from the image search engine. We set the target query as center y and other query expansions have a score E_{xy} which corresponds to the Euclidean distance to the target query.

The semantic distance D_{xy} and visual distance E_{xy} will be used to construct a new two-dimensional feature $V = [D_{xy}; E_{xy}]$. The problem is to calculate the importance weight w and bias penalty b in decision function $f(x) = w^T x + b$ to determine whether or not the expansion is relevant. There are many methods of obtaining these coefficients w and b. Here we take the linear SVM to work around this problem. Although the linear SVM is not the prevailing state-of-the-art method for classification, we find our method to be effective in pruning irrelevant query expansions.

We set the remainder which is not filtered out as the selected expansions and construct raw image dataset by retrieving the top N images from image search engine with these selected query expansions. Regardless of the fact that our method is not able to remove noisy expansions thoroughly in most cases, the raw image dataset constructed by our method still achieves much higher accuracy than directly using the Google image data. Besides, the raw image dataset constructed through the selected query expansions has much richer visual distributions.

4.2.3 Noisy Images Filtering

Although the Google image search engine has ranked the returned images, some noisy images may still be included. In addition, a few noisy expansions which are not filtered out will also bring noisy images to the raw image dataset. In general, these noisy images can be divided into two types: group noisy images (caused by noisy query expansions) and individual noisy images (as a result of the error index of the image search engine). To filter these group and individual noisy images while retaining the images from different distributions, we use MIL methods instead of iterative methods in the process of image selection and noise removal.

By treating each selected expansion as a “bag” and the images corresponding to the expansion as “instances”, we formulate a multi-instance learning problem by selecting a subset of bags and a subset of images from each bag to construct the domain-robust image dataset. Since the precision of images returned from image search engine tends to be relatively high, we define each positive bag as at least having a portion of δ positive instances which effectively filter group noisy images caused by noisy query expansions.

We denote each instance as x_i with its label $y_i \in \{0,1\}$, where i =1,...,n. We also denote the label of each bag B_I as $Y_I \in \{0,1\}$. The transpose of a vector or matrix is represented by superscript $'$ and the element-wise product between two matrices is represented by $\odot$. We define the identity matrix as $\mathbf{I}$ and $\mathbf{0}$, $\mathbf{1} \in \Re^n$ denote the column vectors of all zeros and ones, respectively. The inequality $\mathbf{u} = [u_1, u_2...u_n]' \geq \mathbf{0}$ means that $u_i \geq 0$ for i =1,...,n.

Filtering individual noisy images

The decision function for filtering individual noisy images is assumed in the form of $f(x) = w'\varphi(x) + b$ and has to be learned from the raw image dataset. We employ the formulation of Lagrangian SVM, in which the square bias penalty b^2 and the square hinge loss for each instance are used in the objective function. The decision function can be learned by minimizing the following structural risk function:

$$\min_{y,w,b,\rho,\varepsilon_i} \frac{1}{2}\left(\|w\|^2 + b^2 + C\sum_{i=1}^{n}\varepsilon_i^2\right) - \rho \tag{4.2}$$

$$\text{s.t. } y_i(w'\varphi\left(x_i\right) + b) \geq \rho - \varepsilon_i, i = 1, ...n. \tag{4.3}$$

$$\begin{gathered}\sum_{i:x_i \in B_I} \frac{y_i + 1}{2} \geq \delta\left|B_I\right| \quad for \quad Y_I = 1, \\ y_i = 0 \qquad for \quad Y_I = 0\end{gathered} \tag{4.4}$$

where φ is a mapping function that maps x from the original space into a high dimensional space $\varphi(x)$, $C > 0$ is a regularization parameter and ε_i values are slack variables. The margin separation is defined as $\rho/\left\|w\right\|$. $y = [y_1...y_n]'$ means the vector of instance labels, $\lambda = \{\mathbf{y}|y_i \in \{0,1\}\}$ and $\mathbf{y}$ satisfies constraint (4.4). By introducing a dual variable α_i for inequality constraint (4.3) and kernel trick $k_{ij} = \varphi(x_i)'\varphi(x_j)$, we arrive at the optimization problem below:

$$\min_{\mathbf{y}\in\lambda} \max_{\alpha} -\frac{1}{2}(\sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j k_{ij} + \sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j + \frac{1}{C}) \tag{4.5}$$

where $\alpha_i \geq 0$, $\sum_{i=1}^{n} \alpha_i = 1$ and $\alpha = [\alpha_1, \alpha_2 ... \alpha_n]'$. By defining $\mathbf{K} = [k_{ij}]$ as a $n \times n$ kernel matrix, $\nu = \{\alpha | \alpha \geq \mathbf{0}, \alpha'\mathbf{1} = 1\}$ and $\widetilde{\mathbf{K}} = \mathbf{K} + \mathbf{11'}$ as a $n \times n$ transformed kernel matrix for the augmented feature mapping $\widetilde{\varphi}(x) = [\varphi(x)'. 1]'$ of kernel $\widetilde{k_{ij}} = \widetilde{\varphi}(x_i)'\widetilde{\varphi}(x_j)$. (4.5) can be rewritten as follows:

$$\min_{\mathbf{y} \in \lambda} \max_{\alpha \in \nu} -\frac{1}{2}\alpha'(\widetilde{\mathbf{K}} \odot \mathbf{yy}' + \frac{1}{C}\mathbf{I})\alpha. \tag{4.6}$$

(4.6) is a mixed integer programming problem with respect to the instance labels $y_i \in \{0, 1\}$. We take the Label-Generating MMC (LG-MMC) algorithm proposed in (Li, Tsang, Kwok & Zhou 2009) to solve this mixed integer programming problem. We first consider interchanging the order of $\max_{\alpha \in \upsilon}$ and $\min_{\mathbf{y} \in \lambda}$ in (4.6) and obtain:

$$\max_{\alpha \in \nu} \min_{\mathbf{y} \in \lambda} -\frac{1}{2}\alpha'(\widetilde{\mathbf{K}} \odot \mathbf{yy}' + \frac{1}{C}\mathbf{I})\alpha. \tag{4.7}$$

According to the minmax theorem (Kim & Boyd 2008), the optimal objective of (4.6) is an upper bound of (4.7). We rewrite (4.7) as:

$$\max_{\alpha \in \nu} \left\{ \max_{\theta} -\theta | \theta \geq \frac{1}{2}\alpha'(\widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'} + \frac{1}{C}\mathbf{I})\alpha, \forall \mathbf{y}^t \in \lambda \right\} \tag{4.8}$$

$\mathbf{y}^t$ is any feasible solution in λ. For the inner optimization sub-problem, let $u_t \geq 0$ be the dual variable for inequality constraint. Its Lagrangian can be obtained as:

$$-\theta + \sum_{t: \mathbf{y}_t \in \lambda} u_t(\theta - \frac{1}{2}\alpha'(\widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'} + \frac{1}{C}\mathbf{I})\alpha). \tag{4.9}$$

Setting the derivative of (4.9) with respect to θ to zero, we have $\sum u_t = 1$. $\mathbf{M} = \{\mathbf{u} | \sum u_t = 1, u_t \geq 0\}$ is denoted as the domain of $\mathbf{u}$, where $\mathbf{u}$ is the vector of u_t. The inner optimization sub-problem is replaced by its dual and (4.8) can be rewritten as:

$$\max_{\alpha \in \nu} \min_{\mathbf{u} \in \mathbf{M}} -\frac{1}{2}\alpha'(\sum_{t: \mathbf{y}^t \in \lambda} u_t \widetilde{\mathbf{K}} \odot \mathbf{yy}' + \frac{1}{C}\mathbf{I})\alpha$$

or

$$\min_{\mathbf{u} \in \mathbf{M}} \max_{\alpha \in \nu} -\frac{1}{2}\alpha'(\sum_{t: \mathbf{y}^t \in \lambda} u_t \widetilde{\mathbf{K}} \odot \mathbf{yy}' + \frac{1}{C}\mathbf{I})\alpha. \tag{4.10}$$

Here, we can interchange the order of $\max_{\alpha\in\nu}$ and $\min_{\mathbf{u}\in\mathbf{M}}$ because the objective function is concave in α and convex in $\mathbf{u}$. Additionally, (4.10) can be regarded as a multiple kernel learning (MKL) problem (Bach, Lanckriet & Jordan 2004), and the target kernel matrix is a convex combination of base kernel matrices $\left\{\widetilde{\mathbf{K}} \odot \mathbf{y_t}\mathbf{y_t}'\right\}$. Although λ is finite and (4.10) is an MKL problem, we can not directly use existing MKL techniques like (Rakotomamonjy, Bach, Canu & Grandvalet 2008) to solve this problem. The reason is that the exponential number of possible labellings $\mathbf{y}_t \in \lambda$ and the fact that the base kernels are exponential in size make direct MKL computations intractable.

Fortunately, not all the constraints in (4.8) are active at optimality. Thus we can employ a cutting-plane algorithm (Kelley 1960) to find a subset $\zeta \in \lambda$ of the constraints that can well approximate the original optimization problem. The detailed solutions of the cutting-plane algorithm for (4.10) are described in Algorithm 4.1. Finding the most violated constraint $\mathbf{y}^t$ is the most challenging aspect of the cutting-plane algorithm.

According to (4.5), the most violated $\mathbf{y}^t$ is equivalent to the following optimization problem:

$$\max_{\mathbf{y}\in\lambda} \sum_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j k_{ij}. \tag{4.11}$$

We solve this integer optimization problem by enumerating all possible candidates of $\mathbf{y}^t$. Here we only enumerate the possible labelling candidates of the instances in positive bags as all instances in the negative bags are assumed to be negative in our paper. Lastly, we can derive the decision function from the raw image dataset for the given query as:

$$f(x) = \sum_{i:\alpha_i\neq 0} \alpha_i \widetilde{y}_i \widetilde{k}(x, x_i) \tag{4.12}$$

where $\widetilde{y}_i = \sum_{t:\mathbf{y}^t\in\lambda} u_t y_i^t$ and $\widetilde{k}(x, x_i) = k(x, x_i) + 1$. The decision function will be used to filter individual noisy images in each bag which correspond to selected query expansions.

Algorithm 4.1 Cutting-plane algorithm for solving (4.10)

```
1: Initialize $y_i = Y_I$ for $x_i \in B_I$ as $\mathbf{y}^1$, and set $\zeta = \{\mathbf{y}^1\}$;
2: Use MKL to solve $\alpha$ and $\mathbf{u}$ in (4.10) with $\zeta$;
3: Select most violated $\mathbf{y}^t$ with $\alpha$ and set $\zeta = \mathbf{y}^t \cup \zeta$;
4: Repeat step 2 and step 3 until convergence.
```

Filtering group noisy images

To filter group noisy images, we represent bag B_I with the compound feature $\phi_{f,k}$ of its first k positive instances:

$$\phi_{f,k}(B_I) = \frac{1}{k} \sum_{x_i \in \Psi^*_{f,k}(B_I)} x_i \tag{4.13}$$

with

$$\Psi^*_{f,k}(B_I) =^{\arg\max}_{\Psi \subseteq B_I, |\Psi| = k} \sum_{x_i \in \Psi} f(x_i). \tag{4.14}$$

We refer to the instances in $\Psi^*_{f,k}(B_I)$ as the first k instances of B_I according to classifier f (see Equation 4.12). Since the closer of images in B_I from the bag center, the higher probability of these images to be relevant to the bag. The assignment of relatively heavier weights to images which have short distance to bag center would increase the accuracy of classifying bag B_I to be positive or negative, then increase the efficiency of filtering noisy group images. Following (Carneiro et al. 2007), we assume $\xi_i = [1+\exp(\alpha \log d(x_i) + \beta)]^{-1}$ to be a weighting function, $d(x_i)$ represents the Euclidean distance of images x_i from the bag center, $\alpha \in \mathbb{R}_{++}$ and β are scaling and offset parameters which can be determined by cross-validation. The representation of (4.13) for bag B_I can be generalized to a weighted compound feature:

$$\phi_{f,k}(B_I) = \phi(X, h^*) = \frac{Xh^*}{\xi^T h^*} \tag{4.15}$$

with

$$h^* =^{\arg\max}_{h \in H} f(\frac{Xh}{\xi^T h}), \quad \text{s.t.} \ \sum_i h_i = k \tag{4.16}$$

where $X = [x_1, x_2, x_3.., x_i] \in \mathbb{R}^{D\times i}$ is a matrix whose columns are the instances of bag B_I, $\xi = [\xi_1, \xi_2, \xi_3 ... \xi_i]^T \in \mathbb{R}^i_{++}$ are the vectors of weights, and $h^* \in H = \{0,1\}^i \setminus \{0\}$ ($\sum_i h_i = k$) is an indicator function for the first k positive instances of bag B_I.

Then classifying rule of bag B_I to be selected or not is:

$$f_\omega(X) =\max_{h\in H} \omega^T \phi(X,h), \quad \sum_i h_i = k \tag{4.17}$$

where $\omega \in \mathbb{R}^D$ is the vector of classifying coefficients, $\phi(X,h) \in \mathbb{R}^D$ is the feature vector of (4.15), h is a vector of latent variables and H is the hypothesis space $\{0,1\}^i \setminus \{0\}$. The learning problem is to determine the parameter vector ω.

Given a training set $\tau = \{B_I, Y_I\}_{I=1}^n$, this is a latent SVM learning problem:

$$\min_\omega \frac{1}{2} \|\omega\|^2 + C \sum_{I=1}^n \max\left(0, 1 - Y_I f_\omega\left(X_{B_I}\right)\right). \tag{4.18}$$

Before solving (4.18), we first solve the classifying rule of (4.17). It is necessary to solve the below following problem:

$$\max_{h\in H} \frac{\omega^T X h}{\xi^T h}, \quad \text{s.t.} \sum_i h_i = k. \tag{4.19}$$

This is an integer linear-fractional programming problem. Since $\xi \in \mathbb{R}^i_{++}$, (4.19) is identical to the relaxed problem:

$$\max_{h\in \text{ß}^i} \frac{\omega^T X h}{\xi^T h}, \quad \text{s.t.} \sum_i h_i = k \tag{4.20}$$

where $\text{ß}^i = [0,1]^i$ is a unit box in $\mathbb{R}^i$. (4.20) is a linear-fractional programming problem and can be reduced to a linear programming problem of $i+1$ variables and $i+2$ constraints (Boyd & Vandenberghe 2004).

In this work, we take the concave-convex procedure (CCCP) (Yuille & Rangarajan 2003) algorithm to solve (4.18). We rewrite the objective of

Algorithm 4.2 Concave-convex procedure for solving (4.21)

```
1: Initialize $\omega$ with SVM by setting $h = \mathbf{1} \in \mathbb{R}^i$;
2: Compute a convex upper bound using the current model for the second
   term of (4.21);
3: Minimize this upper bound by solving a structural SVM problem via the
   proximal bundle method (Kiwiel 1990);
4: Repeat step 2 and step 3 until convergence.
```

(4.18) as two convex functions:

$$\begin{gathered}\min_{\omega}\left[\frac{1}{2}\left\|\omega\right\|^2 + C\sum_{I\in D_n}\max\left(0, 1 + f_\omega\left(X_{B_I}\right)\right) + \right.\\ \left. C\sum_{I\in D_p}\max\left(f_\omega\left(X_{B_I}\right), 1\right)\right] - \left[C\sum_{I\in D_p} f_\omega\left(X_{B_I}\right)\right]\end{gathered} \tag{4.21}$$

where D_p and D_n are positive and negative training sets respectively. The detailed solutions of the CCCP algorithm for (4.21) are described in Algorithm 4.2. Lastly, we obtain the bag classifying rule as (4.17) to filter group noisy images which correspond to noisy query expansions.

In summary, the existing automatic methods reduce the cost of manual annotation by leveraging the generalization ability of machine learning models. However, this generalization ability is affected by both the quality of the initial candidate images and the capability of models to retain images from different distributions. Previous works primarily focus on accuracy and scale, and most use an iterative mechanism for the image selection process which often results in a dataset bias problem. To the best of our knowledge, this is the first proposal for automatic domain-robust image dataset construction. We achieve the domain adaptation ability of our dataset by maximizing both the initial candidate images and the final selected images from different distributions.

4.3 Experiments

To demonstrate the effectiveness of our approach, we have constructed an image dataset with 20 categories. We compare the image classification ability, cross-dataset generalization ability, and diversity of our dataset with three manually labelled and three automated datasets. We also report the object detection ability of our dataset and compare it with weakly supervised and web-supervised state-of-the-art methods.

4.3.1 Image Dataset DRID-20 Construction

Since most existing weakly supervised and web-supervised learning methods were evaluated on the PASCAL VOC 2007 dataset, we choose the 20 categories in PASCAL VOC 2007 as the target categories for the construction of DRID-20.

For each given query (e.g.,"horse"), we first expand the given query to a set of query expansions with POS. To filter visual non-salient expansions, we retrieve the top $\mathbf{N} = 100$ images from the image search engine as positive images (in spite of the fact that noisy images might be included). Set the training set and validation set $I_i = \{I_i^t = 75, I_i^v = 25\}$, $\overline{I} = \{\overline{I}^t = 25, \overline{I}^v = 25\}$. By experimentation, we declare a query expansion i to be visually salient if the classification result $(S_i \geq 0.7)$ returns a relatively high score.

To filter the less relevant expansions, we select n_+ positive training samples from these expansions that have a small semantic or visual distance. We calculate the semantic distance and visual distance between the different queries (e.g., "horse" and "cow") to obtain the n_- negative training samples. Here, we set $n = 1000$ and train a classifier based on linear SVM to filter less relevant expansions.

The first $\mathbf{N} = 100$ (for category "plant" expansions, $\mathbf{N} = 350$) images are retrieved from image search engine for each selected query expansion to construct the raw image dataset. We treat the selected query expansions as positive bags and images therein as instances. Specifically, we define each positive bag as having at least a portion of $\delta = 0.7$ positive instances. Negative bags can be obtained by randomly sampling a few irrelevant images. MIL methods are applied to learn the decision function (4.12) for individual noisy images filtering. The decision function (4.12) is also used to select the most k positive instances in each bag, representing this bag for group noisy images filtering. The value of k for different categories may be different. In general, categories with larger query expansions tend to select a smaller value. There are multiple methods for learning the weighting function (e.g., logistic regression or cross-validation), here we follow (Carneiro et al. 2007) and use cross-validation to learn the weighting function. To this end, we label 10 datasets, each containing 100 positive bags and 100 negative bags. This labeling is also for the bag, we do not label instance-level images in the bag. The positive bags and negative bags each have 50 images. Labelling only needs to be carried out once to learn the weighting function and the weighted bag classification rule (4.17). The learned weighted bag classification rule (4.17) would be used to filter noisy bags (corresponding to group noisy images). For better comparison with other datasets, we evenly select positive images from positive bags to construct the dataset DRID-20. Each category in DRID-20 has 1000 images and this dataset has been released publicly on the website.

4.3.2 Comparison of Classification Ability, Cross-dataset Generalization Ability, and Dataset Diversity

Experimental setting

We chose PASCAL VOC 2007 as the third-party testing benchmark dataset for comparing the image classification ability of our dataset with other baseline datasets. For this experiment, the same categories between various datasets are compared. Specifically, we compare the category “airplane”, “bird”, “cat”, “dog”, “horse” and “car/automobile” between STL-10(Coates et al. 2011), CIFAR-10 (Krizhevsky & Hinton 2009) and DRID-20. We sequentially select [200,400,600,800,1000] training images from CIFAR-10, STL-10 and DRID-20 as the positive training images, and use 1000 fixed irrelevant images as the negative training images to learn the image classifiers.

For comparison with ImageNet (Deng et al. 2009), Optimol (Li & Fei-Fei 2010), Harvesting (Schroff et al. 2011) and AutoSet (Yao, Zhang, Shen, Hua, Xu & Tang 2016), we use all the 20 categories among these datasets. In specific, we randomly select 500 training images for each category from these datasets as the positive training images. Similarly, we use 1000 fixed irrelevant images as the negative training images to learn the image classifiers. We then test the performance of these classifiers on the corresponding categories of the PASCAL VOC 2007 dataset. We repeat the above experiment ten times and use the average performance as the final performance for each dataset. The image classification ability of all datasets for each category is shown in Fig. 4.2 and Fig. 4.3.

For the comparison of cross-dataset generalization ability, we randomly select 200 images for each category as the testing data. For the choice of training data, we sequentially select [200,300,400,500,600,700,800] images per category from various datasets as the positive training samples, and use 1000 fixed irrelevant images as the negative training samples to learn the image classifiers. The training images in each category are selected randomly. In addition, the training data and testing data have no duplicates. Like the comparison of image classification ability, we also compare the category "airplane", "bird", "cat", "dog", "horse" and "car/automobile" among STL-10 (Coates et al. 2011), CIFAR-10 (Krizhevsky & Hinton 2009) and DRID-20. For comparison with ImageNet (Deng et al. 2009), Optimol (Li & Fei-Fei 2010), Harvesting (Schroff et al. 2011) and AutoSet (Yao, Zhang, Shen, Hua, Xu & Tang 2016), we also use all the 20 categories among these datasets. The average classification accuracy represents the cross-dataset generalization ability of one dataset on another dataset. The experimental results are shown in Fig. 4.4 and Fig. 4.6 respectively.

For the comparison of dataset diversity, we select five common categories “airplane”, “bird”,“cat”,“dog” and “horse” in STL-10, ImageNet, and DRID-20 as testing examples. Following method (Deng et al. 2009) and (Collins et al. 2008), we compute the average image of each category and measure the lossless JPG file size. In particular, we resize all images in STL-10, ImageNet, DRID-20 to 32×32 images, and create average images for each category from 100 randomly sampled images. Fig. 4.5 (a) presents the lossless JPG file sizes of five common categories in dataset DRID-20, ImageNet and STL-10. The example and average images for five categories in three datasets are shown in Fig. 4.5 (b).

For image classification ability and cross-dataset generalization ability comparison, we set the same options for all datasets. Particularly, we set the type of SVM as C-SVC, the type of kernel as a radial basis function and all other options as the default LIBSVM options. For all datasets, we extract the same dense histogram of oriented gradients (HOG) feature (Dalal & Triggs 2005) and train one-versus-all classifiers.

Baselines

To validate the performance of our dataset, we compare the image classification ability, cross-dataset generalization ability and dataset diversity of our dataset DRID-20 with two sets of baselines:

- Manually labelled datasets. The manually labelled datasets include STL-10 (Coates et al. 2011), CIFAR-10 (Krizhevsky & Hinton 2009) and ImageNet (Deng et al. 2009). The STL-10 dataset has ten categories, and each category of which contains 500 training images and 800 test images. All of the images are color 96 × 96 pixels. The CIFAR-10 dataset consists of 32×32 images in 10 categories, with 6000 images per category. ImageNet is an image dataset organized according to the WordNet hierarchy. It provides an average of 1000 images to illustrate each category.

- Automated datasets. The automated datasets contain Optimol (Li & Fei-Fei 2010), Harvesting (Schroff et al. 2011) and AutoSet (Yao, Zhang, Shen, Hua, Xu & Tang 2016). For (Li & Fei-Fei 2010), 1000 images for each category are collected by using the incremental learning method. Following (Schroff et al. 2011), we firstly obtain the candidate images from the web search and rank the returned images by the text information. Then we use the top-ranked images to learn visual classifiers to re-rank the images once again. We select the categories in DRID-20 as the target queries and accordingly obtain the multiple textual metadata. Following the proposed method in (Yao, Zhang, Shen, Hua, Xu & Tang 2016), we take iterative mechanisms for noisy images filtering and construct the dataset. In total, we construct 20 same categories as DRID-20 for Optimol, Harvesting, and AutoSet.

Experimental results for image classification

By observing Fig. 4.2 and Fig. 4.3, we make the following conclusions:

It is interesting to observe that the categories “airplane”, “tv” and “plant” have a relatively higher classification accuracy than other categories with a small amount of training data. A possible explanation is that the scenes and visual patterns of “airplane”, “tv” and “plant” are relatively simpler than other categories. Even with a small amount of training data, there is still a large number of positive patterns in both auxiliary and target domains. That is to say, the samples are densely distributed in the feature space, and the distribution of the two domains overlaps much more easily.

CIFAR-10 exhibits a much worse performance on image classification than STL-10 and DRID-20 according to its accuracy over six common categories. This demonstrates that the classifier learned with the training data from the auxiliary domain performs poorly on the target domain. The explanation is perhaps that the data distributions of CIFAR-10 are quite different from those of the PASCAL VOC 2007 dataset. The CIFAR-10 dataset has a more serious dataset bias problem than STL-10 and DRID-20.

STL-10 performs much better on category “dog” than CIFAR-10 and DRID-20 when the number of training data is 400. The explanation is that STL-10 may have more effective visual patterns than CIFAR-10 and DRID-

20 on category "dog" with 400 training data. On the other hand, the positive samples from CIFAR-10 and DRID-20 are distributed sparsely in the feature space with 400 training images. It is likely that there is less overlap between the auxiliary and target domains for CIFAR-10 and DRID-20.

DRID-20 outperforms the automated datasets in terms of average accuracy in 20 categories, which demonstrates the domain robustness of DRID-20. A possible explanation is that our DRID-20 dataset, being constructed by multiple query expansions, has many more visual patterns or feature distributions than Harvesting and Optimol. At the same time, compared to AutoSet which uses iterative mechanisms in the process of image selection, MIL mechanisms can maximize the retention of useful visual patterns. Thus, our dataset has a better image classification ability.

Experimental results for cross-dataset generalization

Cross-dataset generalization measures the performance of classifiers learned from one dataset and tested on another dataset. It indicates the robustness of dataset (Torralba & Efros 2011, Yao, Hua, Shen, Zhang & Tang 2016). By observing Fig. 4.4 and Fig. 4.6, we draw the following conclusions:

Compared to STL-10 and DRID-20, CIFAR-10 has a poor cross-dataset generalization ability except on its own dataset. The explanation is that the data distributions of its auxiliary domain and target domain are strongly related, making it difficult for other datasets to exceed its performance when tested on CIFAR-10. All images in CIFAR-10 are cut to 32×32 and the objects in these images are located in the middle of the image. Besides, these images contain relatively fewer other objects and scenes. The images in STL-10 are 96×96 and are full size in DRID-20. These images not only contain target objects but also include a large number of other scenarios and objects. Based on these conditions, CIFAR-10 has a serious dataset bias problem which coincides with its average cross-dataset generalization performance.

AutoSet is better than Optimol, Harvesting, and ImageNet but slightly worse than DRID-20, possibly because the distribution of samples is relatively rich. AutoSet is constructed using multiple textual meta-data and the objects of its images have variable appearances, positions, viewpoints, and poses.

DRID-20 outperforms CIFAR-10, STL-10, ImageNet, Optimol, Harvesting and AutoSet in terms of average cross-dataset performance, which demonstrates the domain robustness of DRID-20. This may be because DRID-20 constructed by multiple query expansions and MIL selection mechanisms has much more effective visual patterns than other datasets given the same number of training samples. In other words, DRID-20 has a much richer feature distribution and is more easily overlapped with unknown target domains.

Experimental results for dataset diversity

The lossless JPG file size of the average image for each category reflects the amount of information in an image. The basic idea is that a diverse image dataset will result in a blurrier average image, the extreme being a gray image. Meanwhile, an image dataset with limited diversity will result in a more structured, sharper average image. Therefore, we expect the average image of a more diverse image dataset to have a smaller JPG file size. By observing Fig. 4.5:

DRID-20 has a slightly smaller JPG file size than ImageNet and STL-10 which indicates the diversity of our dataset. This phenomenon is universal for all five categories. It can be seen that the average image of DRID-20 is blurred and it is difficult to recognize the object, while the average image of ImageNet and STL-10 is relatively more structured and sharper.

DRID-20 is constructed with the goal that images in this dataset should exhibit domain robustness and be able to effectively alleviate the dataset bias problem. To achieve domain robustness, we not only consider the source of the candidate images but also retain the images from different distributions.

4.3.3 Comparison of Object Detection Ability

To compare the object detection ability of our collected data with other baselines (Divvala et al. 2014, Felzenszwalb, Girshick & Ramanan 2010, Siva & Xiang 2011, Prest, Leistner, Civera & Ferrari 2012, Yao, Zhang, Shen, Hua, Xu & Tang 2016), we selected PASCAL VOC 2007 as the test data.

Experimental setting

For each query expansion, we train a separate DPM to constrain the visual variance. We resize images to a maximum of 500 pixels and ignore images with extreme aspect ratios (aspect ratio > 2.5 or < 0.4). To avoid getting stuck to the image boundary during the latent re-clustering step, we initialize our bounding box to a sub-image within the image that ignores the image boundaries. Following (Felzenszwalb et al. 2010), we also initialize components using the aspect-ratio heuristic. Some of the components across different query expansion detectors ultimately learn the same visual pattern. For example, the images corresponding to the query expansion "walking horse" are similar to the images corresponding to "standing horse". In order to select a representative subset of the components and merge similar components, we represent the space of all query expansions components by a graph $G = \{C, E\}$, in which each node represents a component and each edge represents the visual similarity between them. The score d_i for each node corresponds to the average precision. The weight on each edge $e_{i,j}$ is obtained by running the jth component detector on the ith component set. We solve the same objective function proposed in (Divvala et al. 2014) to select the representative components S $(S \subseteq V)$:

$$\max_{S} \sum_{i \in V} d_i \cdot \vartheta(i, S) \tag{4.22}$$

where ϑ is a soft coverage function that implicitly pushes for diversity:

$$\vartheta(i, S) = \begin{cases} 1 & i \in S \\ 1 - \prod_{j \in S}(1 - e_{i,j}) & i \notin S. \end{cases} \tag{4.23}$$

After the representative subset of components has been obtained, we augment them with the method described in (Felzenszwalb et al. 2010) and subsequently merge all the components to produce the final detector.

Baselines

To validate the object detection ability of our collected data, we compare our approach with three sets of baselines:

- Weakly supervised methods. The weakly supervised learning methods include WSL (Siva & Xiang 2011) and SPM-VID (Prest et al. 2012). WSL uses weak human supervision (VOC data with image-level labels for training) and initialization from objectness. SPM-VID is trained on manually selected videos without bounding boxes and shows results in 10 out of 20 categories.
- Web-supervised methods. Such methods include WSVCL (Divvala et al. 2014) and IDC-MTM (Yao, Zhang, Shen, Hua, Xu & Tang 2016). WSVCL takes web supervision and then trains a mixture DPM detector for the object. IDC-MTM collects candidate images with multiple textual metadata and filters these images using an iterative method. Images which are not filtered out are then selected as positive training images for mixture DPM detector learning.
- Fully supervised method. The fully supervised method includes OD-DPM (Felzenszwalb et al. 2010). OD-DPM is a fully supervised object detection method and it is a possible upper bound for weakly supervised and web-supervised approaches.

Experimental results for object detection

We report the performance of object detection on PASCAL VOC 2007 test set. Table 4.1 shows the results of our proposed method and compares it to the state-of-the-art weakly supervised and web-supervised methods (Siva & Xiang 2011, Prest et al. 2012, Divvala et al. 2014, Yao, Zhang, Shen, Hua, Xu & Tang 2016). By observing the Table 4.1, we draw the following conclusions:

Compared to WSL and SPM-VID (which use weak supervision) and OD-DPM (which uses full supervision), the training sets of our proposed approach and WSVCL, IDC-MTM do not need to be labelled manually. Nonetheless, the results of our proposed approach and WSVCL, IDC-MTM surpass the previous best results of weakly supervised object detection methods WSL, SPM-VID. A possible explanation is perhaps that both our approach and that of WSVCL, IDC-MTM use multiple query expansions for candidate image collection, and the training data collected by our approach and WSVCL, IDC-MTM are richer and contain more effective visual patterns.

In most cases, our method surpasses the results obtained from WSVCL, IDC-MTM, which also uses web supervision and multiple query expansions for candidate images collection. The explanation for this is that we use different mechanisms for the removal of noisy images. Compared to WSVCL, IDC-MTM which uses iterative mechanisms in the process of noisy images filtering, our approach applies a MIL method for removing noisy images. This maximizes the ability to retain images from different data distributions while filtering out the noisy images.

Our approach outperforms the weakly supervised and web-supervised methods (Siva & Xiang 2011, Prest et al. 2012, Divvala et al. 2014, Yao, Zhang, Shen, Hua, Xu & Tang 2016). The main reason being that our training data is generated using multiple expansions and MIL filtering mechanisms. Thus, our data contains much richer and more accurate visual descriptions for these categories. In other words, our approach discovers much more useful linkages to visual patterns for the given category.

4.4 Conclusions

In this chapter, we presented a new framework for domain-robust image dataset construction with web images. Three successive modules were employed in the framework, namely query expanding, noisy expansion filtering and noisy image filtering. To verify the effectiveness of our proposed method, we constructed an image dataset DRID-20. Extensive experiments demonstrated the superiority of our method to several weakly supervised and web-supervised state-of-the-art methods.

Table 4.1: Object detection results (A.P.) (%) on PASCAL VOC 2007 (TEST).

Method	WSL	SPM-VID	WSVCL	IDC-MTM	**Our**	OD-DPM
Supervision	weak	weak	web	web	web	full
airplane	13.4	**17.4**	14.0	14.8	15.5	33.2
bike	**44.0**	-	36.2	38.4	40.6	59.0
bird	3.1	9.3	12.5	**16.5**	16.1	10.3
boat	3.1	9.2	**10.3**	7.4	9.69	15.7
bottle	0.0	-	9.2	12.6	**13.7**	26.6
bus	31.2	-	35.0	39.5	**42.0**	52.0
car	**43.9**	35.7	35.9	38.1	37.9	53.7
cat	7.1	9.4	8.4	8.9	**9.8**	22.5
chair	0.1	-	**10.0**	9.3	9.6	20.2
cow	9.3	9.7	17.5	17.9	**18.4**	24.3
table	9.9	-	6.5	10.2	**10.6**	26.9
dog	1.5	3.3	**12.9**	11.5	11.6	12.6
horse	29.4	16.2	30.6	31.8	**36.1**	56.5
motorcycle	**38.3**	27.3	27.5	29.7	36.9	48.5
person	4.6	-	6.0	7.2	**7.9**	43.3
plant	0.1	-	**1.5**	1.1	1.3	13.4
sheep	0.4	-	18.8	19.5	**20.4**	20.9
sofa	3.8	-	10.3	10.3	**10.8**	35.9
train	**34.2**	15.0	23.5	24.2	27.6	45.2
tv/monitor	0.0	-	16.4	15.6	**18.4**	42.1
average	13.87	15.25	17.15	18.22	**19.74**	33.14

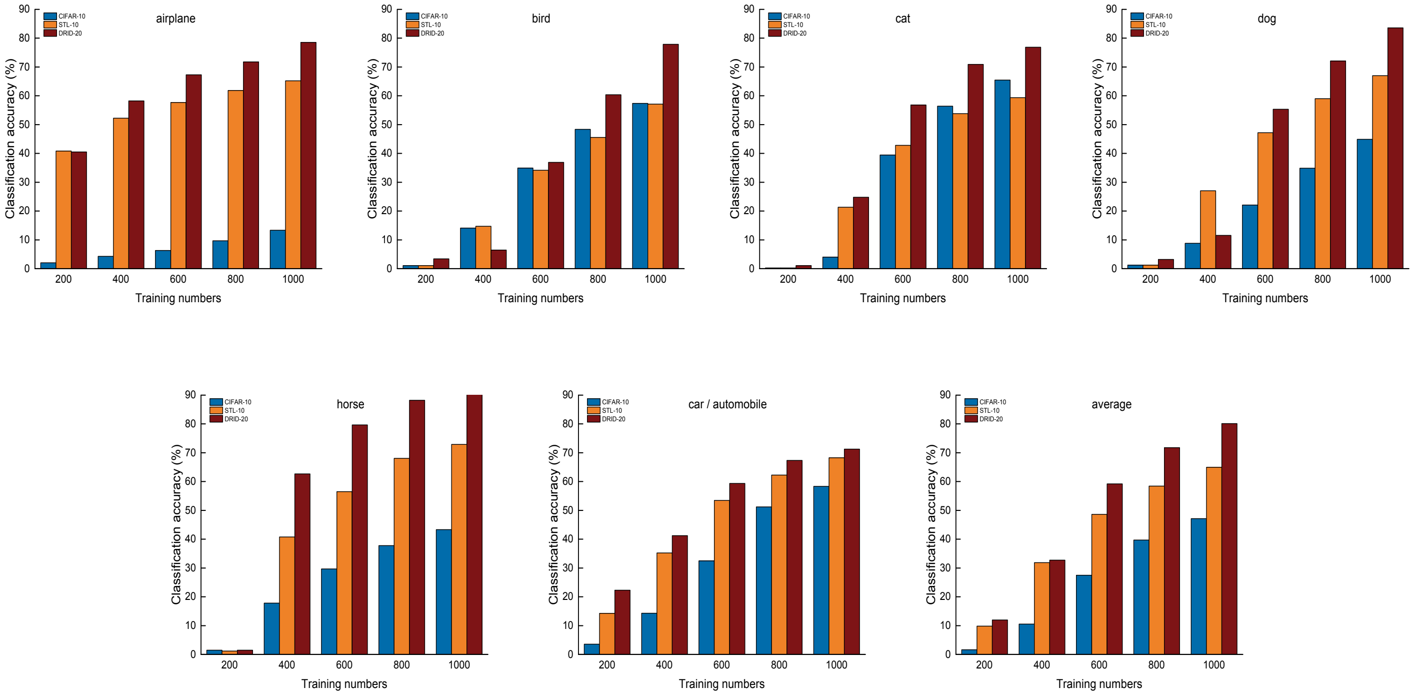

Figure 4.2: Image classification ability of CIFAR-10, STL-10 and DRID-20 on PASCAL VOC 2007 dataset: (a) airplane, (b) bird, (c) cat, (d) dog, (e) horse, (f) car/automobile and (g) average.

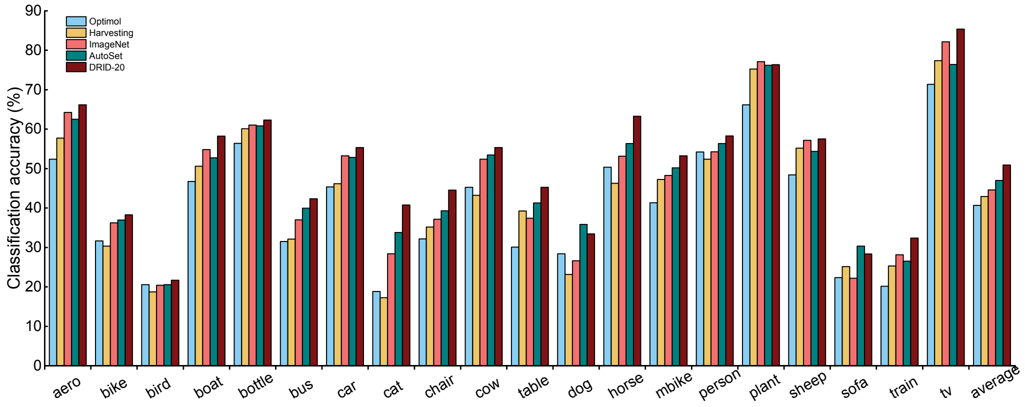

Figure 4.3: Image classification ability of Optimol, Harvesting, ImageNet, AutoSet and DRID-20 on PASCAL VOC 2007 dataset.

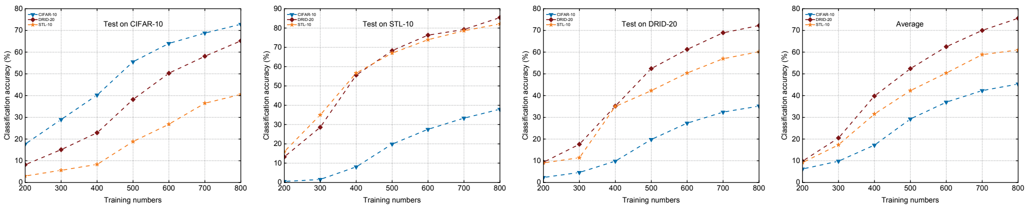

Figure 4.4: Cross-dataset generalization ability of classifiers learned from CIFAR-10, STL-10, DRID-20 and then tested on: (a) CIFAR-10, (b) STL-10, (c) DRID-20, (d) Average.

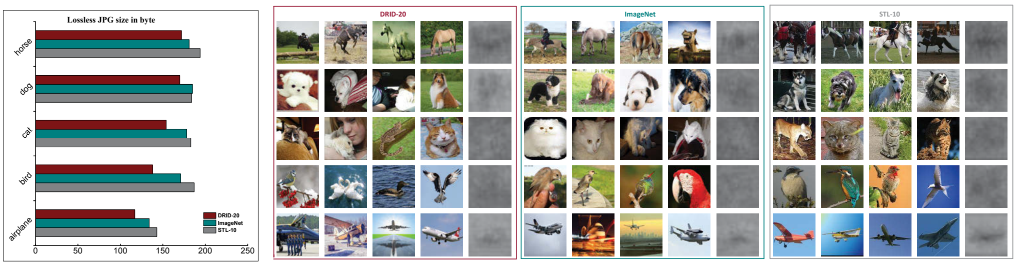

Figure 4.5: (a) Comparison of the lossless JPG file sizes of average images for five different categories in DRID-20, ImageNet and STL-10. (b) Example images from DRID-20, ImageNet, STL-10 and average images for each category indicated by (a).

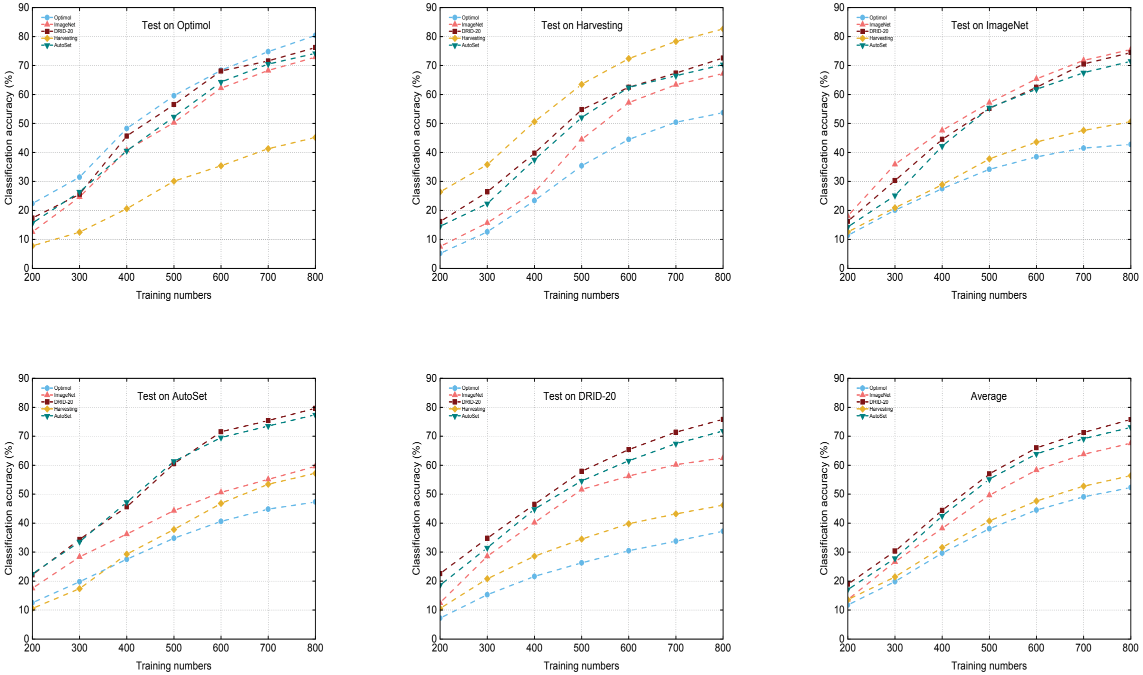

Figure 4.6: Cross-dataset generalization ability of classifiers learned from Optimol, Harvesting, ImageNet, AutoSet, DRID-20 and then tested on: (a) Optimol, (b) Harvesting, (c) ImageNet, (d) AutoSet, (e) DRID-20, (f) Average.

Chapter 5

Accuracy

5.1 Introduction

Directly constructing image dataset with the retrieved images is not practical. It is mainly due to the number of images retrieved from image search engine for each query and the unsatisfactory accuracy of ranking relatively rearward images. In order to improve the overall accuracy, method (Lin et al. 2003) re-ranked images by taking into account of the text contents on the original page from which the images were obtained. The method in (Fergus et al. 2005) involved visual clustering of the images using probabilistic Latent Semantic Analysis (pLSA) (Hofmann 1999) on a visual vocabulary while (Vijayanarasimhan & Grauman 2008) used multiple instance learning and iteratively methods to learn the visual models. Li *et al.* in (Li & Fei-Fei 2010) proposed an incremental learning strategy to learn the visual models. However, all of these methods have a restriction on the total number of images which can be retrieved from the image search engine.

To overcome the restriction of downloading number, method (Berg & Forsyth 2006) and (Schroff et al. 2011) proposed to use web search instead of image search engine to obtain a large pool of candidate images. The method in (Berg & Forsyth 2006) can be mainly divided into two steps: First, train a classifier with manual intervention. Then, the classifier is used to re-

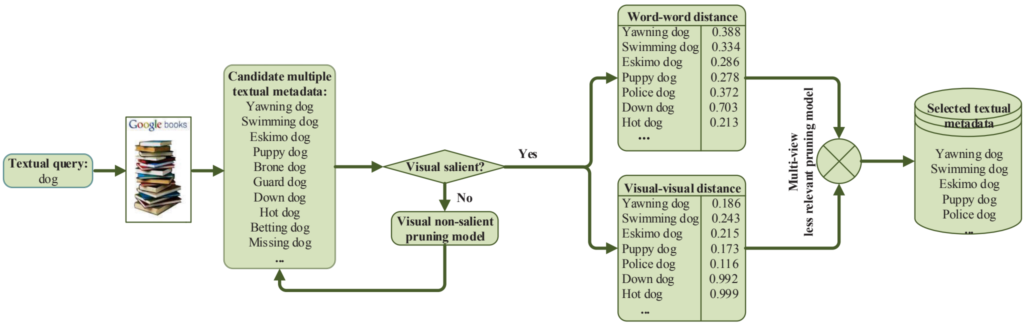

Figure 5.1: Illustration of the process for obtaining multiple textual metadata. The input is a textual query that we would like to find multiple textual metadata for. The output is a set of selected textual metadata which will be used for raw image dataset construction.

rank the retrieved images. The advantages of this method are overcoming the restriction of downloading number, as well as avoiding the problem of polysemy and providing relatively high accuracy images for the given query. However, due to the needs of manual intervention, the cost of this method is high which results in a scale problem. The method in (Schroff et al. 2011) adopt text information to re-rank images retrieved from web search and used these top-ranked images to learn visual models to re-rank images once again. The advantage is eliminating the need for manual intervention. However, the accuracy of image dataset constructed by this method is relatively low. The main reason is the low accuracy of images returned from web search.

In order to leverage the high accuracy as well as overcome the downloading restrictions of image search engine, we propose a novel image dataset constructing framework, through which a large number of highly relevant images are automatically extracted from the web. Specifically, we first discover a set of semantically rich textual metadata, from which the visual non-salient and less relevant textual metadata are removed. The selected textual metadata is used to retrieve sense-specific images to construct the raw image dataset. To suppress the search error and noisy textual metadata (which are not filtered out) induced noisy images, we further divide the retrieved noises into three types and use different methods to filter these noises separately. It should be noted, as we are mainly interested in constructing image datasets for natural image recognition, we would like to remove artificial images from the raw image dataset. To verify the effectiveness of our proposed approach, we construct an image dataset with 100 categories, which we refer to as WSID-100 (web-supervised image dataset 100). Extensive experiments on image classification, cross-dataset generalization, and object detection demonstrate the superiority of our approach.

5.2 Framework and Methods

We are targeting at automatically constructing image dataset in a scalable way while ensuring the accuracy. We automatize the three most labor cost steps. Fig. 5.1 shows the process of multiple textual metadata discovering and noisy textual metadata filtering. Fig. 5.2 demonstrates the process of noisy images filtering. The following subsections describe the details of our proposed framework.

5.2.1 Multiple Textual Metadata Discovering

Images returned from an image search engine tend to have relatively higher accuracy (compared to Flickr and web search), but downloads are restricted to a certain number. In addition, the accuracy of ranking-rearward images is also unsatisfactory. To overcome these restrictions, synonyms are often used to collect more images from image search engine. However, this method only works well for queries which have been defined in an existing ontology (e.g., WordNet (Miller 1995)). Apart from this, images collected by synonyms tend to have the homogenization problem.

Inspired by recent work (Michel et al. 2011), we can use untagged corpora to discover a set of semantically rich textual metadata for modifying the given query. Our motivation is to leverage multiple textual metadata for overcoming the download restriction of image search engine (scalability). We

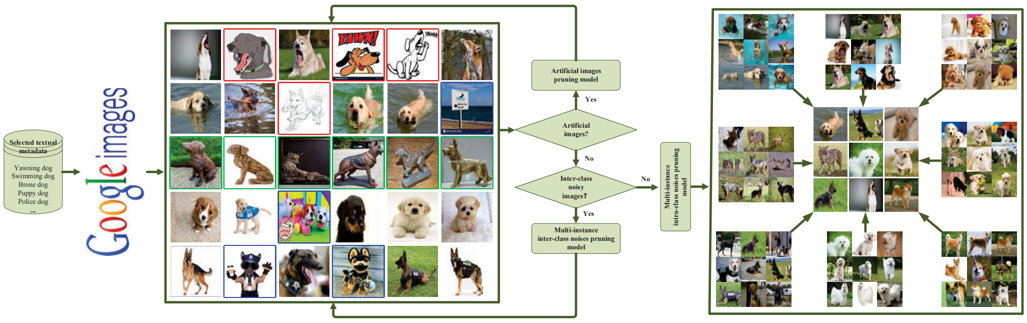
Selected textual metadata
Yawning dog
Swimming dog
Brone dog
Puppy dog
Police dog
...
Google images
Artificial images pruning model
Yes
Artificial images?
No
Inter-class noisy images?
Yes
Multi-instance inter-class noises pruning model
No
Multi-instance intra-class noises pruning model

Figure 5.2: Illustration of the process for obtaining selected images. The input is a set of selected textual metadata. Artificial images, inter-class noisy images, and intra-class noisy images are marked with red, green and blue bounding boxes separately. The output is a group of selected images in which the images corresponding to different textual metadata.

use this semantically rich textual metadata (corresponding images) to reflect the different visual distributions for the given query. The detailed candidate textual metadata discovered in this step can be found on website[1].

5.2.2 Noisy Textual Metadata Filtering

Multiple textual metadata discovering not only brings all the useful data, but also some noises (e.g., "betting dog", "missing dog" and "hot dog" in Fig. 5.3). Using this noisy textual metadata to retrieve images will have a negative effect on the accuracy. To this end, we prune this noisy textual metadata before we collect candidate images for the target query. We divide the noisy textual metadata into two types (visual non-salient and less relevant) and propose to filter these two types of noises separately.

Visual non-salient textual metadata pruning

From the visual consistency perspective, we want to identify visual salient and eliminate non-salient textual metadata in this step (e.g., "betting dog"

[1] http://www.multimediauts.org/dataset/WSID-100.html

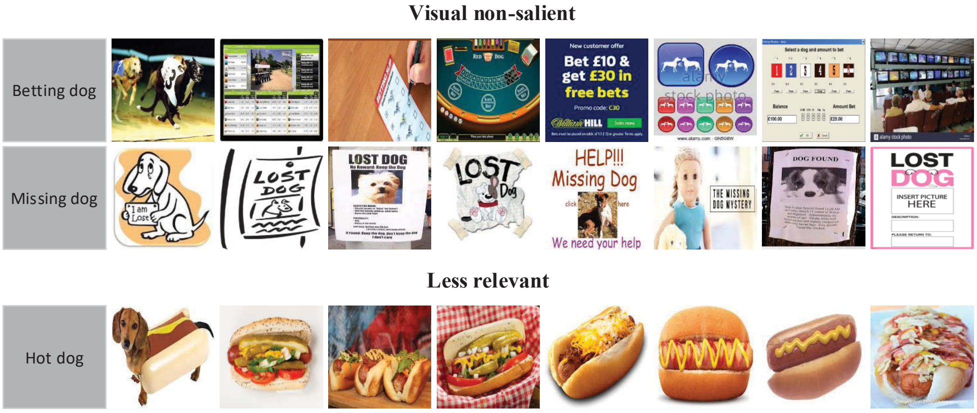

Figure 5.3: A snapshot of the retrieved images for visual non-salient and less relevant textual metadata.

and "missing dog" in Fig. 5.3). The intuition is that visual salient textual metadata should exhibit predictable visual distributions. Hence, we can use the image classifier-based pruning method.

For each textual metadata , we retrieve the top N samples from Google Image Search Engine as positive images; then randomly split them into a training and validation set $I_i = \{I_i^t, I_i^v\}$. A pool of unrelated samples was collected as negative images. Similarly, the negative images were also split into a training and validation set $\overline{I} = \{\overline{I}^t, \overline{I}^v\}$. We extract 4096 dimensional deep features (based on AlexNet (Krizhevsky et al. 2012)) for each image and train a linear support vector machine (SVM) classifier by using I_i^t and $\overline{I}^t$. The validation set $\{I_i^v, \overline{I}^v\}$ were applied to calculate the classification results S_i. When S_i takes a relatively larger value, we think textual metadata i is visually salient.

Less relevant textual metadata pruning

Normalized Google Distance (NGD) (Cilibrasi & Vitanyi 2007) extracts the semantic distance between two terms by using the Google page counts. We denote the semantic distance of all textual metadata by a graph $G_{semantic}$ in which the target query is center y. Other textual metadata x has a score

S_{xy} corresponds to the NGD between term x and y. Semantically relevant textual metadata usually has a smaller semantic distance than less relevant (e.g., "yawning dog", "Eskimo dog" and "police dog" which has 0.388, 0.286 and 0.372 respectively is much smaller than "down dog" which has 0.703).

However, this assumption is not always true from the perspective of visual relevance. For example, "hot dog" has a relatively smaller semantic distance 0.213, but it is not relevant to the target query "dog". Thus, we need to identify both of semantic and visual relevant textual metadata for the target query. Similar to the semantic distance, we denote the visual distance of all textual metadata by graph G_{visual} in which the target query is center y. Other textual metadata x has a score V_{xy} corresponds to the visual distance between term x and y. Similar to the previous step, we obtain the visual distance between target query y and other textual metadata x by the score of the center y node classifier f_y on the xth node retrieved images I_x. The difference lies in the different test images.

By treating word-word (semantic) and visual-visual distance (visual) as features from two different views, we formulate less relevant textual metadata pruning as a multi-view learning problem. Our objective is to find both semantically and visually relevant textual metadata. During training, we model each view with one classifier and jointly learn two classifiers with a regularization term that penalizes the differences between two different classifiers. Two views are reproducing kernel Hilbert spaces $\mathcal{H}_{K^{(1)}}$ and $\mathcal{H}_{K^{(2)}}$. Given l labeled data $(x_1, y_1), ...(x_l, y_l) \in \mathcal{X} \times \{\pm 1\}$ and u unlabeled data $x_{l+1}, ...x_{l+u} \in \mathcal{X}$, we seek to find predictors $f^{(1)*} \in \mathcal{H}_{K^{(1)}}$ and $f^{(2)*} \in \mathcal{H}_{K^{(2)}}$ that minimize the following objective function:

$$\begin{aligned}(f^{(1)*}, f^{(2)*}) = \underset{\substack{f^{(1)} \in \mathcal{H}_{K^{(1)}} \\ f^{(2)} \in \mathcal{H}_{K^{(2)}}}}{\arg\min}\ \mathrm{Loss}(f^{(1)}, f^{(2)}) + \gamma_1 \left\| f^{(1)} \right\|^2_{\mathcal{H}_{K^{(1)}}} \\ + \gamma_2 \left\| f^{(2)} \right\|^2_{\mathcal{H}_{K^{(2)}}} + \lambda \sum_{i=l+1}^{l+u} [f^{(1)}(x_i) - f^{(2)}(x_i)]^2.\end{aligned} \tag{5.1}$$

The first term is loss function and the next two are the regularization terms. The last term is called "co-regularization" which encourages the selection of a pair predictors $(f^{(1)*}, f^{(2)*})$ that agree on the unlabeled data. During testing, we make predictions by averaging the classification results from both of two views and the prediction rule is:

$$\mathcal{J} = \frac{1}{2}(f^{(1)}(x) + f^{(2)}(x)) \tag{5.2}$$

Following (Sindhwani, Niyogi & Belkin 2005, Brefeld, Gärtner, Scheffer & Wrobel 2006), we adopt the form of loss function as:

$$\text{Loss}(f^{(1)}, f^{(2)}) = \frac{1}{2l}\sum_{i=1}^{l}\left(\left[f^{(1)}(x_i) - y_i\right]^2 + \left[f^{(2)}(x_i) - y_i\right]^2\right) \tag{5.3}$$

We give the solution to (5.1) in the Appendix A.1. After we obtain the models for two views, we use (5.2) to prune less relevant textual metadata.

5.2.3 Noisy Images Filtering

The selected textual metadata were used to collect images from image search engine to construct the raw image dataset. Due to the error index of image search engine, some noises may be included (artificial and intra-class noisy images). In addition, a few noisy textual metadata which are not filtered out can also bring some noises (inter-class noisy images). As shown in Fig. 5.2, our process for filtering noisy images consists of three major steps: artificial images pruning, inter-class and intra-class noisy images pruning.

Artificial images pruning

As we are mainly interested in constructing image datasets for natural image recognition, we would like to remove artificial images from the raw image dataset. The artificial images contain "sketches", "drawings", "cartoons", "charts", "comics", "graphs", "plots" and "maps". Since artificial images tend to have only a few colors in large areas or sharp edges in certain orientations, we choose the visual features of color and gradient histogram for separating artificial images from natural images. We train a radial basis function SVM model by using the selected visual features. The artificial images were obtained by retrieving queries: "sketch", "drawings","cartoons", "charts", "comics", "graphs", "plots" and "maps" (250 images for each query, 2000 images in total), natural images were obtained by directly using the images in ImageNet (2000 images in total).

After the pruning model was learned, we apply it to the entire raw image dataset to prune artificial images. The pruning model achieves around 94 percent classification accuracy on artificial images (using two-fold cross-validation) and significantly reduces the number of artificial images in the raw image dataset. There is some loss of the natural images, with, on average, 6 percent removed. Although this seems to be a little high, the accuracy of the resulting dataset is greatly improved.

Inter-class noisy images pruning

Inter-class noisy images were caused by the noisy textual metadata which are not filtered out. As shown in Fig. 5.2 "bronze dog" images, these noises tend to exist in the form of "groups". Hence we proposed to use multi-instance learning (MIL) based method to filter these "group" noisy images. Each selected textual metadata was treated as a "bag" and the images corresponding to the textual metadata were treated as "instances". We formulate inter-class noisy images pruning as a MIL problem. Our objective is to prune group noisy images (corresponding to negative "bags").

We denote the bags as B_i, the positive and negative bags as B_i^+ and B_i^-, respectively. l^+ and l^- denote the numbers of positive and negative bags separately. All instances belong to feature space $\mathbb{Q}$. Bag B_i contains n_i instances x_{ij}, $j = 1, ..., n_i$. For simplicity, we re-index instances as x^k when we line up all instances in all bags together, $k = 1, ..., n$ and $n = \sum_{i=1}^{l+} n_i^+ + \sum_{i=1}^{l-} n_i^-$.

To characterize bags, we take the instance-based feature mapping method proposed in (Chen, Bi & Wang 2006, Maron 1998). Specifically, we assume each bag may consist of more than one target concept and the target concept can be approximated by an instance in the bags. Under this assumption, the most-likely-cause estimator can be written as:

$$\Pr(x^k|B_i) \propto s(x^k, B_i) = \max_j \exp(-\frac{\|x_{ij} - x^k\|}{\sigma^2}), \tag{5.4}$$

where σ is a predefined scaling factor. $s(x^k, B_i)$ can be explained as a similarity between bag B_i and concept x^k. It is determined by the concept and the closest instance in the bag. Then the bag B_i can be embedded with coordinates

$$\mathbf{m}(B_i) = [s(x^1, B_i), s(x^2, B_i), ...s(x^n, B_i)]^\top. \tag{5.5}$$

Given a training set which contains l^+ positive bags and l^- negative bags, we apply the mapping function (5.5) and obtain the following matrix representation of all training bags:

$$\begin{bmatrix} s(x^1, B_1^+) & \cdots & s(x^1, B_{l^-}^-) \\ s(x^2, B_1^+) & \cdots & s(x^2, B_{l^-}^-) \\ \vdots & \ddots & \vdots \\ s(x^n, B_1^+) & \cdots & s(x^n, B_{l^-}^-) \end{bmatrix}. \tag{5.6}$$

Each column corresponds to a bag, and the kth feature realizes the kth row of the matrix. Generally speaking, when x^k achieves a high similarity to some positive bags and low similarity to negative bags, we think that the feature $s(x^k, \cdot)$ induced by x^k provides "useful" information in separating the positive from negative bags.

Instance-based feature mapping tends to has a better generalization ability. The disadvantage is that it may require an expensive computational cost. Our solution is to construct 1-norm SVM classifiers and select important features simultaneously. The motivation is 1-norm SVM can be formulated as a linear programming (LP) problem and the computational cost will not be an issue. The 1-norm SVM is formulated as follows:

$$\begin{aligned} \min_{\mathbf{w},b,\varepsilon,\eta} \quad & \lambda \sum_{k=1}^{n} |w_k| + C_1 \sum_{i=1}^{l^+} \varepsilon_i + C_2 \sum_{j=1}^{l^-} \eta_j \\ \text{s.t.} \quad & (\mathbf{w}^\top \mathbf{m}_i^+ + b) + \varepsilon_i \geqslant 1, i = 1, ..., l^+, \\ & -(\mathbf{w}^\top \mathbf{m}_j^- + b) + \eta_j \geqslant 1, j = 1, ..., l^-, \\ & \varepsilon_i, \eta_j \geqslant 0, i = 1, ..., l^+, j = 1, ..., l^- \end{aligned} \tag{5.7}$$

where $\boldsymbol{\varepsilon}$ and $\boldsymbol{\eta}$ are hinge losses. Choosing different parameters C_1 and C_2 will penalize on false negatives and false positives. We usually let $C_1 = \delta$, $C_2 = 1 - \delta$ and $0 < \delta < 1$ so that the training error is determined by a convex combination of the training errors occurred on positive bags and on negative bags.

To solve the 1-norm SVM (5.7) with linear programming, we rewrite $w_k = u_k - v_k$, where $u_k, v_k \geqslant 0$. Then we can formulate linear programming in variables $\mathbf{u}$, $\mathbf{v}$, b, $\boldsymbol{\varepsilon}$ and $\boldsymbol{\eta}$ as:

$$\begin{aligned} \min_{\mathbf{u},\mathbf{v},b,\boldsymbol{\varepsilon},\boldsymbol{\eta}} \ & \lambda \sum_{k=1}^{n} (u_k + v_k) + \delta \sum_{i=1}^{l^+} \varepsilon_i + (1 - \delta) \sum_{j=1}^{l^-} \eta_j \\ \text{s.t.} \quad & \left[(\mathbf{u} - \mathbf{v})^\top \mathbf{m}_i^+ + b\right] + \varepsilon_i \geqslant 1, i = 1, ..., l^+, \\ & - \left[(\mathbf{u} - \mathbf{v})^\top \mathbf{m}_j^- + b\right] + \eta_j \geqslant 1, j = 1, ..., l^-, \\ & \varepsilon_i, \eta_j \geqslant 0, i = 1, ..., l^+, j = 1, ..., l^- \\ & u_k, v_k \geqslant 0, k = 1, ..., n. \end{aligned} \tag{5.8}$$

The solutions of linear programming (5.8) equivalent to those obtained by the 1-norm SVM (5.7). The reason is that for all $k = 1, ..., n$, any optimal solution to (5.8) has at least one of the two variables u_k and v_k equal to 0.

Suppose $\mathbf{w}^* = \mathbf{u}^* - \mathbf{v}^*$ and b^* are the solutions of (5.8), then the influence of the kth feature on the classifier can be determined by the value of w_k^*. Specifically, we select features $\{s(x^k, \cdot) : k \in \phi\}$ to meet the conditions:

$$\phi = \{k : |w_k^*| > 0\}. \tag{5.9}$$

Algorithm 5.1 The algorithm for learning bag classifier

```
Input:
    Positive bags $B_i^+$ and negative bags $B_i^-$.
For (each bag $B_i = \{x_{ij} : j = 1, ..., n_i\}$)
    for (every instance $x^k$)
        $d \leftarrow \min_j \left\| x_{ij} - x^k \right\|$
        the $k$th element of $\mathbf{m}(B_i)$ is $s(x^k, B^i) = e^{-\frac{d^2}{\sigma^2}}$
    end
End
Solve the linear programming in (5.8)
Output:
    The optimal solutions $\mathbf{w}^*$ and $b^*$, the bag classifier (5.10).
```

Finally, we obtain the classification rule of bag B_i to be positive or negative is:

$$y = \text{sign}\left(\sum_{k\in\phi} w_k^* s(x^k, B_i) + b^*\right). \tag{5.10}$$

The detailed process of learning the bag classifier is described in Algorithm 5.1. We apply the rule (5.10) to classify bags. When the bag is classified to be negative, the group images corresponding to the bag will be filtered out.

Intra-class noisy images pruning

After we prune inter-class noisy images, we then only care the intra-class noises corresponding to the positive bags. Intra-class noises were induced by the error index of image search engine. As shown in Fig. 5.2, these noises usually exist in the form of "individuals".

The basic idea of pruning intra-class noises in positive bags is according to their contributions to the classification of the bag. Instances (corresponding to images) in the bags can be divided into two types: positive class and negative class. An instance is assigned to the positive class when its contribution to $\sum_{k\in\phi} w_k^* s(x^k, B_i)$ is greater than a threshold θ. For instance x_{ij} in

Algorithm 5.2 The algorithm for pruning intra-class noises

```
Input:
    $\phi = \{k : |w_k^*| > 0\}$,
    $\varphi = \left\{j^* : j^* = \arg\min_j \left\|x_{ij} - x^k\right\|, k \in \phi\right\}$.
Initialize $\nu_k = 0$ for every $k$ in $\phi$
For (every $j^*$ in $\varphi$)
    $\phi_{j^*} = \{k : k \in \phi, j^* = \arg\min_j \left\|x_{ij} - x^k\right\|\}$
    for (every $k$ in $\phi_{j^*}$)
        $\nu_k \leftarrow \nu_k + 1$
    end
End
For (every $x_{ij^*}$ with $j^*$ in $\varphi$)
    Compute $g(x_{ij^*})$ using (5.12)
End
Output:
    All positive instances $x_{ij^*}$ satisfying $g(x_{ij^*}) > \theta$
```

bag B_i, we define an index set φ as:

$$\varphi = \left\{ j^* : j^* = \arg\max_j \exp\left(-\frac{\left\|x_{ij} - x^k\right\|^2}{\sigma^2} \right), k \in \phi \right\}. \tag{5.11}$$

Then the bag classification rule (5.10) only needs the instances $x_{ij^*}, j^* \in \varphi$. Removing an instance $x_{ij^*}, j^* \notin \varphi$ from the bag will not affect the value of $\sum_{k \in \phi} w_k^* s(x^k, B_i)$ in (5.10). There may exist more than one instance in bag B_i maximizes $\exp(-\frac{\left\|x_{ij} - x^k\right\|^2}{\sigma^2})$ for a given $x^k, k \in \phi$. We denote the number of maximizers for x^k by ν_k. We then rewrite the bag classification rule (5.10) in terms of the instances indexed by φ as:

$$y = \operatorname{sign}\left(\sum_{j^* \in \varphi} g(x_{ij^*}) + b^* \right),$$

where

$$g(x_{ij^*}) = \sum_{k \in \phi} \frac{w_k^* s(x^k, x_{ij^*})}{\nu_k} \tag{5.12}$$

determines the contribution of x_{ij^*} to the classification of the bag B_i. Instance x_{ij^*} belongs to the positive class if $g(x_{ij^*}) > \theta$. Otherwise, x_{ij^*} belongs to the negative class. The choice of threshold θ is a application specific problem. In our experiments, the parameter θ is chosen to be bag dependent as $-\frac{b^*}{|\varphi|}$. The detailed process of pruning intra-class noises is described in Algorithm 5.2. We apply the rule (5.12) to prune negative instances (corresponding to the intra-class noises).

5.3 Experiments

In this section, we first construct an image dataset with 100 categories and conduct experiments on image classification, cross-dataset generalization, and object detection to verify the effectiveness of our dataset. Then we quantitative analyze the parameter sensitivity of our proposed approach. Finally, we introduce how to use our provided platform for evaluating various algorithms in the task of pruning noisy images.

5.3.1 Image Dataset Construction

We choose all the 20 categories in PASCAL VOC 2007 dataset plus 80 other categories as the target categories to construct our dataset WSID-100. The reason is existing weakly supervised and web-supervised methods were evaluated on this dataset.

For each category, we first discover the multiple textual metadata from Google Books with POS. Then the first $N = 100$ images were retrieved for each discovered textual metadatato represent its visual distribution. In spite of the fact that noises may be contained, we treat the retrieved images as positive samples and split them into a training and validation set $I_i = \{I_i^t = 75, I_i^v = 25\}$. We gather a random pool of negative images and split them into a training and validation set $\overline{I} = \{\overline{I}^t = 25, \overline{I}^v = 25\}$. Through experiments, we declare a textual metadata i to be visual salient when the classification result $S_i \geq 0.6$. We have released the discovered textual metadata for 100 categories and the corresponding images (original image URL) on website[1].

To prune less relevant textual metadata, we calculate the word-word and visual-visual distance between visual salient textual metadata and target query. We label $l_1 = 500$ positive data and $l_2 = 500$ negative data. This labelling is for the textual metadata. We use a total of $l = l_1 + l_2 = 1000$ labeled and $u = 500$ unlabeled data to learn the multi-view prediction rule (5.2). This labeling work only needs to be done once and the prediction rule (5.2) will be used for pruning all less relevant textual metadata.

We construct the raw image dataset by using the textual metadata which are not filtered out. Specifically, we collect the top 100 images for each selected textual metadata. Since not enough textual metadata was found for query "potted plant", we collect the top 500 images for "potted plant" textual metadata. To filter artificial images, we learn a radial basis function SVM model by using the visual feature of color and gradient histogram. Although the color and gradient histogram + SVM framework that we use is not the prevailing state-of-the-art method for image classification, we found our method to be effective and sufficient in pruning artificial images.

By treating each selected textual metadata as a "bag" and the images therein as "instances", we formulate inter-class and intra-class noisy images pruning as a multi-instance learning problem. Our objective is to prune "group" (bag-level) inter-class noisy images and "individual" (instance-level) intra-class noisy images. To learn the bag prediction rule (5.10), we directly use the previously labeled $l_1 = 500$ positive textual metadata and $l_2 = 500$ negative textual metadata corresponding images as the $l^+ = l_1 = 500$ positive bags and $l^- = l_2 = 500$ negative bags. We apply the prediction rule (5.10) to filter "group" inter-class noisy images. The value of $g(x_{ij^*})$ in (5.12) determines the contribution of x_{ij^*} to the classification of the bag B_i. In our experiment, we choose the threshold θ as bag dependent $\theta = -\frac{b^*}{|\varphi|}$. That is we choose positive instance x_{ij^*} satisfying $g(x_{ij^*}) > -\frac{b^*}{|\varphi|}$. The value of b^* and φ can be obtained by solving (5.8) and (5.11), respectively.

5.3.2 Comparison of Image Classification Ability and Cross-dataset Generalization Ability

Experimental setting

For the comparison of image classification ability, we choose PASCAL VOC 2007 (Everingham et al. 2010) as the testing benchmark dataset. The same categories among various datasets are compared. Specifically, we randomly select 500 images for each category from various datasets as the positive training samples. 1000 unrelated images are chosen as the negative samples to train SVM classification models.

We test the classification ability of these models on PASCAL VOC 2007 dataset. The experiments are repeated for ten times and the average classification ability is taken as the final performance for various datasets. The experimental results are shown in Fig. 5.4 and Table 5.1.

For cross-dataset generalization ability comparison, we randomly select 200 images per category from various datasets as the testing data. [200,300,400, 500,600,700,800] images for each category from various datasets are sequentially chosen as the positive training samples. Similar to the comparison of image classification ability, we use the same 1000 unrelated images as the negative training samples to learn image classification models. Training and testing data for each category has no duplicates. Since dataset STL-10 (Coates et al. 2011) and CIFAR-10 (Krizhevsky & Hinton 2009) have only 6 same categories “airplane”, “bird”, “cat”, “dog”, “horse” and “car/automobile” with other datasets, they won’t be compared with our dataset and other datasets in this experiment. For other datasets, we compare all the 20 same categories. The average classification accuracy on all categories illustrates the cross-dataset generalization ability of one dataset on another dataset (Deng et al. 2009). The experimental results are shown in Fig. 5.5.

For image classification and cross-dataset generalization ability comparison, we set the same options to learn classification models for all datasets. Specifically, we train SVM classifiers by setting the kernel as a radial basis function. The other settings use the default of LIBSVM (Chang & Lin 2011). For all images, we extract the 4096 dimensional deep features based on AlexNet (Krizhevsky et al. 2012).

Baselines

We compare our dataset with two sets of baselines:

- Manually labeled datasets. This set of baselines consists of STL-10 (Coates et al. 2011), CIFAR-10 (Krizhevsky & Hinton 2009) and ImageNet. STL-10 contains ten categories in which per category has 500 training and 800 testing images. Both of training and testing images are used to represent this dataset. CIFAR-10 includes 10 categories and each category contains 6000 images. ImageNet provides an average of 1000 images to represent each category and is organized according to the WordNet hierarchy.
- Web-supervised datasets. This set of baselines consists of DRID-20 (Yao et al. 2017), Optimol (Li & Fei-Fei 2010) and Harvesting (Schroff et al. 2011). DRID-20 contains 20 categories and each category has 1000 images. For Optimol (Li & Fei-Fei 2010), we select all the categories in PASCAL VOC 2007 as the target categories and collect 1000 images for each category by taking the incremental learning mechanism. For Harvesting (Schroff et al. 2011), we first retrieve the possible images from Google web search engine and rank the retrieved images through the text information. The top-ranked images are then leveraged to learn classification models to re-rank the images once again. In total, we construct 20 same categories as PASCAL VOC 2007 for Harvesting dataset.

Experimental results

Cross-dataset generalization ability and image classification ability on third-party testing dataset measure the performance of classifiers learned from one dataset and tested on another dataset. It indicates the robustness of the dataset (Torralba & Efros 2011).

Table 5.1: The average accuracy (%) comparison over 14 and 6 common categories on the PASCAL VOC 2007 dataset.

Method	**PASCAL VOC 2007**	
	14 categories	6 categories
STL-10	-	39.75
CIFAR-10	-	19.04
ImageNet	48.95	41.02
Optimol	42.69	35.97
Harvesting	46.33	34.89
DRID-20	51.13	46.04
Ours	**53.88**	**49.48**

From Fig. 5.4, we observe that the categories “plant”, “tv” and “airplane” present a relatively higher classification accuracy than other categories when using the same number of training images. One possible explanation is that the “diversity” of “plant”, “tv” and “airplane” are simpler than other categories. The images are densely distributed in the feature space. For categories “plant”, “tv” and “airplane”, training and testing images overlaps much more easily.

According to the average accuracy over 6 common categories on the PASCAL VOC 2007 dataset in Table 5.1, the performance of CIFAR-10 is much lower than other datasets. The explanation is that CIFAR-10 has a limited diversity and a serious dataset bias problem (Torralba & Efros 2011). In CIFAR-10, the objects are pure and located in the middle of the images. However, in the testing dataset and other compared datasets, these images not only consist of target objects, but also plenty of other scenarios and objects.

By observing Fig. 5.4, Fig. 5.5 and Table 5.1, our dataset outperforms the web-supervised and manually labeled datasets in terms of image classification ability and cross-dataset generalization ability. Compared with

STL-10, CIFAR-10, ImageNet, Optimol and Harvesting, our dataset which was constructed by multiple textual metadata has a better diversity and can well adapt to the third-party testing dataset. Compared with DRID-20, our method treats textual and visual relevance as features from two different views and takes multi-view based method to leverage both of textual and visual distance for pruning less relevant textual metadata. Our method can be more effective in pruning textual metadata, and then obtain a more accurate dataset. At the same time, we convert the inter-class and intra-class noises pruning into solving a linear programming problem, not only improves the accuracy but also the efficiency.

5.3.3 Comparison of Object Detection Ability

Due to the success of DPM (Felzenszwalb et al. 2010) detector, training detection models without bounding boxes has received renewed attention. Since recently state-of-the-art web-supervised and weakly supervised methods have been evaluated on PASCAL VOC 2007 dataset, we also test the object detection ability of our collected data on this dataset.

Experimental setting

We firstly remove images which have extreme aspect ratios (> 2.5 or < 0.4) and resize images to a maximum of 500 pixels. Then we train a separate DPM for each selected textual metadata to constrain the visual variance. Specifically, we initialize our bounding box with a sub-image in the process of latent re-clustering to avoid getting stuck to the image boundary. Following (Felzenszwalb et al. 2010), we take the aspect-ratio heuristic method to initialize our components. Some components across different textual metadata detectors share visual similar patterns (e.g., “police dog” and “guard dog”). We take the method proposed in (Divvala et al. 2014) to merge visual similar and select representative components. After we obtain the representative components, we leverage the approach proposed in (Felzenszwalb et al. 2010) to augment and subsequently generate the final detector.

Baselines

Three sets of baselines are chosen to compare with our collected data:

• Weakly supervised methods. This set of baselines consists of (Siva & Xiang 2011) and (Prest et al. 2012). Method (Siva & Xiang 2011) leverages image-level labels for training and initializes from objectness. Method (Prest et al. 2012) takes manually labeled videos without bounding box for training and presents the results in 10 out of 20 categories.

• Web-supervised methods. The web-supervised method (Divvala et al. 2014) leverages web information as a supervisor to train DPM detector.

• Fully supervised method. The fully supervised method (Felzenszwalb et al. 2010) is a possible upper bound for weakly supervised and web-supervised methods.

Experimental results

Table 5.4 presents the object detection results of our collected data and other state-of-the-art methods on the PASCAL VOC 2007 test set. From Table 5.4, we have the following observations:

Compared with method (Siva & Xiang 2011) and (Prest et al. 2012) which leverages weak supervision and (Felzenszwalb et al. 2010) which requires full supervision, our method and (Divvala et al. 2014) don't need to label the training data. Nonetheless, our method and (Divvala et al. 2014) achieve better detection results than previously best weakly supervised methods (Siva & Xiang 2011) and (Prest et al. 2012). One possible explanation is our approach as well as (Divvala et al. 2014) takes multiple textual metadata for images collection, the accuracy of training data collected by (Divvala et al. 2014) and our method is much higher than (Siva & Xiang 2011) and (Prest et al. 2012). The training data collected by our approach and (Divvala et al. 2014) contains more effective visual patterns.

Compared to method (Divvala et al. 2014) which also leverages multiple textual metadata for images collection and web supervision, our method achieves the best results in most cases. Possibly because we take different

Table 5.2: The average recall and precision for ten categories corresponding to different S_i.

S_i	0.8	0.7	0.6	0.5	0.4	0.3
Recall	35.6%	72.3%	97.4%	98.7%	100%	100%
Precision	87.2%	78.8%	71.2%	52.7%	46.4%	39.6%

Table 5.3: The average accuracy of inter-class noisy images filtering for ten categories corresponding to different δ.

δ	10^{-3}	10^{-2}	10^{-1}	10^{0}	10^{1}	10^{2}
Accuracy	96.2%	97.5%	96.6%	98.2%	97.6%	98.5%

methods to filter noisy textual metadata and images. Method (Divvala et al. 2014) takes iterative approaches during the process of noisy textual metadata and images removing while our method leverages a multi-view based method for noisy textual metadata removing and multi-instance learning-based method for noisy images removing. Our method can obtain a better diversity of the selected images in the condition of ensuring the accuracy. Our method discovers much richer as well as more useful linkages to visual descriptions for the target category.

5.3.4 Parameter Sensitivity Analysis

There are lots of parameters in the process of our experiments, we mainly analyze two parameters S_i and δ in our proposed framework ($C_1 = \delta$, $C_2 = 1-\delta$ and $0 < \delta < 1$). To analyze parameter S_i and δ, we choose 10 categories and manually label 50 textual metadata for each category. For each textual metadata, we retrieve the top 100 images from image search engine to represent the visual distribution. The value of S_i is selected from the set of {0.3, 0.4, 0.5, 0.6, 0.7, 0.8} by applying the 3-fold cross-validation method. Table 5.2 demonstrates the average recall and precision for 10 categories corresponding to different S_i. Finally, we choose the value of S_i to be 0.6. The reason is we want to get a relatively higher recall while ensuring an acceptable precision.

For the parameter δ, the value is selected from $\{10^{-3}, 10^{-2}, ..., 10^{2}\}$. We also use the 3-fold cross-validation to select the value of δ. Table 5.3 shows the average accuracy of inter-class noisy images filtering. By observing Table 5.3, we found our method is robust to the parameter δ when it is varied in a certain range.

5.3.5 Platform Introduction

Due to the cost of manual labeling is too high, crawling data from the Internet and using the web data (without manual annotation) to train models for various computer vision tasks have attracted broad attention. However, due to the complex of the Internet, the crawled data tend to have noise. Removing noise and choosing high-quality instances for training often plays a key role in the quality of the last trained model. To this end, we provide a benchmark platform for evaluating the performance of various algorithms in the task of pruning noise. The specific steps are as follows:

step 1: obtaining the raw image data for 100 categories from our website[1];

step 2: performing algorithms to prune noise and select useful data from the raw image data;

step 3: running cross-dataset generalization experiments on the selected data and our publicly released dataset WSID-100.

Algorithms which have a better cross-dataset generalization ability tend to have a better ability in the task of pruning noise and selecting high-quality data.

5.4 Conclusions

In this chapter, we presented an automatic image dataset construction framework. To verify the effectiveness of the proposed framework, we built an image dataset with 100 categories. Extensive experiments have shown the superiority of our dataset over manually labeled datasets STL-10, CIFAR-10, ImageNet and web-supervised datasets Harvesting, Optimol and DRID-20 on image classification and cross-dataset generalization. In addition, we successfully applied our data to improve the object detection performance on the PASCAL VOC 2007 dataset. We have publicly released our web-supervised image dataset on website to facilitate the research in the web-vision and other related fields.

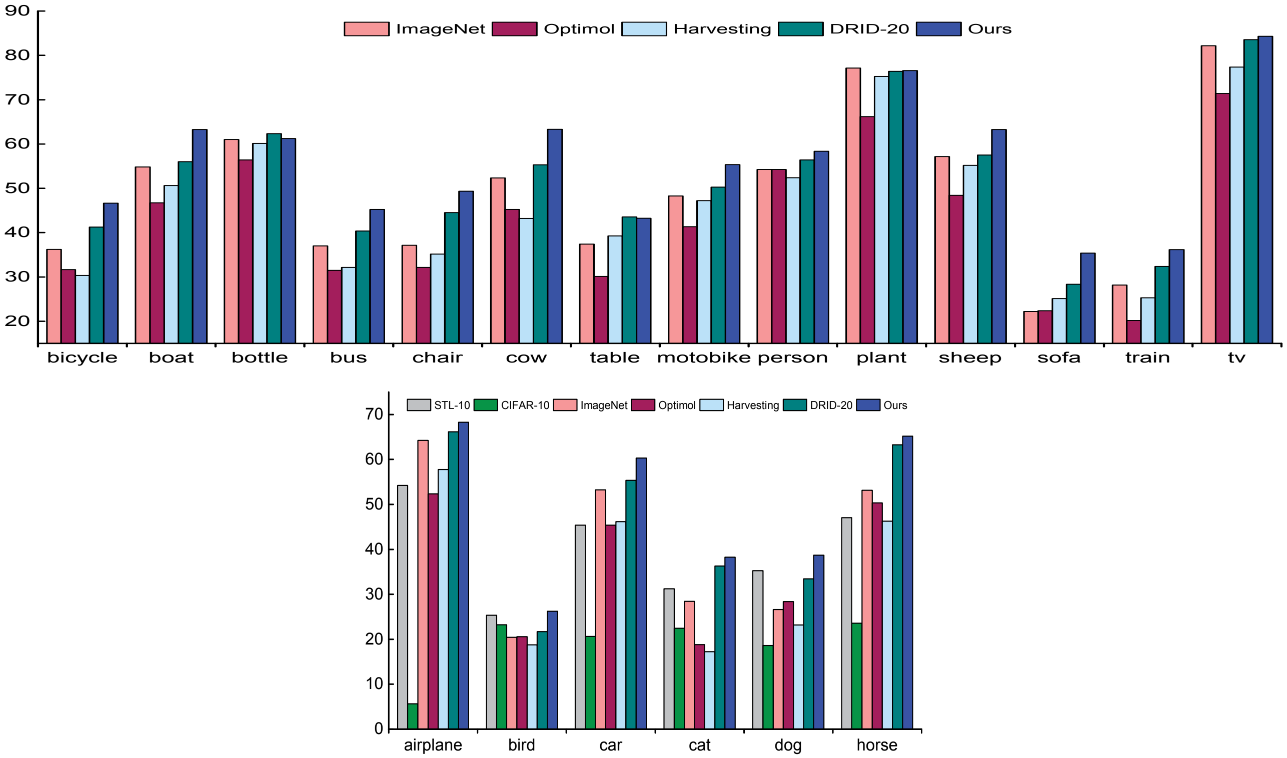

Figure 5.4: The image classification accuracy (%) comparison over 14 and 6 categories on the PASCAL VOC 2007 dataset.

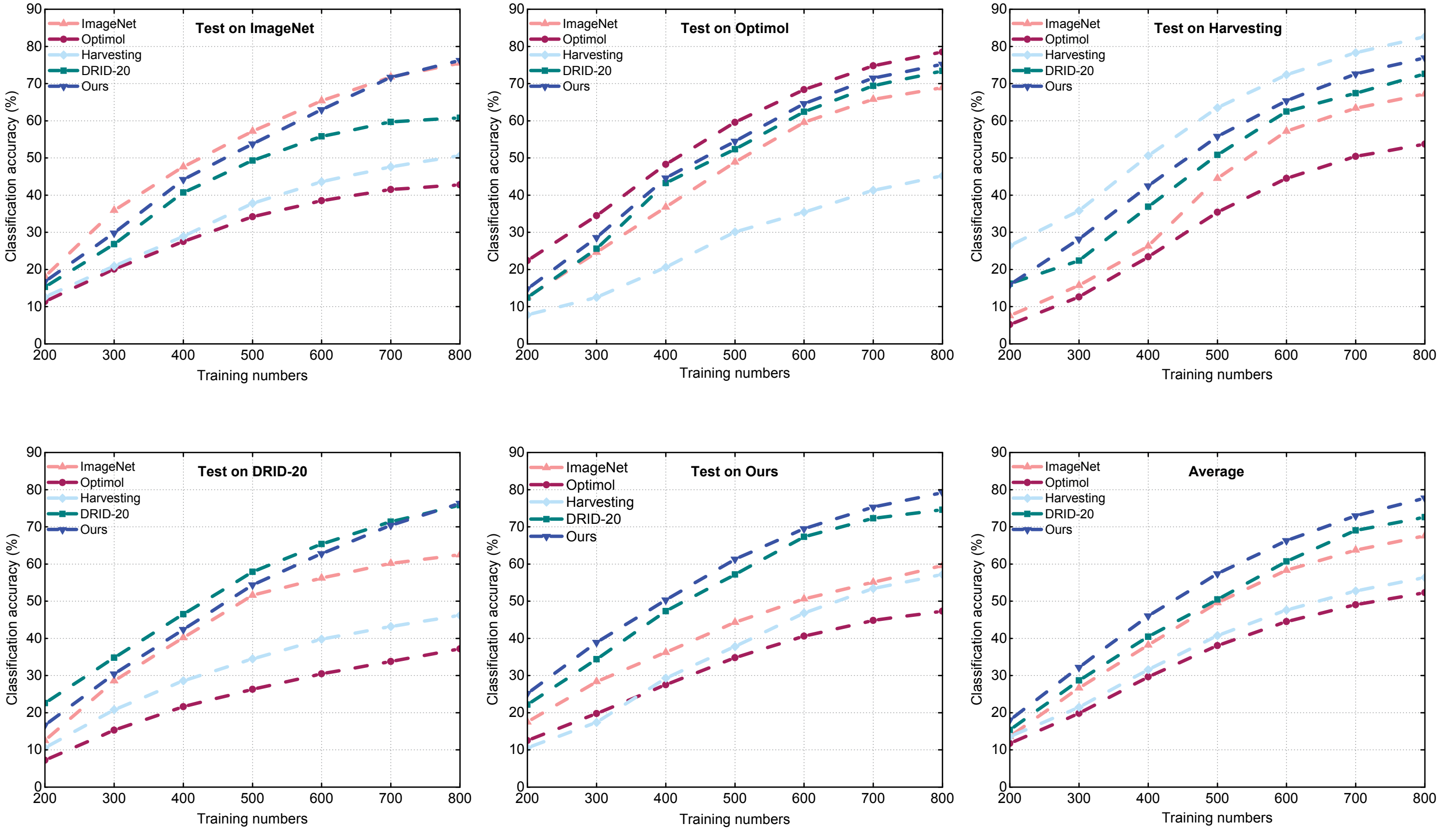

Figure 5.5: The cross-dataset generalization ability of various datasets by using a varying number of training images, and tested on (a) ImageNet, (b) Optimol, (c) Harvesting, (d) DRID-20, (e) Ours, (f) Average.

Table 5.4: Object detection results (A.P.) (%) on PASCAL VOC 2007 dataset (Test).

Method	(Siva & Xiang 2011)	(Prest et al. 2012)	(Divvala et al. 2014)	Ours	(Felzenszwalb et al. 2010)
Supervision	weak	weak	web	web	full
airplane	13.4	17.4	14.0	**17.8**	33.2
bike	**44.0**	-	36.2	42.4	59.0
bird	3.1	9.3	12.5	**17.7**	10.3
boat	3.1	9.2	**10.3**	9.8	15.7
bottle	0.0	-	9.2	**16.2**	26.6
bus	31.2	-	35.0	**44.6**	52.0
car	**43.9**	35.7	35.9	39.7	53.7
cat	7.1	9.4	8.4	**11.2**	22.5
chair	0.1	-	**10.0**	9.4	20.2
cow	9.3	9.7	17.5	**19.8**	24.3
table	9.9	-	6.5	**12.3**	26.9
dog	1.5	3.3	**12.9**	12.4	12.6
horse	29.4	16.2	30.6	**39.5**	56.5
motorcycle	**38.3**	27.3	27.5	36.3	48.5
person	4.6	-	6.0	**8.2**	43.3
plant	0.1	-	**1.5**	1.2	13.4
sheep	0.4	-	18.8	**23.7**	20.9
sofa	3.8	-	10.3	**12.6**	35.9
train	**34.2**	15.0	23.5	31.5	45.2
tv/monitor	0.0	-	16.4	**20.2**	42.1
average	13.87	15.25	17.15	**21.32**	33.14

Chapter 6

Privileged Information

6.1 Introduction

Data-driven classifier learning approaches become very brittle and prone to over-fitting when the training data is inadequate either in quantity or quality. Unfortunately, this is often the case in many real-world applications. A natural solution to alleviate this limitation is incorporating additional privileged information (Wang & Ji 2015, Li et al. 2014, Niu et al. 2017, Divvala et al. 2014). For example, in object recognition, in addition to the image features and labels (e.g., , "horse"), the learner may also leverage object attributes (e.g., , "walking" and "jumping") in the training process. In human action recognition, besides the RGB features and human action labels, human joint positions can be incorporated into the classifier training. In practice, the privileged information can be tags, properties, attributes, positions or the context of the web images.

However, learning classifier with privileged information is a challenging problem. The difficulty lies in three aspects. Firstly, the process of manually labeling privileged information is very expensive. Secondly, it is only available during training and unseen during testing. We cannot combine the privileged information with input features to predict the category label. Thirdly, learning classifiers with PI overly depends on the quality of the collected PI.

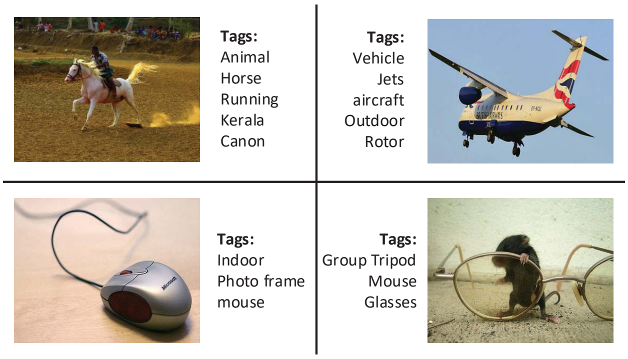

Figure 6.1: Examples of textual tags (privileged information) for images on image sharing website "Flickr". Both of useful and noisy tags are included.

As shown in Fig 6.1, images on website Flickr[1] tend to have multiple textual tags. These textual tags are often associated with noise in practice. If we failed to remove noise, the accuracy and robustness of the learned classifier would be greatly reduced, and, in extreme cases, may become even worse.

Motivated by that, we seek to extract and leverage useful privileged information to enhance classifier learning. Different from previous works which discover privileged information from manually labeled descriptions, our approach extracts the privileged information from untagged corpora. The motivation is to eliminate the dependency on manually labeled data and obtain a relatively more accurate and richer privileged information. Besides, different from previous works which usually encode privileged information into the parameters of the classifier during training, we focus on encoding privileged information into the structure of the classifier during training.

In our work, there are two tags for the image that we collected in the previous chapter: one for the category tag and one for the semantic refinement subcategory tag. So we can use the semantic refinement subcategory tags as PI information to enhance our classifier learning.

[1] https://www.flickr.com/

6.2 Framework and Methods

We treat each selected privileged information as a subcategory for the target category. Suppose we obtain M subcategories in the previous step, we retrieve the top few candidate training images from image search engine for each subcategory.

6.2.1 Formulation

Since the retrieved training images may contain noise, we need to select appropriate samples to train robust classifiers (Fergus et al. 2004, Berg & Forsyth 2006). To this end, a binary indicator $h_i \in \{0, 1\}$ is used to indicate whether or not training instance x_i is selected. To be exact, $h_i = 1$ when x_i is selected, and $h_i = 0$ otherwise. Due to the precision of images returned from the image search engine tends to have a relatively high accuracy, we define each positive subcategory as at least having a portion of η positive images. The value of η can be estimated from some prior knowledge (Li et al. 2011, Yao, Hua, Shen, Zhang & Tang 2016). We define $\mathbf{h} = [h_1, ...h_N]^\top$ as the indicator vector, and use $\mathrm{H} = \{\mathbf{h} | \sum_{i \in I_m} h_i = \eta \left| G_m \right|, \forall m\}$ to represent the feasible set of $\mathbf{h}$, where I_m represents the set of instance indices in G_m, and $|G_m|$ denotes the cardinality of G_m.

We assume there are N retrieved web images coming from C categories and belonging to S subcategories. $z_{i,s} \in \{0, 1\}$ is a binary indicator variable and takes the value of 1 when x_i belongs to the s-th subcategory, and 0 otherwise. We denote $N_s = \sum_{i=1}^{N} z_{i,s}$ as the number of web training images from the s-th subcategory. By treating each subcategory as a “bag” and the retrieved images therein as “instances”, we formulate noisy images removing and integrated classifiers learning as an instance-level MIL problem:

$$\min_{\mathbf{h}, \mathbf{w}_{c,s}, \xi_m} \quad \frac{1}{2} \sum_{c=1}^{C} \sum_{s=1}^{S} \left\| \mathbf{w}_{c,s} \right\|^2 + C_1 \sum_{m=1}^{M} \xi_m \tag{6.1}$$

$$\begin{aligned} \text{s.t. } \frac{1}{|G_m|} \sum_{i \in I_m} h_i(\sum_{s=1}^{S} P_{i,s} (\mathbf{w}_{Y_m,s})^\top \phi(x_i) - \\ (\mathbf{w}_{\hat{c},\hat{s}})^\top \phi(x_i)) \geqslant \eta - \xi_m, \forall m, \hat{s}, \hat{c} \neq Y_m \\ \xi_m \geqslant 0, \forall m \end{aligned} \tag{6.2}$$

where C_1 is a trade-off parameter, ξ_m are slack variables and $\phi(\cdot)$ is the feature mapping function. $P_{i,s}$ is the probability that the i-th training sample comes from the s-th subcategories. It can be obtained by calculating $P_{i,s} = (z_{i,s}/N_s)/\sum_{s=1}^{S}(z_{i,s}/N_s)$. The explanation for constraint (6.2) is that we force the total decision value of each bag obtained based on the classifier corresponding to its own category to be larger than those obtained by using the classifiers for the other categories. The motivation is we want to reduce the bag-level loss by removing noise and identifying the good instances within the training bags.

Since the visual distributions of the training samples from same category or subcategory are generally more similar than different categories and subcategories, we train one classifier for each category and each subcategory. In general, a total of $C \times S$ classifiers $f_{c,s}(x)|c = 1, ...C, s = 1, ...S$ will be learned. For better representation, we omit the bias term and use

$$f_{c,s}(x) = (\mathbf{w}_{c,s})^\top \o(x) \tag{6.3}$$

representing the classifier of the s-th subcategory and the c-th category. The decision function for category C is obtained by integrating the learned classifiers from multiple subcategories:

$$f_c(x_i) = \sum_{s=1}^{S} P_{i,s} f_{c,s}(x_i). \tag{6.4}$$

During testing, we want to find the labels of the most matched subcategory and category, whose classifier achieves the largest decision value from all the subcategories and categories respectively. Thus, the subcategory label of image $\mathbf{x}$ can be predicted by:

$$\arg\max_{s} \mathbf{w}_{c,s}^\top \phi(\mathbf{x}) \tag{6.5}$$

and the category label by:

$$\arg\max_{c}(\max_{s}\mathbf{w}_{c,s}^{\top}\phi(\mathbf{x})). \tag{6.6}$$

6.2.2 Optimization

Problem (6.1) is a non-convex mixed integer problem and is hard to solve directly. However, the dual form of (6.1) can be relaxed as a multiple kernel learning (MKL) problem (Bach et al. 2004) which is much easier to solve. The dual form of (6.1) is:

$$\begin{aligned} \min_{\mathbf{h}}\max_{\boldsymbol{\alpha}} & -\frac{1}{2}\boldsymbol{\alpha}^{\top}\mathbf{Q}^{\mathbf{h}}\boldsymbol{\alpha}+\boldsymbol{\zeta}^{\top}\boldsymbol{\alpha} \\ \text{s.t.} & \sum_{c,s}\alpha_{m,c,s}=C_1,\ \forall m, \\ & \alpha_{m,c,s}\geqslant 0, \quad \forall m,c,s. \end{aligned} \tag{6.7}$$

$\boldsymbol{\alpha}\in\mathbb{R}^{D}$ ($D=M\cdot C\cdot S$) is a vector containing dual variables $\alpha_{m,c,s}$. $\boldsymbol{\zeta}\in\mathbb{R}^{D}$ is a vector, in which $\zeta_{m,c,s}=0$ if $c=Y_m$ and $\zeta_{m,c,s}=\eta$ otherwise. Each element in matrix $\mathbf{Q}^{\mathbf{h}}\in\mathbb{R}^{D\times D}$ can be calculated through: $\mathbf{Q}^{\mathbf{h}}=(1/\left|G_m\right|\left|G_{\hat{m}}\right|)\sum_{i\in I_m}\sum_{j\in I_{\hat{m}}}h_ih_j\o(\mathbf{x}_i)^{\top}\o(\mathbf{x}_j)\lambda(i,j,c,\hat{c},s,\hat{s})$.

Problem (6.7) is a mixed integer programming problem and is hard to directly optimize the indicator vector $\mathbf{h}$. Inspired by recent works (Li et al. 2011, Li, Kwok, Tsang & Zhou 2009), we can find the coefficients of $\mathbf{h}_t\mathbf{h}_t^{\top}$. For consistent presentation, we denote $\mathbf{d}=[d_1,...d_T]^{\top}$, $T=|\mathrm{H}|$, and the feasible set of $\boldsymbol{\alpha}$, $\mathbf{d}$ as ν and $D=\{\mathbf{d}|\mathbf{d}^{\top}\mathbf{1}=1,\mathbf{d}\geqslant 0\}$, respectively. Then we can get the following optimization problem:

$$\min_{\mathbf{d}\in D}\max_{\boldsymbol{\alpha}\in\nu} \quad -\frac{1}{2}\sum_{t=1}^{T}d_t\boldsymbol{\alpha}^{\top}\mathbf{Q}^{\mathbf{h}_t}\boldsymbol{\alpha}+\boldsymbol{\zeta}^{\top}\boldsymbol{\alpha}. \tag{6.8}$$

When we set the base kernel as $\mathbf{Q}^{\mathbf{h}_t}$, the above problem is similar to the MKL dual form and we are able to solve it on its primal form, which is a convex optimization problem:

$$\min_{\mathbf{d\in D,w_t,\xi_m}} \frac{1}{2}\sum_{t=1}^{T}\frac{\left\|\mathrm{w}_t\right\|^2}{d_t}+C_1\sum_{m=1}^{M}\xi_m \tag{6.9}$$

$$\text{s.t.} \sum_{t=1}^{T} \mathbf{w}_t^\top \varphi(\mathbf{h}_t, G_m, c, s) \geqslant \zeta_{m,c,s} - \xi_m, \ \forall m, c, s \tag{6.10}$$

where $\varphi(\mathbf{h}_t, G_m, c, s)$ is the feature mapping function induced by $\mathbf{Q}^{\mathbf{h}_t}$. We solve the convex problem in (6.9) by updating $\mathbf{d}$ and $\{\mathbf{w_t}, \xi_m\}$ in an alternative way.

Update $\mathbf{d}$: We firstly fix $\{\mathbf{w_t}, \xi_m\}$ to solve $\mathbf{d}$. By introducing a dual variable β for constraint $\mathbf{d}^\top \mathbf{1} = 1$, the Lagrangian form of (6.9) can be derived as:

$$\begin{aligned} \pounds = \frac{1}{2}\sum_{t=1}^{T} \frac{\|\mathrm{w}_t\|^2}{d_t} + C_1 \sum_{m=1}^{M} \xi_m - \sum_{m,c,s} \alpha_{m,c,s} \\ (\sum_{t=1}^{T} \mathbf{w}_t^\top \varphi(\mathbf{h}_t, G_m, c, s) - \zeta_{m,c,s} + \xi_m) + \beta(\sum_{t=1}^{T} d_t - 1). \end{aligned} \tag{6.11}$$

Through set the derivative of (6.11) with respect to d_t as zero, we can get:

$$d_t = \frac{\|\mathbf{w}_t\|}{\sqrt{2\beta}}, \ \forall t = 1, ..., T. \tag{6.12}$$

For parameter β, $\|\mathbf{w}_t\|/\sqrt{2\beta}$ is monotonically decreasing. In addition, parameter d_t satisfy $\sum_{t=1}^{T} d_t = 1$. Therefore, we can use binary search method to solve β and recover d_t according to (6.12).

Update $\mathbf{w}_t$: When $\mathbf{d}$ is fixed, $\mathbf{w}_t$ can be obtained by solving $\boldsymbol{\alpha}$ in (6.8). Problem (6.8) is a quadratic programming problem w.r.t $\boldsymbol{\alpha}$. Since there are $M \cdot C \cdot S$ variables in our problem, it is time-consuming to employ the existing quadratic programming solvers. Inspired by recent work (Li, Tsang, Kwok & Zhou 2009), we can employ the cutting-plane algorithm (Kelley 1960) to solve this quadratic programming problem.

We start from a small number of base kernels and at each iteration we add a new violating base kernel. Therefore, only a small set of $\mathbf{h}$ need to be solved at each iteration and the whole problem can be optimized more effectively. By setting the derivatives of (6.11) with respect to $\{\mathbf{w}_t, \xi_t, d_t\}$ as

Algorithm 6.1 Cutting-plane algorithm for solving the proposed instance-level MIL model.

```
Input:
    Image bags $\{(G_m, Y_m)|_{m=1}^{M}\}$.
 1: Initialize $y_i = 1$ for all $x_i$ in selected bags $G_m$.
 2: Set $t = 1$ and $C = \{\mathbf{h}_1\}$;
 3: Repeat
 4:   $t = t + 1$;
 5:   Solve MKL to obtain $(\mathbf{d}, \boldsymbol{\alpha})$ in (6.8) based on $C$;
 6:   //Find the most violating $\mathbf{h}_t$
 7:   for each bag $G_m$
 8:     Fix the labelling of instances in all other bags;
 9:     Enumerate the candidates of $y_i$ in $G_m$;
10:     Find the optimal $\mathbf{y}_m$ by maximizing (6.15);
11:   end
12:   repeat lines 7-11 until there is no change in $\mathbf{h}$;
13:   Add the most violating $\mathbf{h}_t$ to the violation set $C =$
14:   $C \cup \mathbf{h}_t$;
15: Until The objective of (6.8) converges.
Output:
    The learnt image classifier $f(\mathrm{x})$.
```

zeros, (6.8) can be rewritten as:

$$\begin{aligned} &\max_{\beta, \boldsymbol{\alpha} \in \nu} -\beta + \boldsymbol{\zeta}^\top \boldsymbol{\alpha} \\ &\text{s.t. } \frac{1}{2} \boldsymbol{\alpha}^\top \mathbf{Q}^{\mathbf{h}_t} \boldsymbol{\alpha} \leqslant \beta, \ \forall t. \end{aligned} \tag{6.13}$$

We solve (6.13) by solving α with only one constraint at the first, then add a new violating constraint iteratively. Particularly, since each constraint is associated with an $\mathbf{h}_t$, we can obtain the most violated constraint by optimizing:

$$\max_{\mathbf{h}} \frac{1}{2} \boldsymbol{\alpha}^\top \mathbf{Q}^{\mathbf{h}} \boldsymbol{\alpha} \tag{6.14}$$

After a simple derivation, we can rewrite (6.14) as:

$$\max_{\mathbf{h}} \mathbf{h}^{\top}(\frac{1}{2}\hat{\mathbf{Q}} \odot (\hat{\boldsymbol{\alpha}}\hat{\boldsymbol{\alpha}}^{\top}))\mathbf{h} \tag{6.15}$$

where $\hat{\alpha}_i = 1/\left|G_m\right| \sum_{c,s} \alpha_{m,c,s}$ for $i \in I_m$ and $\hat{\mathbf{Q}} = \sum_{c,\hat{c},s,\hat{s}} \phi(x_i)^{\top}\phi(x_j)\lambda(i,j,c,\hat{c},s,\hat{s})$. Problem (6.15) can be solved approximately through enumerate the binary indicator vector $\mathbf{h}$ in a bag by bag fashion iteratively to maximize (6.15) until there is no change in $\mathbf{h}$. The detailed solutions for our instance-level MIL model are described in the Algorithm 6.1.

6.3 Experiments

In this section, we first conduct experiments on both image categorization and sub-categorization to demonstrate the superiority of our proposed approach. Then we analyze the parameter sensitivity and time complexity of our proposed approach in this section.

6.3.1 Image categorization

Experimental setting

We follow the setting in (Bergamo & Torresani 2010, Li & Fei-Fei 2010) and exploit web images as the training set, human-labeled images as the testing set. Particularly, we evaluate the performance of our approach and other baselines on the following datasets:

• PASCAL VOC 2007 (Everingham et al. 2010). The PASCAL VOC 2007 dataset contains 9963 images in 20 categories. Each category has training/validation data and test data. For this experiment, we only use the test data in PASCAL VOC 2007 as the benchmark testing set.

• STL-10 (Coates et al. 2011). The STL-10 dataset has ten categories, and each category of which contains 500 training images and 800 test images. We also use the test images in STL-10 as the benchmark testing set.

• CIFAR-10 (Krizhevsky & Hinton 2009). The CIFAR-10 dataset consists of 60000 images in 10 categories, with 6000 images per category, of which

5000 are training images and 1000 are test images. Similarly, we only use the test images in CIFAR-10 as the benchmark testing set.

After we obtain the selected PI, we treat each selected PI as a subcategory for the target category. The top 100 images were chosen for constructing the positive bags which corresponding to the selected subcategories. Negative bags can be obtained by randomly sampling a few irrelevant images. By treating each subcategory as a “bag” and the images therein as “instances”, we formulate noisy images removing and classifiers learning as an instance-level MIL problem. We specifically propose a new MIL model to select a subset of training images from each bag and simultaneously learn the optimal classifiers based on the selected images. We define each positive bag as having at least a portion of $\eta = 0.7$ positive instances and set the trade-off parameter $C_1 = 10^{-1}$. To compare with other baseline methods, we evenly select 500 images from positive bags for each category to learn the integrated classifier. In this experiment, the features are 4096-dimensional deep features based on AlexNet (Krizhevsky et al. 2012).

Baselines

To quantify the performance of our proposed approach, three set of weakly supervised baselines are selected to compare with our approach:

• Sub-categorization methods. The sub-categorization methods Sub-Cate (Hoai & Zisserman 2013) and RN-CMF(Ristin, Gall, Guillaumin & Van Gool 2015) can also be used to do image categorization. For method Sub-Cate, the candidate images are retrieved from the image search engine. Then we discover the subcategories of these candidate images through clustering by using visual features. We also evenly select 500 images from these subcategories to train image classifier. For method RN-CMF, we obtain the candidate images from the image search engine. We take the framework of Random Forests and the proposed regularized objective function to select 500 images for each category to train the image classier.

• MIL methods. The MIL methods contain instance-level method mi-

SVM (Andrews et al. 2003) and bag-level method sMIL (Bunescu & Mooney 2007). For method mi-SVM, the training images are also retrieved from the image search engine. Particularly, we take the proposed heuristic way to iteratively select 500 images for each category and train the image classifier. For method sMIL, we first retrieve the candidate images from the image search engine, then we partition the candidate images into a set of clusters. Each cluster is treated as a “bag” and the images therein as “instances”. Correspondingly, we take the proposed MIL method to select the 500 training images for each category and train the image classifier.

• Privileged information methods. The privileged information methods include sMIL-PI (Li et al. 2014), LIR (Wang & Ji 2015), WSDG-PI (Niu et al. 2017) and VCL (Divvala et al. 2014). For method sMIL-PI and WSDG-PI, we cope with noise in the labels of retrieved web images and incorporate the textual features extracted from the surrounding descriptions to modify the parameters of the classifier during training. For method LIR, we encode privileged information as regularization term to refine parameter estimation during training. Similarly, we also collect 500 images for each category to learn classifiers for sMIL-PI, WSDG-PI and LIR. For method VCL, we obtain the candidate privileged information from untagged corpora and leverage the proposed iterative method to purify the noisy privileged information and images. We evenly select 500 images from this selected privileged information for each category to learn the classifier.

Experimental results

The detailed and average performance comparison results are summarized in Fig 6.4, Fig 6.5, and Table 6.1. From Fig 6.4, Fig 6.5, and Table 6.1, we have the following observations:

Among the 20 categories in PASCAL VOC 2007, we achieved the best results in 19 categories. In the ten categories of STL-10 and CIFAR-10, we obtained the best results in all categories. Moreover, our approach also achieved the best average results on all three datasets.

The performance of privileged information methods sMIL-PI, LIR, WSDG-PI, VCL and our method was better than sub-categorization and MIL methods. The explanation is perhaps that it is necessary to leverage privileged information during the process of classifier learning. Learning directly from web images without privileged information may affect the performance of the classifier due to the limitations of only using visual features.

Privileged information methods VCL and our method performed better than three other PI methods sMIL-PI, LIR and WSDG-PI on the task of image categorization. One possible explanation is that the privileged information extracted from untagged corpora in both of our method and VCL is much richer and more accurate than three other methods in which the PI is obtained from the surrounding textual descriptions. Due to the limitations of personal knowledge, manually labeled PI is usually not rich enough. In addition, useful PI is often associated with noise in practice, and it is necessary to remove the noise before using them.

Finally, our proposed approach achieved the best average performance on all three datasets. Compared to MIL and sub-categorization methods, the classifiers learned by our approach not only using the visual features, but also the textual PI. Privileged information is usually more discriminative than the visual features in practical applications. For example, text descriptions are usually better than the raw image pixels to classify the objects. Compared to privileged information methods which extract PI from the surrounding textual descriptions, the privileged information extracted by our method from untagged corpora is much more accurate and general. So the learned classifiers are more robust. Compared to VCL which leverages an iterative mechanism in the process of noisy privileged information and web images removing, the PI and training images extracted by our method are much richer and have a better diversity. In addition, our method exploits multiple PI to learn integrated classifier is more robust than VCL which takes multiple PI to learn a single classifier.

Table 6.1: The average performance comparison on the PASCAL VOC 2007, STL-10 and CIFAR-10 dataset.

Method	**Dataset**		
	PASCAL	STL-10	CIFAR-10
sMIL	0.383	0.351	0.254
mi-SVM	0.414	0.381	0.278
RN-CMF	0.499	0.394	0.313
Sub-Cate	0.432	0.426	0.336
sMIL-PI	0.437	0.454	0.355
LIR	0.482	0.472	0.376
WSDG-PI	0.522	0.485	0.432
VCL	0.545	0.513	0.429
Ours	**0.582**	**0.557**	**0.464**

6.3.2 Image sub-categorization

Experimental setting

For image sub-categorization, we choose a subset of ImageNet as the testing benchmark dataset. The reason is that ImageNet which constructed according to the WordNet has a hierarchy structure. In particular, we select five categories including “airplane”, “bird”, “cat”, “dog” and “horse” as the target categories and all their leaf synsets as the subcategories. We are only concerned with the two-tier structure and deeper structure synsets are ignored. We obtain 5 categories and 97 subcategories. A detailed number of subcategories for each category in this experiment is provided in Table 6.2. The top 1000 images for each subcategory were retrieved from image search engine (Bing Image Search API-v7). We perform a cleanup step for broken links, webpages and obtain top ranked 700 images for each subcategory. We have a total number of 97 × 700 training images. We leverage the proposed MIL model to remove noise and learn classifiers. Specifically, we exploit the learned classifiers to re-rank the images in each subcategory according to the

Table 6.2: The detailed number of subcategories used for image subcategorization in this experiment.

Category	airplane	horse	bird	cat	dog
Subcategories	15	29	26	9	18

probability to be a positive sample. We sequentially select the top-ranked [100, 150, 200, 250, 300, 350, 400, 450, 500] images from each subcategory as the positive training samples to learn classifiers. 500 images per subcategory from ImageNet were selected as the testing data. In addition, we leveraged the top-ranked 500 images per subcategory as the positive training samples to learn classifiers and sequentially select [100, 150, 200, 250, 300, 350, 400, 450, 500] images per subcategory from ImageNet as the testing data. For this experiment, we also use the deep features based on AlexNet (Krizhevsky et al. 2012).

Baselines

We compare the image sub-categorization ability of our method with four baseline methods:

- multi-SVM (Andrews et al. 2003). For the multi-SVM method, the class number is 97. We directly use the retrieved images from the image search engine as the positive samples to learn classifiers (without noise pruning operation).

- Sub-Cate (Hoai & Zisserman 2013). Method Sub-Cate takes joint clustering and classification for subcategories discovering. For this experiment, the latent cluster number for each coarse category is known and equal to the number of given subcategories.

- RN-CMF (Ristin et al. 2015). For RNCMF method, the labeled training data is unavailable for both "coarse" categories and "fine" subcategories. The training images are retrieved from image search engine which may include noise due to the error index of image search engine. We assume there are

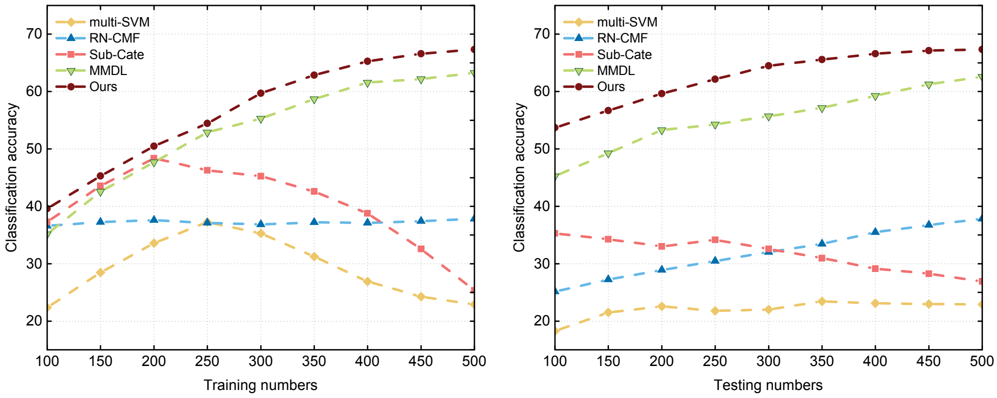

Figure 6.2: Sub-categorization accuracy (%) of the different methods (a) using a varying number of training images for per subcategory, and (b) using a varying number of testing images for per subcategory.

five trees corresponding to our five coarse categories and start the recursively learning. The depth of the tree for this experiment is all limited to two levels.

• MMDL (Wang, Wang, Bai, Liu & Tu 2013). MMDL formulate image selection as a multi-instance learning problem. For this experiment, the subcategories are assumed as "bags" and the retrieved images therein as instances. We take the proposed multi-instance learning function to select images from the retrieved images and learn the image classifiers.

Experimental results

Fig. 6.2 presents the image sub-categorization results achieved by different methods when using a varying number of training and testing images. The accuracy is measured by the average classification rate per subcategory.

By observing Fig. 6.2, we can see the best performance is achieved by our method, which produces significant improvements over method Sub-Cate and multi-SVM, particularly the number of positive training images over 250 for each subcategory. The reason is that our method considers the noisy images during the process of classifier learning. Due to the error index of image search engine, some noise may be included. We need to remove noise and select useful images from the retrieved web images to learn robust classifiers for each subcategory.

From Fig. 6.2, we notice that the performance of the multi-SVM and Sub-Cate peaks at the value of training numbers 200 or 250 and decreases monotonically after this peaks. One possible explanation is that the image search engine provides images based on the estimated relevancy concerning the query. Images far down in the ranking list are more likely to be noise, which may result in degrading of the sub-categorization accuracy, especially for non-robust methods.

It is interesting to note in Fig. 6.2, while method RN-CMF implements a form of noise removing, the classification accuracy did not improve with the number of positive training images increase. One possible explanation is that the noise in the training data is not the only factor that affects the classification accuracy. The visual distribution of the selected images is another important factor that we can't ignore. Furthermore, the poor accuracy of Sub-Cate suggests that naively adding the number of training images without considering the visual distributions not only does not help but worsens the classification accuracy.

By observing Fig. 6.2, our approach compares very favorably with competing algorithms, in terms of different numbers of training and testing images. Compared to method multi-SVM, Sub-Cate, RN-CMF, and MMDL, our approach achieves significant improvements in the sub-categorization accuracy. The reason is our proposed MIL model not only considers the possible presence of noise in the web training data, but also tries to ensure the diversity of the selected images for classifier learning.

6.3.3 Parameter Sensitivity Analysis

For parameter sensitivity analysis, we mainly concern the number of labeled positive and negative PI in the process of PI purifying and two parameters C_1 and η in our MIL model. PASCAL VOC 2007 was selected as the benchmark testing dataset to evaluate the performance variation of our proposed

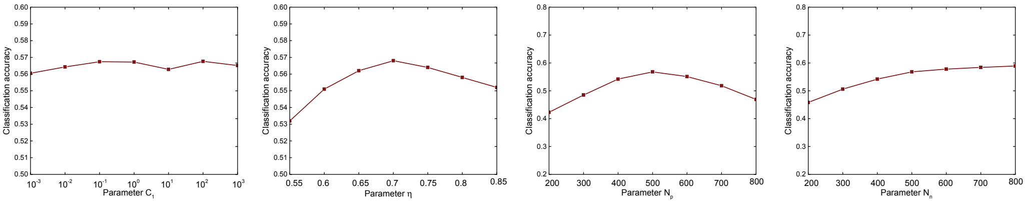

Figure 6.3: The parameter sensitiveness of C_1, η, N_p and N_n in terms of image categorization accuracy.

approach. In particular, we vary one parameter by fixing other parameters as the default value. Fig 6.3 presents the parameter sensitiveness of C_1, η, N_p and N_n in terms of image categorization accuracy on testing dataset.

By observing Fig 6.3 (a), we found our method is robust to the parameter C_1 when it is varied in a certain range [10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}]. From Fig 6.3 (b), we noticed that the performance of our method is growing when η increases but less than 0.7. The reason is perhaps that our training data was derived from image search engine. Due to the error index of image search engine, there may be too much noise in each bag which will result in decreasing the classification accuracy when $\eta \leqslant 0.7$. When η increases over 0.7, the performance of our method decreases. One possible explanation is that the training set is less diverse. With the increase of η, the number of subcategories is decreasing, which may lead to the degradation of domain robustness of the classifier.

By observing Fig 6.3 (c), we found the performance of our method is growing when N_p increases but less than 500. The explanation is that when $N_p \leqslant 500$, the performance of the noisy PI purifying classifier increases, and when N_p increases over 500, the noisy PI purifying classifier may have been over-fitted. In this condition, some positive PI may be removed by mistake, resulting in a decrease in the performance of the final integrated classifier. From Fig 6.3 (d), we observed that the performance of our method shows a relatively rapid increase when the number of $N_n \leqslant 500$. When the number of negative PI is larger than 500, the performance of our method increases at a relatively slower rate.

6.3.4 Time Complexity Analysis

During the process of our proposed multi-instance learning, we solve the convex problem in (6.9) by using the cutting-plane algorithm. Through finding the most violating candidate $\mathbf{h}_t$ and solve the MKL subproblem at each iteration, the time complexity of (6.9) can be approximately computed as $T \cdot O(\text{MKL})$, where the T is the number of iterations and the $O(\text{MKL})$ is the time complexity of the MKL sub-problem. According to (Platt 1999), the time complexity of MKL is between $t \cdot O(LCM)$ and $t \cdot O((LCM)^{2.3})$, where M, L, C are the numbers of latent domains, bags and categories respectively. t is the number of iterations in MKL.

6.4 Conclusions

In this chapter, we presented a new approach for enhancing classifier learning by using the collected web images. Different from previous works, our approach, while improving the accuracy and robustness of the classifier, greatly reduces the time and labor dependence. Specifically, we proposed a new instance-level MIL model to select a subset of training images from each selected privileged information and simultaneously learn the optimal classifiers based on the selected images. Extensive experimental results demonstrated the superiority of our proposed approach over existing weakly supervised state-of-the-art methods.

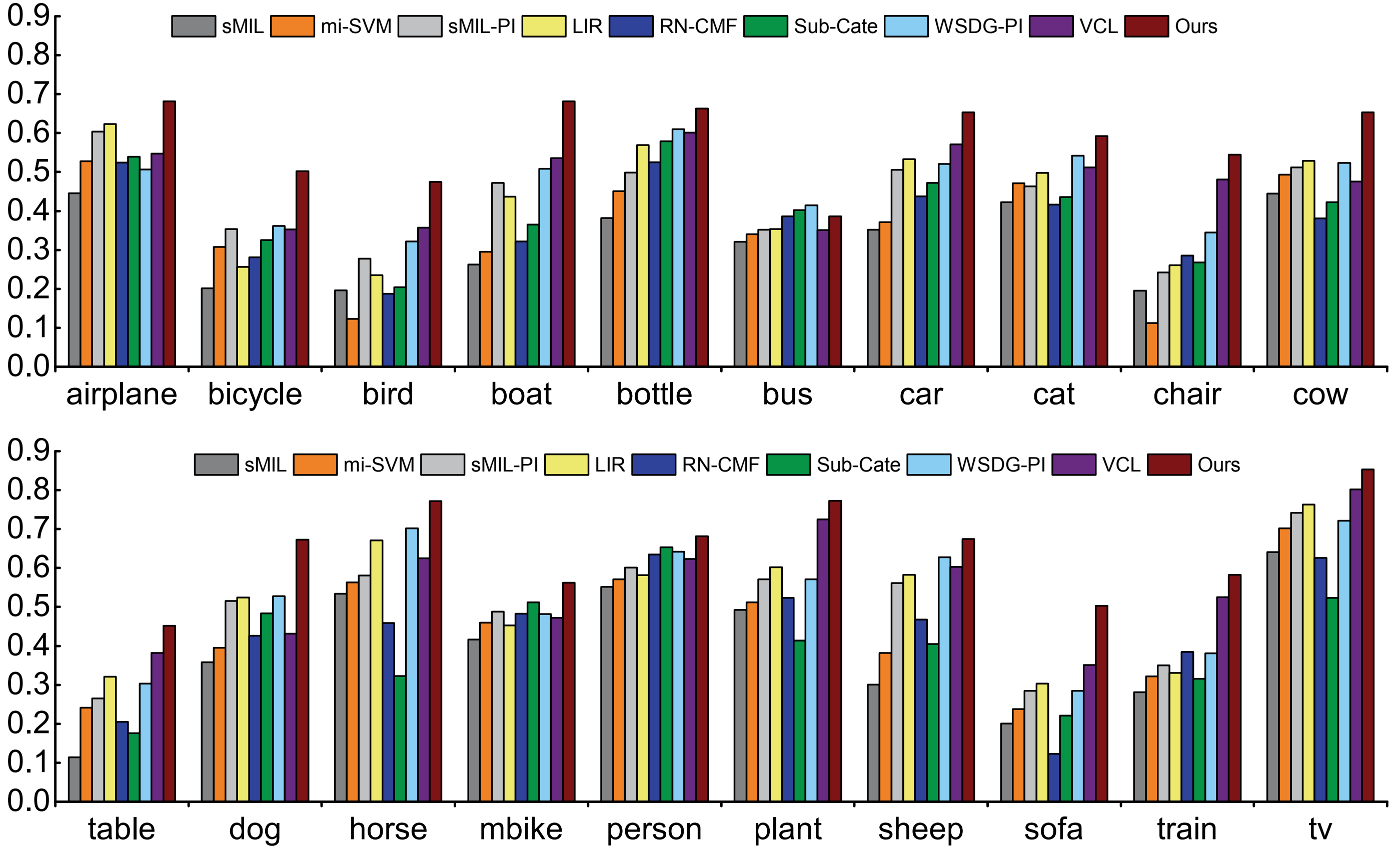

Figure 6.4: The detailed performance comparison over 20 categories on the PASCAL VOC 2007 dataset.

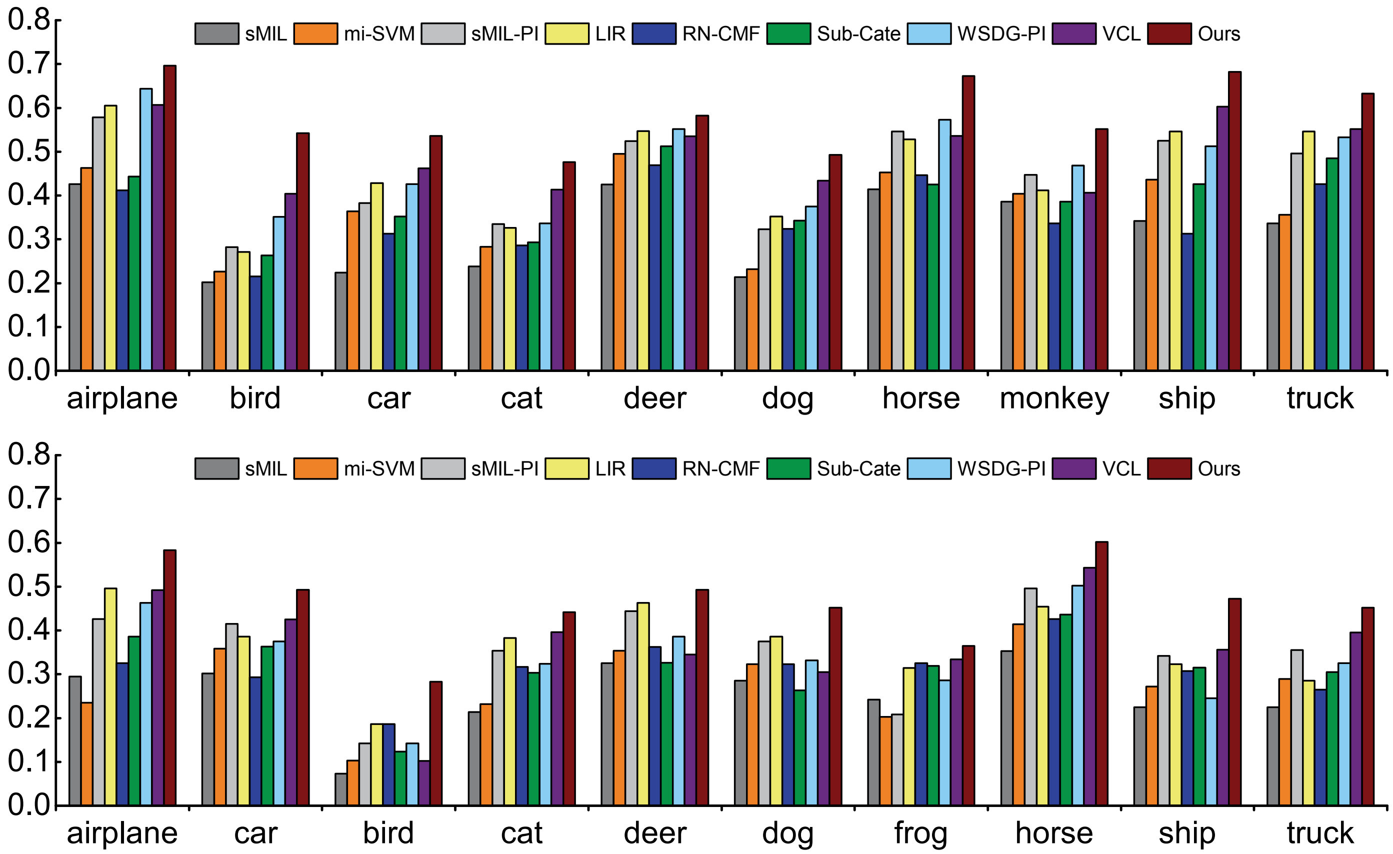

Figure 6.5: The detailed performance comparison over 10 categories on the (a) STL-10 dataset, (b) CIFAR-10 dataset.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

As the computer vision community considers more visual categories and greater intra-class variations; it is clear that larger and more exhaustive datasets are needed. However, the process of constructing such datasets is laborious and monotonous. It is unlikely that the manual annotation can keep pace with the growing need for annotated datasets. To reduce the cost of manual annotation, automatically constructing image datasets by using the web data has attracted more and more peoples attention. However, there are still three unsolved and partially solved problems that can be further discussed. In consideration of these problems, the main content and innovation of this thesis follow.

This thesis proposes a new framework for discovering and distinguishing multiple visual senses for polysemous words. We argue that the current poor performance of existing methods for image dataset construction is due to the visual polysemy. We solved the problem by allowing sense-specific diversity in search results. Compared to existing methods, our proposed method can not only figure out the right sense but also generates the right mapping between semantic and visual senses. The experimental results demonstrated the superiority of our proposed approach over existing weakly supervised state-of-the-art approaches.

This thesis presents a new framework for domain-robust image dataset construction which can be generalized well to unseen target domains. Three successive modules were employed in the framework, namely query expanding, noisy expansion filtering, and noisy image filtering. To verify the effectiveness of our proposed method, we constructed an image dataset DRID-20. Extensive experiments demonstrated the superiority of our approach.

This thesis presents an automatic image dataset construction framework. We aim at collecting accurate images for given queries from the Web. Specifically, we formulate noisy textual metadata removing and noisy images filtering as a multi-view and multi-instance learning problem separately. Our proposed approach not only improves the accuracy, but also enhances the diversity of the selected images. To verify the effectiveness of our proposed approach, we construct an image dataset with 100 categories. The experiments show significant performance gains by using the generated data of our approach on several tasks, such as image classification, cross-dataset generalization and object detection.

This thesis proposes a new approach for enhancing classifier learning by using the collected web images. Different from previous works, our approach, while improving the accuracy and robustness of the classifier, greatly reduces the time and labor dependence. Specifically, we proposed a new instance-level MIL model to select a subset of training images from each selected privileged information and simultaneously learn the optimal classifiers based on the selected images. Extensive experimental results demonstrated the superiority of our proposed approach over existing weakly supervised state-of-the-art methods.

7.2 Future Work

The purpose of this study is to reduce the cost of obtaining diverse, accurate, and high-quality images from the web. This thesis presents a series of theoretical research and experimental demonstrations on image dataset construction. Further directions of research are discussed below.

1) Due to the superiority of the deep model in the process of learning parameters, more and more scholars began to use deep learning based methods to solve problems and achieved good results. In the future, we will try to use deep learning based method to solve the polysemy, diversity, and accuracy problem in the process of image dataset construction.

2) Large-scale visual recognition. We eliminate the dependency on manually labelled data and propose to learning classifiers directly through web data. This makes large-scale visual recognition possible. In the future work, the use of web data for large-scale visual recognition is one of our research directions.

3) Understanding actions. Actions (e.g., "horse fighting", "reining horse") are too complex to be explained using simple primitives. Our collected images which have semantic refinement tags help in discovering a comprehensive vocabulary that covers all nuances of any action. For example, we have discovered over 150 different variations of the walking action including "primitives walking", "couple walking", "frame walking". Such an exhaustive vocabulary helps in generating fine-grained descriptions of images. In the future work, understanding actions through web data is also one of our research directions.

4) Paraphrasing. Rewriting a textual phrase in other words while preserving its semantics is an active research area in NLP. Our method can be used to discover paraphrases. For example, we discover that a "grazing horse" is visually similar to a "eating horse". Our collected data can be used to produce a visual similarity score for textual phrases. Establishing a measure of semantic similarity from the visual point of view is also a research direction for us.

Appendix A

Solutions

A.1 The detailed solutions to (5.1)

We denote reproducing kernels corresponding to $\mathcal{H}_{K^{(1)}}$ and $\mathcal{H}_{K^{(2)}}$ by

$$k_{\mathcal{H}_{K^{(1)}}} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$$

and

$$k_{\mathcal{H}_{K^{(2)}}} : \mathcal{X} \times \mathcal{X} \to \mathbb{R},$$

respectively. We introduce the notation for the "span of the data" in space:

$$\mathcal{L}_{\mathcal{H}_{K^{(1)}}} := \operatorname{span}\{k_{\mathcal{H}_{K^{(1)}}}(x_i, \cdot)\}_{i=1}^{l+u} \subset \mathcal{H}_{K^{(1)}}$$

and

$$\mathcal{L}_{\mathcal{H}_{K^{(2)}}} := \operatorname{span}\{k_{\mathcal{H}_{K^{(2)}}}(x_i, \cdot)\}_{i=1}^{l+u} \subset \mathcal{H}_{K^{(2)}}.$$

According to the Representer Theorem (Argyriou, Micchelli & Pontil 2009), we have

$$(f^{(1)*}, f^{(2)*}) \in \mathcal{L}_{\mathcal{H}_{K^{(1)}}} \times \mathcal{L}_{\mathcal{H}_{K^{(2)}}}.$$

Thus we can write the solution as:

$$f^{(1)*}(\cdot) = \sum_{i=1}^{u+l} \alpha_i k_{\mathcal{H}_{K^{(1)}}}(x_i, \cdot) \in \mathcal{L}_{\mathcal{H}_{K^{(1)}}}$$

$$f^{(2)*}(\cdot) = \sum_{i=1}^{u+l} \beta_i k_{\mathcal{H}_{K^{(2)}}}(x_i, \cdot) \in \mathcal{L}_{\mathcal{H}_{K^{(2)}}}$$

where

$$\alpha = (\alpha_1, ..., \alpha_{u+l}) \in \mathbb{R}^{u+l}$$

and

$$\beta = (\beta_1, ..., \beta_{u+l}) \in \mathbb{R}^{u+l}.$$

We denote an arbitrary element of $\mathcal{L}_{\mathcal{H}_{K^{(1)}}}$ and $\mathcal{L}_{\mathcal{H}_{K^{(2)}}}$ by

$$f_\alpha^{(1)} = \sum_{i=1}^{u+l} \alpha_i k_{\mathcal{H}_{K^{(1)}}}(x_i, \cdot)$$

and

$$f_\beta^{(2)} = \sum_{i=1}^{u+l} \beta_i k_{\mathcal{H}_{K^{(2)}}}(x_i, \cdot),$$

respectively. Kernel matrices

$$(K_{\mathcal{H}_{K^{(1)}}})_{ij} = k_{\mathcal{H}_{K^{(1)}}}(x_i, x_j)$$

and

$$(K_{\mathcal{H}_{K^{(2)}}})_{ij} = k_{\mathcal{H}_{K^{(2)}}}(x_i, x_j)$$

are partitioned into blocks corresponding to labeled and unlabeled points:

$$K_{\mathcal{H}_{K^{(1)}}} = \begin{pmatrix} A_{u\times u} \, C_{u\times l} \\ C'_{l\times u} \, B_{l\times l} \end{pmatrix} \; K_{\mathcal{H}_{K^{(2)}}} = \begin{pmatrix} D_{u\times u} \, F_{u\times l} \\ F'_{l\times u} \, E_{l\times l} \end{pmatrix}.$$

We now can rewrite the co-regularization term as:

$$\sum_{i=l+1}^{l+u} [f_\alpha^{(1)}(x_i) - f_\beta^{(2)}(x_i)]^2 = \left\| (AC)\alpha - (DF)\beta \right\|^2,$$

and it follows from the reproducing property

$$\left\| f_\alpha^{(1)} \right\|_{\mathcal{H}_{K^{(1)}}}^2 = \alpha' K_{\mathcal{H}_{K^{(1)}}} \alpha$$

and

$$\left\| f_\beta^{(2)} \right\|_{\mathcal{H}_{K^{(2)}}}^2 = \beta' K_{\mathcal{H}_{K^{(1)}}} \beta.$$

The objective function is quadratic in α and β and thereby a solution $(f^{(1)*}, f^{(2)*})$ can be found by differentiating and solving a system of linear equations:

$$\left[\frac{1}{l}JK_{\mathcal{H}_{K^{(1)}}} + \gamma_1 I + \lambda K_{\mathcal{H}_{K^{(1)}}}\right]\alpha - \lambda K_{\mathcal{H}_{K^{(2)}}}\beta = \frac{1}{l}Y$$

$$\left[\frac{u}{l}JK_{\mathcal{H}_{K^{(2)}}} + \gamma_2 I + \lambda K_{\mathcal{H}_{K^{(2)}}}\right]\beta - \lambda K_{\mathcal{H}_{K^{(1)}}}\alpha = \frac{u}{l}Y$$

where Y is a label vector given by $Y_i = y_i$ for $1 \leqslant i \leqslant l$ and $Y_i = 0$ for $l+1 \leqslant i \leqslant l+u$; J is a diagonal matrix given by $J_{ii} = |Y_i|$. The detailed solutions for the above linear equations can be found in (Sindhwani et al. 2005).

Bibliography

Andrews, S., Tsochantaridis, I. & Hofmann, T. (2003), Support vector machines for multiple-instance learning, *in* 'Advances in neural information processing systems', pp. 577–584.

Argyriou, A., Micchelli, C. A. & Pontil, M. (2009), 'When is there a representer theorem? vector versus matrix regularizers', *Journal of Machine Learning Research* **10**(Nov), 2507–2529.

Bach, F. R., Lanckriet, G. R. & Jordan, M. I. (2004), Multiple kernel learning, conic duality, and the smo algorithm, *in* 'Proceedings of the twenty-first international conference on Machine learning', ACM, p. 6.

Baktashmotlagh, M., Harandi, M. T., Lovell, B. C. & Salzmann, M. (2013), Unsupervised domain adaptation by domain invariant projection, *in* 'Computer Vision (ICCV), 2013 IEEE International Conference on', IEEE, pp. 769–776.

Batra, D., Yadollahpour, P., Guzman-Rivera, A. & Shakhnarovich, G. (2012), Diverse m-best solutions in markov random fields, *in* 'European Conference on Computer Vision', Springer, pp. 1–16.

Berg, T. L. & Forsyth, D. A. (2006), Animals on the web, *in* 'Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on', Vol. 2, IEEE, pp. 1463–1470.

Bergamo, A. & Torresani, L. (2010), Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach, *in* 'Advances in neural information processing systems', pp. 181–189.

Borth, D., Ji, R., Chen, T., Breuel, T. & Chang, S.-F. (2013), Large-scale visual sentiment ontology and detectors using adjective noun pairs, *in* 'Proceedings of the 21st ACM international conference on Multimedia', ACM, pp. 223–232.

Boyd, S. & Vandenberghe, L. (2004), *Convex optimization*, Cambridge university press.

Brefeld, U., Gärtner, T., Scheffer, T. & Wrobel, S. (2006), Efficient co-regularised least squares regression, *in* 'Proceedings of the 23rd international conference on Machine learning', ACM, pp. 137–144.

Bruzzone, L. & Marconcini, M. (2010), 'Domain adaptation problems: A dasvm classification technique and a circular validation strategy', *IEEE transactions on pattern analysis and machine intelligence* **32**(5), 770–787.

Bunescu, R. C. & Mooney, R. J. (2007), Multiple instance learning for sparse positive bags, *in* 'Proceedings of the 24th international conference on Machine learning', ACM, pp. 105–112.

Carneiro, G., Chan, A. B., Moreno, P. J. & Vasconcelos, N. (2007), 'Supervised learning of semantic classes for image annotation and retrieval', *IEEE transactions on pattern analysis and machine intelligence* **29**(3), 394–410.

Chang, C.-C. & Lin, C.-J. (2011), 'Libsvm: a library for support vector machines', *ACM transactions on intelligent systems and technology (TIST)* **2**(3), 27.

Chatterjee, N. & Mohan, S. (2008), Discovering word senses from text using random indexing, *in* 'International Conference on Intelligent Text Processing and Computational Linguistics', Springer, pp. 299–310.

Chen, X., Ritter, A., Gupta, A. & Mitchell, T. (2015), Sense discovery via co-clustering on images and text, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', pp. 5298–5306.

Chen, Y., Bi, J. & Wang, J. Z. (2006), 'Miles: Multiple-instance learning via embedded instance selection', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**(12), 1931–1947.

Chua, T.-S., Tang, J., Hong, R., Li, H., Luo, Z. & Zheng, Y. (2009), Nus-wide: a real-world web image database from national university of singapore, *in* 'Proceedings of the ACM international conference on image and video retrieval', ACM, p. 48.

Cilibrasi, R. L. & Vitanyi, P. M. (2007), 'The google similarity distance', *IEEE Transactions on knowledge and data engineering* **19**(3).

Coates, A., Ng, A. & Lee, H. (2011), An analysis of single-layer networks in unsupervised feature learning, *in* 'Proceedings of the fourteenth international conference on artificial intelligence and statistics', pp. 215–223.

Collins, B., Deng, J., Li, K. & Fei-Fei, L. (2008), Towards scalable dataset construction: An active learning approach, *in* 'European conference on computer vision', Springer, pp. 86–98.

Collobert, R. & Weston, J. (2008), A unified architecture for natural language processing: Deep neural networks with multitask learning, *in* 'Proceedings of the 25th international conference on Machine learning', ACM, pp. 160–167.

Dalal, N. & Triggs, B. (2005), Histograms of oriented gradients for human detection, *in* 'Computer Vision and Pattern Recognition, 2005. CVPR

2005. IEEE Computer Society Conference on', Vol. 1, IEEE, pp. 886–893.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. & Fei-Fei, L. (2009), Imagenet: A large-scale hierarchical image database, *in* 'Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on', IEEE, pp. 248–255.

Ding, Z., Shao, M. & Fu, Y. (2014), Latent low-rank transfer subspace learning for missing modality recognition., *in* 'AAAI', pp. 1192–1198.

Ding, Z., Shao, M. & Fu, Y. (2015), Deep low-rank coding for transfer learning., *in* 'IJCAI', pp. 3453–3459.

Divvala, S. K., Farhadi, A. & Guestrin, C. (2014), Learning everything about anything: Webly-supervised visual concept learning, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', pp. 3270–3277.

Duan, L., Li, W., Tsang, I. W.-H. & Xu, D. (2011), 'Improving web image search by bag-based reranking', *IEEE Transactions on Image Processing* **20**(11), 3280–3290.

Duan, L., Xu, D. & Tsang, I. W.-H. (2012), 'Domain adaptation from multiple sources: A domain-dependent regularization approach', *IEEE Transactions on Neural Networks and Learning Systems* **23**(3), 504–518.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J. & Zisserman, A. (2010), 'The pascal visual object classes (voc) challenge', *International journal of computer vision* **88**(2), 303–338.

Felzenszwalb, P. F., Girshick, D. & Ramanan, D. (2010), 'Object detection with discriminatively trained part-based models', *IEEE transactions on pattern analysis and machine intelligence* **32**(9), 1627–1645.

Fergus, R., Fei-Fei, L., Perona, P. & Zisserman, A. (2005), Learning object categories from google's image search, *in* 'Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on', Vol. 2, IEEE, pp. 1816–1823.

Fergus, R., Perona, P. & Zisserman, A. (2004), A visual category filter for google images, *in* 'European Conference on Computer Vision', Springer, pp. 242–256.

Fernando, B., Habrard, A., Sebban, M. & Tuytelaars, T. (2013), Unsupervised visual domain adaptation using subspace alignment, *in* 'Computer Vision (ICCV), 2013 IEEE International Conference on', IEEE, pp. 2960–2967.

Gong, B., Shi, Y., Sha, F. & Grauman, K. (2012), Geodesic flow kernel for unsupervised domain adaptation, *in* 'Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on', IEEE, pp. 2066–2073.

Gopalan, R., Li, R. & Chellappa, R. (2011), Domain adaptation for object recognition: An unsupervised approach, *in* 'Computer Vision (ICCV), 2011 IEEE International Conference on', IEEE, pp. 999–1006.

Griffin, G., Holub, A. & Perona, P. (2007), 'Caltech-256 object category dataset'.

Hoai, M. & Zisserman, A. (2013), 'Discriminative sub-categorization', pp. 1666–1673.

Hofmann, T. (1999), Probabilistic latent semantic analysis, *in* 'Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence', Morgan Kaufmann Publishers Inc., pp. 289–296.

Hua, X.-S. & Li, J. (2015), Prajna: Towards recognizing whatever you want from images without image labeling., *in* 'AAAI', pp. 137–144.

Huang, J., Gretton, A., Borgwardt, K. M., Schölkopf, B. & Smola, A. J. (2007), Correcting sample selection bias by unlabeled data, *in* 'Advances in neural information processing systems', pp. 601–608.

Jhuo, I.-H., Liu, D., Lee, D. & Chang, S.-F. (2012), Robust visual domain adaptation with low-rank reconstruction, *in* 'Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on', IEEE, pp. 2168–2175.

Kelley, Jr, J. E. (1960), 'The cutting-plane method for solving convex programs', *Journal of the society for Industrial and Applied Mathematics* **8**(4), 703–712.

Kim, S.-J. & Boyd, S. (2008), 'A minimax theorem with applications to machine learning, signal processing, and finance', *SIAM Journal on Optimization* **19**(3), 1344–1367.

Kiwiel, K. C. (1990), 'Proximity control in bundle methods for convex nondifferentiable minimization', *Mathematical programming* **46**(1-3), 105–122.

Krizhevsky, A. & Hinton, G. (2009), 'Learning multiple layers of features from tiny images'.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012), Imagenet classification with deep convolutional neural networks, *in* 'Advances in neural information processing systems', pp. 1097–1105.

Kulis, B., Saenko, K. & Darrell, T. (2011), What you saw is not what you get: Domain adaptation using asymmetric kernel transforms, *in* 'Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on', IEEE, pp. 1785–1792.

Li, L.-J. & Fei-Fei, L. (2010), 'Optimol: automatic online picture collection via incremental model learning', *International journal of computer vision* **88**(2), 147–168.

Li, W., Duan, L., Xu, D. & Tsang, I. W.-H. (2011), Text-based image retrieval using progressive multi-instance learning, *in* 'Computer Vision (ICCV), 2011 IEEE International Conference on', IEEE, pp. 2049–2055.

Li, W., Niu, L. & Xu, D. (2014), Exploiting privileged information from web data for image categorization, *in* 'European Conference on Computer Vision', Springer, pp. 437–452.

Li, Y.-F., Kwok, J. T., Tsang, I. W. & Zhou, Z.-H. (2009), A convex method for locating regions of interest with multi-instance learning, *in* 'Joint European Conference on Machine Learning and Knowledge Discovery in Databases', Springer, pp. 15–30.

Li, Y.-F., Tsang, I. W., Kwok, J. & Zhou, Z.-H. (2009), Tighter and convex maximum margin clustering, *in* 'Artificial Intelligence and Statistics', pp. 344–351.

Lin, W.-H., Jin, R. & Hauptmann, A. (2003), Web image retrieval re-ranking with relevance model, *in* 'Web Intelligence, 2003. WI 2003. Proceedings. IEEE/WIC International Conference on', IEEE, pp. 242–248.

Lin, Y., Michel, W. & Petrov, S. (2012), Syntactic annotations for the google books ngram corpus, *in* 'Proceedings of the ACL 2012 system demonstrations', Association for Computational Linguistics, pp. 169–174.

Loeff, N., Alm, C. O. & Forsyth, D. A. (2006), Discriminating image senses by clustering with multimodal features, *in* 'Proceedings of the COLING/ACL on Main conference poster sessions', Association for Computational Linguistics, pp. 547–554.

Maron, O. (1998), 'Learning from ambiguity'.

Michel, J.-B., Shen, J. et al. (2011), 'Quantitative analysis of culture using millions of digitized books', *science* **331**(6014), 176–182.

Mihalcea, R. (2007), Using wikipedia for automatic word sense disambiguation, *in* 'Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference', pp. 196–203.

Miller, G. A. (1995), 'Wordnet: a lexical database for english', *Communications of the ACM* **38**(11), 39–41.

Niu, L., Li, W. & Xu, D. (2015), Visual recognition by learning from web data: A weakly supervised domain generalization approach, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', pp. 2774–2783.

Niu, L., Li, W., Xu, D. & Cai, J. (2017), 'Visual recognition by learning from web data via weakly supervised domain generalization', *IEEE transactions on neural networks and learning systems* **28**(9), 1985–1999.

Pantel, P. & Lin, D. (2002), Discovering word senses from text, *in* 'Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, pp. 613–619.

Pennington, J., Socher, R. & Manning, C. (2014), Glove: Global vectors for word representation, *in* 'Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)', pp. 1532–1543.

Platt, J. C. (1999), '12 fast training of support vector machines using sequential minimal optimization', *Advances in kernel methods* pp. 185–208.

Prest, A., Leistner, C., Civera, C. & Ferrari, V. (2012), Learning object class detectors from weakly annotated video, *in* 'Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on', IEEE, pp. 3282–3289.

Rakotomamonjy, A., Bach, F. R., Canu, S. & Grandvalet, Y. (2008), 'Simplemkl', *Journal of Machine Learning Research* **9**(Nov), 2491–2521.

Ristin, M., Gall, J., Guillaumin, M. & Van Gool, L. (2015), From categories to subcategories: large-scale image classification with partial class label refinement, *in* 'Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on', IEEE, pp. 231–239.

Russell, B. C., Torralba, A., Murphy, K. P. & Freeman, W. T. (2008), 'Labelme: a database and web-based tool for image annotation', *International journal of computer vision* **77**(1-3), 157–173.

Saenko & Trevor (2009), Unsupervised learning of visual sense models for polysemous words, *in* 'Advances in Neural Information Processing Systems', pp. 1393–1400.

Schroff, F., Criminisi, A. & Zisserman, A. (2011), 'Harvesting image databases from the web', *IEEE transactions on pattern analysis and machine intelligence* **33**(4), 754–766.

Shao, M., Kit, D. & Fu, Y. (2014), 'Generalized transfer subspace learning through low-rank constraint', *International Journal of Computer Vision* **109**(1-2), 74–93.

Siddiquie, B. & Gupta, A. (2010), Beyond active noun tagging: Modeling contextual interactions for multi-class active learning, *in* 'Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on', IEEE, pp. 2979–2986.

Sindhwani, V., Niyogi, P. & Belkin, M. (2005), A co-regularization approach to semi-supervised learning with multiple views, *in* 'Proceedings of ICML workshop on learning with multiple views', Vol. 2005, Citeseer, pp. 74–79.

Siva, P. & Xiang, T. (2011), Weakly supervised object detector learning with model drift detection, *in* 'Computer Vision (ICCV), 2011 IEEE International Conference on', IEEE, pp. 343–350.

Snow, R., Prakash, S., Jurafsky, D. & Ng, A. Y. (2007), Learning to merge word senses, *in* 'Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (emnlp-conll)'.

Speer, R. & Havasi, C. (2013), Conceptnet 5: A large semantic network for relational knowledge, *in* 'The People?s Web Meets NLP', Springer, pp. 161–176.

Torralba, A. & Efros, A. A. (2011), Unbiased look at dataset bias, *in* 'Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on', IEEE, pp. 1521–1528.

Torralba, A., Fergus, R. & Freeman, W. T. (2008), '80 million tiny images: A large data set for nonparametric object and scene recognition', *IEEE transactions on pattern analysis and machine intelligence* **30**(11), 1958–1970.

Veronis, J. & Ide, N. M. (1990), Word sense disambiguation with very large neural networks extracted from machine readable dictionaries, *in* 'Proceedings of the 13th conference on Computational linguistics-Volume 2', Association for Computational Linguistics, pp. 389–394.

Vijayanarasimhan, K. (2014), 'Large-scale live active learning: Training object detectors with crawled data and crowds', *International Journal of Computer Vision* **108**(1-2), 97–114.

Vijayanarasimhan, S. & Grauman, K. (2008), Keywords to visual categories: Multiple-instance learning forweakly supervised object categorization, *in* 'Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on', IEEE, pp. 1–8.

Wan, K.-W., Tan, A.-H., Lim, J.-H., Chia, L.-T. & Roy, S. (2009), A latent model for visual disambiguation of keyword-based image search., *in* 'BMVC', Vol. 2, p. 7.

Wang, X., Wang, B., Bai, X., Liu, W. & Tu, Z. (2013), Max-margin multiple-instance dictionary learning, *in* 'International Conference on Machine Learning', pp. 846–854.

Wang, Z. & Ji, Q. (2015), Classifier learning with hidden information, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', pp. 4969–4977.

Xiao, J., Hays, J., Ehinger, K. A., Oliva, A. & Torralba, A. (2010), Sun database: Large-scale scene recognition from abbey to zoo, *in* 'Computer vision and pattern recognition (CVPR), 2010 IEEE conference on', IEEE, pp. 3485–3492.

Xu, Z., Li, W., Niu, L. & Xu, D. (2014), Exploiting low-rank structure from latent domains for domain generalization, *in* 'European Conference on Computer Vision', Springer, pp. 628–643.

Yao, Y., Hua, X.-s., Shen, F., Zhang, J. & Tang, Z. (2016), A domain robust approach for image dataset construction, *in* 'Proceedings of the 2016 ACM on Multimedia Conference', ACM, pp. 212–216.

Yao, Y., Zhang, J., Shen, F., Hua, X., Xu, J. & Tang, Z. (2016), Automatic image dataset construction with multiple textual metadata, *in* 'Multimedia and Expo (ICME), 2016 IEEE International Conference on', IEEE, pp. 1–6.

Yao, Y., Zhang, J., Shen, F., Hua, X., Xu, J. & Tang, Z. (2017), 'Exploiting web images for dataset construction: A domain robust approach', *IEEE Transactions on Multimedia* **19**(8), 1771–1784.

Yarowsky, D. (1992), Word-sense disambiguation using statistical models of roget's categories trained on large corpora, *in* 'Proceedings of the 14th conference on Computational linguistics-Volume 2', Association for Computational Linguistics, pp. 454–460.

Yarowsky, D. (1995), Unsupervised word sense disambiguation rivaling supervised methods, *in* ‘Proceedings of the 33rd annual meeting on Association for Computational Linguistics’, Association for Computational Linguistics, pp. 189–196.

Yuille, A. L. & Rangarajan, A. (2003), ‘The concave-convex procedure’, *Neural computation* **15**(4), 915–936.