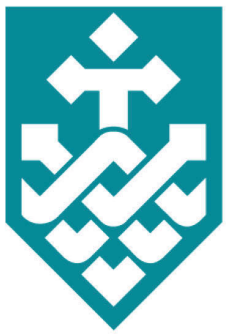
Multinomial Latent Logistic Regression



Zhe Xu

Faculty of Engineering and Information Technology

University of Technology, Sydney

A thesis submitted for the degree of

Doctor of Philosophy

November 2016

Certificate of Original Authorship

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.

Student: Zhe Xu
Date: 21/10/2016

I would like to dedicate this thesis to my loving parents
Nan Huang and *Yuan Xu*

Acknowledgements

I would like to take this good opportunity to appreciate my advisors, colleagues, friends and also my family for their significant help during my doctoral study in University of Technology, Sydney.

First of all, I would like to express my sincere appreciation and deep gratitude to my advisor supervisor **Prof. Dacheng Tao** for his unlimited patience, generous support, and supportive guidance. I'm particularly impressed by his incredible enthusiasm and high standard in academic research, which encourage me to breakthrough my own setting limits and submit papers to the leading journals or conferences in my research field.

I would also like to thank my advisor **Prof. Ya Zhang** from Shanghai Jiao Tong University. She always gives me plenty of freedom to explore and timely constructive suggestions to help me out of difficulties. Without the effect by her, Prof. Xiaokang Yang and Prof. Chengqi Zhang, I would not have the opportunity to study here in UTS as a dual-degree PhD student.

I have been fortunate to work in UTS and Centre for Quantum Computation and Intelligent Systems (QCIS) directed by Prof. Chengqi Zhang. QCIS provides full support for me to attend top conferences including ECCV, ICCV and KDD, where I got the opportunities to learn from many world-famous experts. It's really a pleasure for me to work in thus a great team and around so many brilliant minds in QCIS. Studying in QCIS and also UTS will be a fantastic memory that I will never forget.

I am also deeply indebted to Dr. Jun Zhu who led me into the field of computer vision. He shows me how interesting my research objective

is, which encourages me constantly during the journey of exploration in the following years. Moreover, I also want to give special thanks to my excellent collaborators: Zhibin Hong and Shaoli Huang for their brilliant work and timely support, and also to my dear colleagues and friends I met in QCIS: Dr. Bozhong Liu, Chunyang Liu, Meng Fang, Tongliang Liu, Mingming Gong, Maoying Qiao, Qiang Li, Runxin Wang, Changxing Ding, Zhiguo Long, Caishi Fang, Wei Bian, Shirui Pan, Yong Luo, Xiao Liu, Dianshuang Wu, Weilong Hou, Chang Xu, Chen Gong, Sujuan Hou, Haishuang Wang, Jia Wu, Wei Yang, Qin Zhang, Yali Du, Hao Xiong, Jiankang Deng, Xiao Liu, Peng Hao, Liu Liu, Guodong Long, Jing Jiang, Peng Zhang, Barbara Munday, Prof. Bo Du, Prof. Xianhua Ben, Prof. Wankou Yang, Prof. Shigang Liu, Prof. Xianhua Zeng, for the inspiring discussions, kind support and companionship.

I am also grateful to all the other friends: Guanbo Huang, Wei Xu, Weiyuan Chen, Xiaohang Ren, Zhiyi Tan, Jialin Li, Qing Wang, Weiyuan Chen, for their support and company during both joyful and stressful times.

Finally, I would like to express my deeply felt gratitude to my parents, who never excoriate me when I make mistakes but show me how to do it the right way, who never asks for anything but give me everything they have, who gives me such a wonderful place to grow up. It's you who make me the man who I am now. Thank you, from the bottom of my heart.

Abstract

We are arriving at the era of big data. The booming of data gives birth to more complicated research objectives, for which it is important to utilize the superior discriminative power brought by explicitly designed feature representations. However, training models based on these features usually requires detailed human annotations, which is being intractable due to the exponential growth of data scale.

A possible solution for this problem is to employ a restricted form of training data, while regarding the others as latent variables and performing latent variable inference during the training process. This solution is termed *weakly supervised learning*, which usually relies on the development of latent variable models. In this dissertation, we propose a novel latent variable model - **multinomial latent logistic regression (MLLR)**, and present a set of applications on utilizing the proposed model on weakly supervised scenarios, which, at the same time, cover multiple practical issues in real-world applications.

We first derive the proposed MLLR in Chapter 3, together with theoretical analysis including the concave and convex property, optimization methods, and the comparison with existing latent variable models on structured outputs. Our key discovery is that by performing "maximization" over latent variables and "averaging" over output labels, MLLR is particularly effective when the latent variables have a large set of possible values or no well-defined graphical structure is existed, and when probabilistic analysis is preferred on the output predictions. Based on it, the following three sections will discuss the application of MLLR in a variety of tasks on weakly supervised learning.

In Chapter 4, we study the application of MLLR on a novel task of architectural style classification. Due to a unique property of this

task that rich inter-class relationships between the recognizing classes make it difficult to describe a building using "hard" assignments of styles, MLLR is believed to be particularly effective due to its ability to produce probabilistic analysis on output predictions in weakly supervised scenarios. Experiments are conducted on a new self-collected dataset, where several interesting discoveries on architectural styles are presented together with the traditional classification task.

In Chapter 5, we study the application of MLLR on an extreme case of weakly supervised learning for fine-grained visual categorization. The core challenge here is that the inter-class variance between subordinate categories is very limited, sometimes even lower than the intra-class variance. On the other hand, due to the non-convex objective function, latent variable models including MLLR are usually very sensitive to the initialization. To conquer these problems, we propose a novel multi-task co-localization strategy to perform warm start for MLLR, which in turn takes advantage of the small inter-class variance between subordinate categories by regarding them as related tasks. Experimental results on several benchmarks demonstrate the effectiveness of the proposed method, achieving comparable results with latest methods with stronger supervision.

In Chapter 6, we aim to further facilitate and scale weakly supervised learning via a novel knowledge transferring strategy, which introduces detailed domain knowledge from sophisticated methods trained on strongly supervised datasets. The proposed strategy is proved to be applicable in a much larger web scale, especially accounting for the ability of performing noise removal with the help of the transferred domain knowledge. A generalized MLLR is proposed to solve this problem using a combination of strongly and weakly supervised training data.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

With the rapid evolution of information systems and online sharing medias in the Internet, the volume of data we are facing nowadays is becoming increasingly enormous. Automatically discovering, analyzing, and understanding knowledge, therefore, is widely acknowledged to be crucial in various real-world applications, including finance, environment, healthcare, engineering, *etc.* Theoretically, in the context of data mining and artificial intelligence, the approach of learning patterns or structures from data automatically is termed "statistical machine learning", or "machine learning" for short. In general, machine learning algorithms can be categorized into the following two classes.

Unsupervised Learning. Unsupervised learning aims to discover underlying structures of data without any human labeling effort. In particular, an unsupervised learning method uses a set of unlabeled training examples $\{x_i\}_{i=1}^n$, where each $x_i \in \mathbb{R}^d$ is represented as a d-dimensional feature vector. Typical examples include clustering, the expectation-maximization algorithm, and principal component analysis [65].

Supervised Learning. Supervised learning targets on improving the recognizing performance by introducing certain forms of supervised signal. Training data in supervised learning scenarios contain a set of training examples, each of which is a pair (x_i, y_i) consisting of an input object and a desired output value.

The goal is then to produce an inferred function from training data that could be used for mapping new examples, such as classification and regression [143].

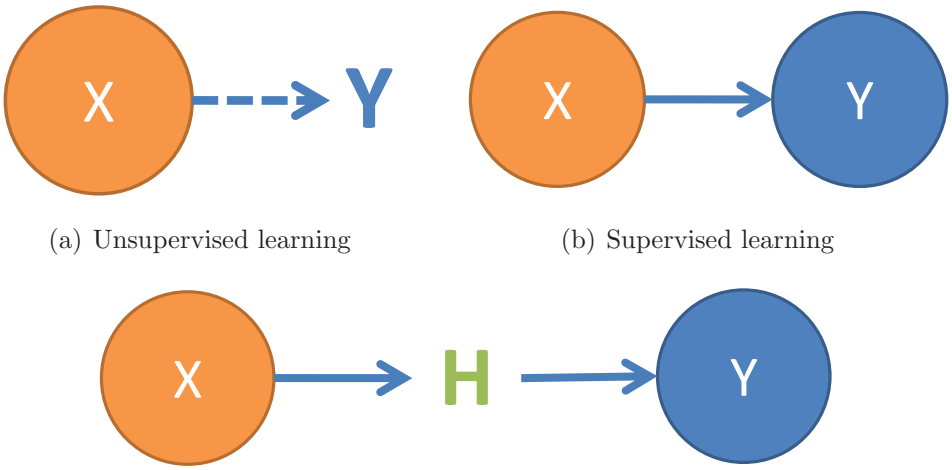
Compared to unsupervised learning methods, supervised learning is believed to be more specific and powerful due to its employment of manually labeled supervision signals, leading to many successful real-world applications such as face recognition [140], handwritten digit recognition [81], and fraud detection [135]. However, there is always a cost in the performance boost brought by the additional supervised signal. In practice, it is extremely time consuming to provide detailed annotations in certain applications, for example pixel-level labels for image segmentation, especially considering the increase of data scale in the era of big data. Therefore, one may want to seek a method that is able to generate high performing models with only a limited supply of supervised signals.

In general, there are two routines for learning with limited training data. The first one is to learn with limited amount of supervised training data, while exploiting a larger number of unlabeled data to facilitate the learning procedure. This strategy is called semi-supervised learning [24]. The other routine is to learn with limited forms of supervised signals. In this case, although some kind of supervised signals are given for all training examples, there is still some critical information, which is not directly provided or observable but plays an important role in statistical modeling, remained missing in certain applications. Algorithms are then designed to infer this hidden information from observed labels, and to train classifiers or regressors based on the inferred information.

1.2 Weakly Supervised Learning and Latent Variable Models

1.2.1 What is weakly supervised learning?

Although being arguable in the literature, in this thesis, the term *"weakly supervised learning"* is specifically referred to the second strategy discussed in the last paragraph, *i.e.*, learning with "weak" supervision covering only part of the functional variables. The remaining unlabeled variables, or hidden information,



(a) Unsupervised learning (b) Supervised learning

(c) Weakly supervised learning

Figure 1.1: Illustration of unsupervised learning, supervised learning, and weakly supervised learning. A variable within a circle indicates given information, while others indicate hidden variables.

are usually represented by latent variables and can be inferred from the observed information. It is critical to note that the definition of weakly supervised learning does not require the supervised signal to be absolutely "weak" to some certain extent; instead, the word "weak" is in fact a **relative concept** - it indicates that the given supervision during training is weaker than the feature representation employed in the model. From this perspective, the employment of latent variables is the key of weakly supervised learning that act as a bridge between the feature vectors and supervised labels.

The relationship between weakly supervised learning and two basic strategies of unsupervised learning and supervised learning is illustrated in Figure 1.1. Similar to standard supervised learning scenarios, in weakly supervised learning, each input sample x_i is associated with an output label y_i. However, by introducing an intermediate latent variable h, each input sample now corresponds to a set of feature vectors according to multiple assignment of h. The goal is therefore to learn a hypothesis given x and y, while the assignment of latent variable h is inferred automatically through the observed information.

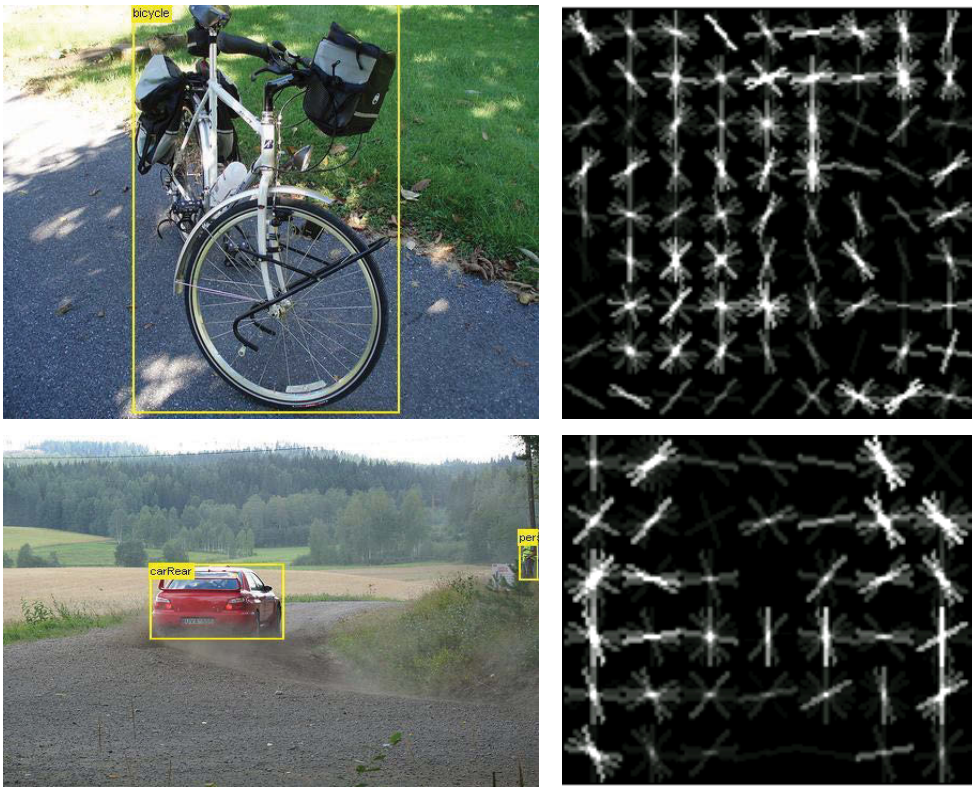


Figure 1.2: Demonstration of the requirement of object-level bounding-box annotations. The models in the right shows template-HOG results learned by DPM [47] using object-level supervision.

1.2.2 An intuitive example

There is a wide variety of hidden information (*e.g.*, spatial relations, data structures, behavioral and mental states) in diverse research fields such as computer vision, natural language processing, speech analysis and public health. As a result, many learning models have been proposed to exploit the value of hidden information based on latent variables, attracting increasing attention in applications in the above research fields.

Here we give a heuristic example of weakly supervised learning in object recognition to show how it operates in practical applications. Specifically, our goal is to predict the existence and also the accurate location of a given object class in an image. We consider a weakly supervised setting, where only image-level labels, which indicate whether a concept exists in an image or not, are given during

training. As shown in Figure 1.2, the discriminative power of image-level labels is rather limited in this case, especially considering that the background regions of the object categories "car" and "bicycle" are very similar and confusing.

To explicitly model the recognizing objects, we employ object-level feature representations extracted from object bounding boxes, *i.e.*, a region that bounds the object. These features are supposed to be much more powerful than image-level features since they directly model the visual appearance of a specific object than the whole image. Since the exact object locations in training examples are not given, they are regarded as latent variables and are inferred using the predicted class label and image features. The learning process consists of two stages: predicting the object location through latent variable inferring, and training object classifiers based on the inferred locations. Classifiers learned using this strategy are naturally more focused on the object itself and thus are more specific for the recognizing task (as shown in the rightmost column in Figure 1.2).

1.2.3 Latent variable models with structured output or multi-class prediction

Weakly supervised learning is closely related to the topic of latent variable models. In particular, in applications including object recognition, protein structure mining, and natural language parsing, usually the hidden variables or output labels have a graphical structure or involve multi-class predictions.

Formally, in latent variable models, each training data is represented by a triplet (x, y, h), where x and y are an input object and the respective output label, similar to generic supervised learning scenarios, while h is an additional latent variable. The problem can be modeled as $x \rightarrow h \rightarrow y$, which is further defined in the following two phases.

Latent variable completion. The first phase conducts $x \rightarrow h$, which is also called a latent variable completion problem. Specifically, this process is performed on each single input example x; given a finite (or an infinite) set of latent variables $H = \{h_j\}$, the goal is to infer the feature vector for the input example based on the current model status, considering all possible assignments of h.

There are some typical strategies in this phase:

- Maximum a *posteriori* (MAP) inference, also can be regarded as performing "maximizing over h". This method deterministically assigns the latent variables to their most likely states; thus the feature vector with the optimized assignment h^* is then representing the input example.

- Marginal inference, also referred to as "averaging over h". This method follows the Bayesian assumption that assigns the resultant feature representation as a weighted sum of the feature vector for each latent assignment.

- Other inference. Except for the previous two strategies, one can also design the inference approach flexibly. For example, in object recognition tasks, a possible selection is to perform the marginal inference over a limited range of latent variables reside near the optimized location, which in fact combines the two strategies in a reasonable way.

Hypothesis training. Given the inferred feature vectors for input examples, the second phase of latent variable models solves the learning problem of $(x \rightarrow h) \rightarrow y$. It is nearly identical to standard supervised learning formulations, where the goal is to obtain a hypothesis based on the input feature vectors and the desired output labels.

Methods in this phase also have several possible selections:

- Maximum a *posteriori* (MAP) inference, *i.e.*, "maximizing over y". This strategy is inspired by the support vector machines (SVMs) which maximize the margin between positive and negative examples. Although the objective function is usually non-convex, it is proved that surrogate upper bounds are effective for solving this problem.

- Marginal inference, *i.e.*, "averaging over y". The Bayesian method in this strategy can be regarded as generalization of the traditional conditional random fields or logistic regression.

To present a more intuitive explanation, considering the example in Section 1.2.2, the latent variable completion phase finds the optimal location of objects and extracts features on the inferred location. Hypothesis training is then conducted based on the inferred latent variables. These two phases are performed iteratively to produce the final models.

Different inference approaches in these two phases lead to a variety of latent variable models, which show different characteristics in various applications. In practice, it is impossible to identify a best performing strategy under all circumstances. One should select a proper latent variable algorithm based on real-world considerations flexibly.

1.2.4 Motivation of the proposed latent variable paradigm

Although existing latent variable models such as the latent structural support vector machines (LSSVMs) [167] and the hidden conditional random fields (HCRFs) [113] have seen reasonable achievements, they may be suboptimal in certain applications. Due to the lack of an investigation and comparison of the properties of existing latent variable models, a significant problem in real-world applications is then how to select a proper algorithm based on the characteristics of the applications.

In this dissertation, we propose a novel latent variable paradigm termed multinomial latent logistic regression (MLLR) that addresses certain problems of existing latent variable models on particular applications. By introducing latent variables into multimonial logistic regression algorithm, the new paradigm MLLR incorporates an MAP inference over latent variable h and a marginal inference over output labels y. It has certain advantages by: 1) being efficient in latent variable inference; 2) enabling powerful optimization methods for solving the objective function; 3) providing effective probabilistic analysis on output labels. Based on these discussions, a thorough investigation of existing latent variable models with structured output or multi-class prediction is presented, which provides practical advice on how to select a proper algorithm in real-world applications.

Following the standard way to study the characteristics of the a new learning paradigm, we will concentrate on two major components: theoretical analysis and practical analysis. The main content of this thesis is summarized as follows.

- Derive the objective function of the proposed paradigm from logistic regression.
- Study the convexity and smoothness property of the proposed latent variable paradigm and design effective optimization methods.

- Conduct a thorough investigation on the difference and connection between the proposed algorithm and existing latent variable paradigms, so as to propose advice on how to select the appropriate algorithm under different application scenarios.

- Propose a novel application that the proposed algorithm can have the most of its advantage over existing algorithms, *i.e.*, efficient inference on latent variables and reasonable probabilistic explanation on output predictions.

- Study practical issues of latent variable models, including probabilistic output analysis, forms of initialization, and optimization methods.

- Based on the proposed paradigm, study weakly supervised learning on novel applications including fine-grained visual categorization and webly supervised object recognition.

1.3 Significance and Organization

By proposing a new latent variable paradigm and studying the characteristics of the new paradigm from multiple aspects, the significance of this thesis is summarized as follows:

Theoretical Significance. This thesis introduces a new latent variable paradigm MLLR which implements maximization inference over latent variables and employs logistic loss functions. The proposed paradigm MLLR, along with HCRF [113], LSSVM [167] and MSSVM [107], compose a complete set of latent variable models with structured outputs. Two optimization algorithms are proposed for solving MLLR. Meanwhile, by analyzing the properties of the proposed MLLR, many fundamental issues in machine learning are studied, including multi-instance learning, non-convex optimization, and sub-gradient descent. The effectiveness of MLLR is discussed in several challenging computer vision tasks, such as weakly-supervised object classification, fine-grained object recognition, architectural style classification, and human action classification. MLLR provides a new selection for researchers on various applications that may benefit from the use of latent variables, especially in weakly-supervised multi-class classification

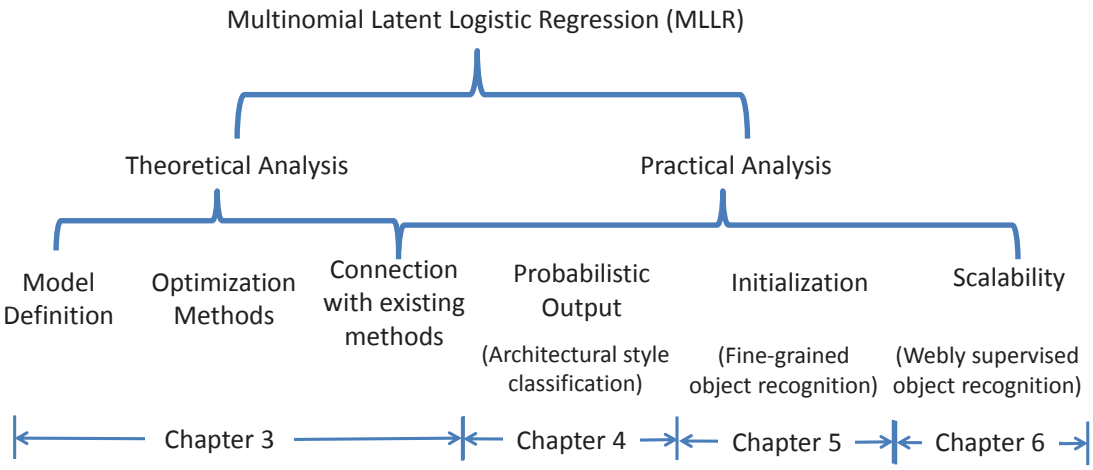


Figure 1.3: The structure of the thesis.

problems where efficient inference is preferred and rich inter-class relationships exist.

Practical Significance. Object recognition is an essential issue in many practical computer vision applications, such as robotics, object counting and monitoring, automatic car parking, and giving expert advices in medical applications. Weakly-supervised learning, by exploiting cheap and robust image-level annotations, could lead to applications with a large number of training data and training objectives (fine-grained categories) without extensive labeling effort. The proposed latent variable paradigm is particularly effective in weakly-supervised object recognition tasks. As a result, this research will contribute to building more robust object recognition system from vast number of online data, especially when there are rich relationships between object categories.

This thesis is organized as follows (Figure 1.3):

- Chapter 2 briefly reviews existing works on latent variable models, weakly supervised learning, and the application of fine-grained visual categorization.

- Chapter 3 describes the theoretical fundamentals of the proposed multinomial latent logistic regression, including objective function, convexness

analysis, and optimization methods. A thorough investigation of the difference and connection between MLLR and existing related methods is then presented, showing discussions about the optimized application scenarios for those methods.

- Chapter 4 studies MLLR on a novel application of architectural style classification. The proposed MLLR is proved to be particularly effective in this application as the rich inter-class relationships require an effective probabilistic analysis on output class labels, while efficient latent variable inference is also important.

- Chapter 5 investigates weakly supervised learning on an extreme scenario of fine-grained visual categorization. A carefully designed initialization strategy is proved to be crucial in this problem considering the non-convex objective function of MLLR.

- Chapter 6 generalizes the application scenario of weakly supervised learning to web scale. A fine-grained object recognition framework is proposed that utilizes a large number of weakly supervised web images to augment existing strongly supervised datasets with relatively smaller scale.

- Chapter 7 concludes the thesis and discusses some possible future directions.

Chapter 2

Related Work

As discussed in Chapter 1, the main task of this thesis is to propose a new latent variable model that is effective in weakly supervised learning scenarios, and to study practical issues when exploiting the latent variable models in various applications. This chapter provides a comprehensive overview of existing latent variable models with structured outputs and the associated optimization methods, and outlines the position of the proposed MLLR in the literature. Meanwhile, we also include a brief review of weakly supervised learning in object recognition, and introduce recent developments on two of the latest object recognition tasks: fine-grained visual categorization and webly supervised object recognition.

2.1 Latent Variable Models with Structured Outputs

Here we first briefly review related works on latent variable models, mainly with multi-class outputs or structured predictions, and demonstrate the connections and differences between them. As will be shown later, the proposed multinomial latent logistic regression algorithm can be regarded as a special case of the generalized latent variable model.

Given a training set S of N examples associated with their labels as $S = \{(x_1, y_1), ..., (x_N, y_N)\}$, where $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$. In latent variable models, we assume that the output values are not only characterized by the input x, but

also depend on some latent or hidden variables $h \in \mathcal{H}$. Suppose that $\phi(x, y, h) : \mathcal{X} \times \mathcal{Y} \times \mathcal{H} \rightarrow \mathbb{R}^D$ defines the feature vector describing the relations among (x, y, h), and $w \in \mathbb{R}^D$ are the model parameters, then each pair of (x, y, h) attains a model response of $w \cdot \phi(x, y, h)$.

Specifically, in multi-class problems, let $w = [w_1, ..., w_K]$, where K is the number of classes, and w_i stand for the model parameters of the i-th class. We can define the catenated feature vector as:

$$\tilde{\phi}(x, y, h) = [0_{D_1}, 0_{D_2}, ..., \phi(x, y, h), ..., 0_{D_K}], \tag{2.1}$$

where D_i is the length of w_i, $\phi(x, y, h)$ stands for the feature vector of example x regarding the y-th class model. Therefore, the model response can be written in a similar form as $w \cdot \tilde{\phi}(x, y, h)$.

The difference of existing latent variable models mostly lies in two phases of the training process. The first phase solves an inference over h on each training example. The process can also be regarded as a score function of example x_i on class $\hat{y}$ over current model parameters w. In the following, we denote the score function as $\Psi(x_i, \hat{y}, w)$. The second one solves parameter estimation over y, *i.e.*, the formulation of objective function $l(w)$ given all the training examples and ground-truth labels $\{(x_i, y_i)\}$.

2.1.1 Hidden conditional random field

The HCRF by Quattoni *et al.* [113] models the possible class labels y and latent variables h consistently by a conditional marginal model. The computation of conditional probability considers all possible values of the latent variables and class labels, integrated by a sigmoid function,

$$P(y, h|x; w) = \frac{\exp[w^T \phi(x, y, h)]}{\sum_{y,h} \exp[w^T \phi(x, y, h)]}. \tag{2.2}$$

The score function is defined as:

$$\Psi(x_i, \hat{y}, w) = \log \sum_h \exp[w^T \phi(x_i, \hat{y}, h)] \tag{2.3}$$

HCRF estimates the optimized parameters using a quasi-Newton method by solving the maximum likelihood hypothesis. The objective function is denoted as:

$$\begin{aligned} l(w) &= R(w) - L(w) = R(w) - \sum_{i=1}^{N} \log P(y_i|x_i; w) \\ &= R(w) - \sum_{i=1}^{N} \log \frac{\exp[\Psi(x_i, y_i, w)]}{\sum_{y \in \mathcal{Y}} \exp[\Psi(x_i, y, w)]}. \end{aligned} \tag{2.4}$$

where $R(w)$ is the regularization form. The gradient of $L(w)$ is further decomposed in terms of $P(h_j = a|x_i; w)$ and $P(y|x_i; w)$, which can be computed using belief propagation.

HCRFs have the same advantages and disadvantages as general CRFs. They perform well when there are enough training examples and when the model assumptions fit the data well. Given the efficient belief propagation approach, HCRFs are widely used in applications of gesture recognition [152] and object recognition [118]; in particular when the latent variables are highly related, forming a tree structure or other undirected graph structures.

2.1.2 Latent structural support vector machine

The LSSVM [167], opposite to HCRF, solves a joint maximum a *posteriori* (MAP) inference. It only considers the most likely latent state in computing the model response of training examples, and the most violated constraint when computing the objective function. The score function is denoted as:

$$\Psi(x_i, \hat{y}, w) = \max_{h \in \mathcal{H}} w^T \phi(x, \hat{y}, h). \tag{2.5}$$

The form of the objective function involves a user-specified loss function $\Delta(y_i, \hat{y}_i)$, which quantifies the gap between the correct output y_i and the esti-

mator $\hat{y}_i$. In particular,

$$\begin{aligned} l(w) = R(w) \quad &+ \quad \sum_{i=1}^{N} (\max_{y \in \mathcal{Y}} [\Psi(x_i, y, w) + \Delta(y_i, y)]) \\ &- \quad \sum_{i=1}^{N} \Psi(x_i, y_i, w). \end{aligned} \tag{2.6}$$

LSSVMs introduce a semi-convexity property in optimizing the objective function. Thus, they are usually solved by stochastic gradient descent or concave-convex procedure (CCCP). Due to the maximization operator in both the latent variables and possible class labels, LSSVMs requires less computational effort, and are more robust to noises, leading to numerous applications like object detection [178], human activity recognition [154], pose estimation [123], and discriminative motif finding [167]. However, the loss of convexity leads to local minima and requires careful designing of local search methods.

2.1.3 Marginal structured support vector machine

Ping *et al.* [107] proposed MSSVMs which properly account for the uncertainty of latent variables by using a marginal MAP predictor. MSSVMs remain the same $\Psi(x_i, \hat{y}, w)$ function with HCRFs, while designing the objective function $l(w)$ identical to LSSVMs. By combining (2.3) and (2.6) together, the final objective function of MSSVM is:

$$\begin{aligned} l(w) \quad &= \quad \sum_{i=1}^{N} \max_{y} \{\Delta(y_i, y) + \log \sum_{h} \exp[w^T \phi(x_i, y, h)]\} \\ &- \quad \sum_{i=1}^{N} \log \sum_{h} \exp[w^T \phi(x_i, y_i, h)] + R(w). \end{aligned} \tag{2.7}$$

The authors provide two approaches to solve the objective function of MSSVMs, including a sub-gradient descent algorithm and a CCCP algorithm. MSSVMs perform well in applications where there are large uncertainties in latent variables, and when the number of training examples is insufficient.

2.1.4 Latent support vector machine

The LSVMs by Felzenszwalb *et al.* [47] can be regarded as a special case of LSSVMs. The reason why LSVMs are discussed independently here is mainly due to the success of the Deformable Part-based Model - latent SVM (DPM-LSVM) framework in object detection, which encourages a series of extensions such as the kernel latent SVMs [165]. LSVM is designed for a binary classification task, *i.e.*, $\mathcal{Y} = \{-1, +1\}$. In the task of object detection, there are a large set of possible object position which is regarded as a latent variable in the model. LSVMs suppose that the set of latent variables relies on different training examples, *i.e.*, $h(x_i) \in \mathcal{Z}(x_i) \subset \mathcal{H}$. Therefore, in each iteration, the algorithm only stores the feature vectors for latent values whose model response scores exceed a given threshold. In that way the computational effort is much reduced. From the theoretical perspective, latent SVM is believed to be identical to multi-instance SVM [4].

With the form of

$$l(w) = R(w) + \sum_{i=1}^{N} \max(0, 1 - y_i \max_{h \in \mathcal{Z}(x_i)} w^T \phi(x_i, y, h)), \tag{2.8}$$

the objective function of LSVM is further decomposed into a concave part and a convex part, then optimized using a coordinate gradient descent approach.

2.1.5 Weak-label structural support vector machine

The weak-label structural SVMs (WL-SSVMs) [59] are designed for weakly-labeled data. For examples, in segmentation tasks, only bounding boxes or just the names of the object occurring in the image are given as weak labels. WL-SSVMs generalize SSVMs and LSSVMs to support weak training labels y_i together with strong predictions s, by introducing a relaxed loss $L_{output}(y_i, s)$. Since we focus on latent variable models in this paper, we will only review the WL-SSVMs with latent variables.

The score function of WL-SSVM follows the same formulation as LSSVMs, where a maximization operator is adopted to select the best scored latent value. The most notable difference appears in the form of the objective function, where

a surrogate training loss is defined in terms of two different loss augmented predictions:

$$\begin{aligned} l(x_i, w) &= \max_{s \in \mathcal{S}} [\Psi(x_i, s, w) + L_{margin}(s, y_i)] \\ &- \max_{s \in \mathcal{S}} [\Psi(x_i, s, w) + L_{output}(s, y_i)]. \end{aligned} \tag{2.9}$$

The marginal loss $L_{margin}(s, y_i) = \Delta(s, y_i)$, which corresponds to the user-specified loss in LSSVMs. The difference appears in the formulation of L_{output}. LSSVMs assigns $L_{output} = \mathbf{I}(s, y_i)$, where $\mathbf{I}(a, b) = 0$ when $a = b$, and $\mathbf{I}(a, b) = \inf$ when $a \neq b$. This loss function requires the predicting label s to be the identical with the training label y_i. On the contrary, L_{output} of WL-SSVMs relaxes the loss between weak labels y_i and strong predictors s, regarding s who are similar to the weak labels y_i as ground-truth labels under certain criteria, such as highly overlapped bounding boxes. Therefore, WL-SSVMs introduce more flexible definition of training labels, leading to applications including object detection and segmentation.

2.1.6 Epsilon-extension model

The discussions above show that the differences between these algorithms rely on the choice of *max* operator or *log-sum-exp* operator in the two training phases. In general, existing algorithms including HCRFs, LSSVMs and MSSVMs can all be regarded as special cases of a more general latent variable framework, which introduces a "temperature" parameter that smooths between *max* and *log-sum-exp* [109]. The ϵ-extension model was proposed by Schwing *et al.* [121], and further generalized by Ping *et al.* [107], who introduced separate ϵ parameters for latent variables h and predicted label y.

Let

$$\rho_h(f(\cdot)) = \epsilon_h \log \sum_h \exp(\frac{f(\cdot)}{\epsilon_h}), \tag{2.10}$$

and

$$\rho_y(f(\cdot)) = \epsilon_y \log \sum_y \exp(\frac{f(\cdot)}{\epsilon_y}). \tag{2.11}$$

Inspired by [109], (2.10) and (2.11) define a temperature function ρ which

Table 2.1: Relationship between latent variable models in the view of unified extension model. We use the same representation form as [107].

Model	$\epsilon_h \to 0^+(\max_h)$	$\epsilon_h = 1(\sum_h)$
$\epsilon_y \to 0^+(\max_y)$	LSSVM(LSVM)	MSSVM
$\epsilon_y = 1(\sum_y)$	**MLLR**	HCRF
$\epsilon_y = \epsilon_h \in (0,1)$	ϵ-extension model in [121]	

reduces to the *max* operator if $\epsilon \to 0+$, and becomes the *log-sum-exp* function if $\epsilon = 1$.

In the ϵ-extension model, the score of a training example is defined as:

$$\Psi(x_i, \hat{y}, w) = \rho_h(w^T \phi(x_i, \hat{y}, h)), \tag{2.12}$$

The objective function has the form of:

$$\begin{aligned} l(w) = R(w) \quad &+ \quad \sum_{i=1}^{N} \rho_y(\Delta(y_i, y) + \Psi(x_i, y, w)) \\ &- \quad \sum_{i=1}^{N} \Psi(x_i, y_i, w) \end{aligned} \tag{2.13}$$

The ϵ-extension model introduces a unified framework of existing latent variable algorithms as shown in Table 2.1. We will show later that the proposed algorithm MLLR can be also regarded as a special case of the ϵ-extension model with $\epsilon_y = 1$ and $\epsilon_h \to 0^+$.

2.1.7 Three-dimensional uncertainty model

The ϵ-extension model by Ping *et al.* [107] is so far the most general form in the area and provides a lot of insights. However, as discussed in Sec. 2.1.5, WL-SSVMs do not belong to the ϵ-extension model. In fact, following the idea of WL-SSVMs, we can define a more generalized form that measures the uncertainty from three dimensions, *i.e.*, latent variables h, ground-truth weak labels y, and strong

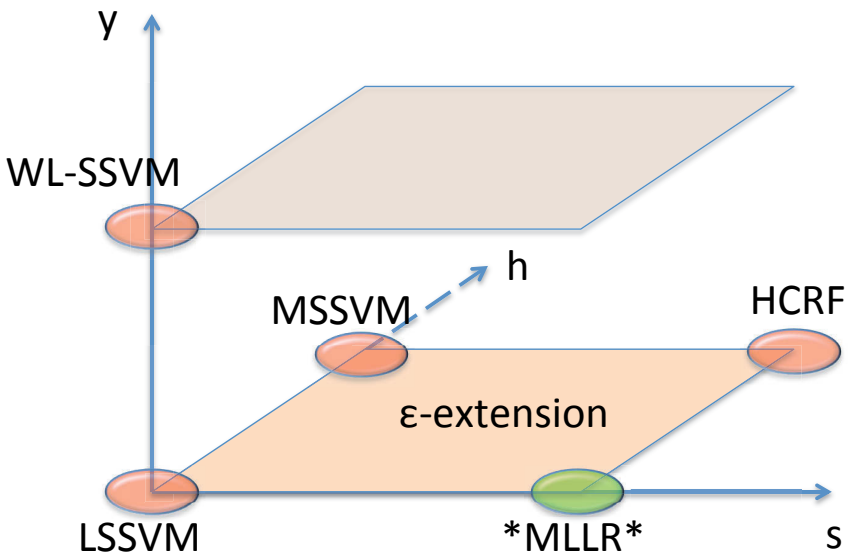


Figure 2.1: Illustration of latent variable models with structured prediction. The three dimensions are: h standing for latent variables, s standing for strong predictions and y standing for weak labels. The axes represent degree of uncertainty over the three dimensions.

predict labels s. The objective function of the three-dimensional uncertainty model can be derived by replacing (2.13) by:

$$\begin{aligned} l(w) &= R(w) + \sum_{i=1}^{N} \rho_y(\Psi(x_i, y, w) + L_{margin}(y_i, s)) \\ &- \sum_{i=1}^{N} \rho_y(\Psi(x_i, y, w) + L_{output}(y_i, s)) \end{aligned} \tag{2.14}$$

Figure 2.1 illustrates the connections between latent variable models with structured prediction.

2.2 Optimization methods

The objective functions of the latent variable models mentioned above are non-convex; some of them are even non-smooth, presenting much difficulty in the

training procedure. Most of the algorithms employ concave-convex procedure (CCCP) [169] to minimize the objective function, while the detailed inference operators and convex optimization solvers differ between SVMs and logistic functions, and various regularizers.

2.2.1 Concave-convex procedure

The concave-convex procedure (CCCP) [169] is a general non-convex optimization algorithm widely used in machine learning. The basic idea of CCCP is to rewrite the non-convex objective function into a sum of a convex function and a concave function (or equivalently a difference of two convex functions). By linearizing the concave part, the non-convex optimization problem is transformed into a sequence of convex sub-problems.

CCCP provides a straightforward paradigm for solving latent variable models. Take the ϵ-extension model in Section 2.1.6 as an example. The objective in (2.13) can be rewritten as:

$$l(w) = l^+(w) - l^-(w), \tag{2.15}$$

where

$$\begin{aligned} l^+(w) &= R(w) + \sum_{i=1}^{N} \rho_y(\Delta(y_i, y) + \Psi(x_i, y, w)), \\ l^-(w) &= \sum_{i=1}^{N} \Psi(x_i, y_i, w). \end{aligned}$$

The parameter vector is updated by minimizing a convex auxiliary function where the concave part $l^-(w)$ is linearized:

$$w^{t+1} \leftarrow \underset{w}{\operatorname{argmin}}((f^+(w) - w^T \Delta f^-(w^t)), \tag{2.16}$$

where $f^-(w^t) = \sum_i \mathbb{E}_{p(h|x_i,y_i)}[\phi(x_i, y_i, h)]$. $\mathbb{E}_{p(h|x_i,y_i)}$ denotes the expectation over the distribution $p(h|x_i, y_i)$. Computing the expectation equals to solving an inference problem over latent values. Possible approaches include:

- Belief propagation (BP). The (sub-)gradients of LSSVM and HCRF can be conducted using max-product BP and sum-product BP respectively. For MSSVM, Ping *et al.* [107] employed a mixed-product BP [94] to solve the loss-augmented marginal MAP prediction.

- Latent variable completion [167]. Regarding the maximum inference conducted in LSSVM and LSVM, the easiest while the most efficient way to solve the inference problem is to find the optimized latent assignments for positive examples and fix the assignments when optimizing the convex auxiliary function. Specifically, in (2.5), the linearized function is $w^T\phi(x_i, \hat{y}, h^*)$, where h^* is the optimized latent assignment according to the current model parameters w. The latent assignments h^* remain fixed to achieve a convex auxiliary function.

2.2.2 Convex optimization solver

After linearizing the concave part, the auxiliary objective functions become convex and therefore can be solved using standard convex optimization solvers. Existing latent variable models employ varied solvers, including stochastic gradient descent, cutting plane methods and quasi-Newton methods.

The most straightforward approach is stochastic gradient descent employed by Felzenswalb *et al.* in their implementation of Latent SVM [47]. The process involves the computation of a sub-gradient of the LSVM objective function as follows,

$$\Delta l(w) = w + C\sum_{i=1}^{N} g(w, x_i, y_i), \tag{2.17}$$

where

$$g(w, x_i, y_i) = \begin{cases} 0, \text{ if } y_i\Psi(x_i, y_i, w) \geq 1 \\ -y_i\phi(x_i, h_i(w)), \text{ otherwise} \end{cases}$$

where $\phi(x_i, h_i(w))$ is defined similar to (2.5).

In stochastic gradient descent, the sub-gradient is approximated using a single example at each iteration, *i.e.*, approximating $\sum_{i=1}^{n} g(w, x_i, y_i)$ with $ng(w, x_i, y_i)$. A learning rate of $\alpha_t = 1/t$ is used in the algorithm.

Although stochastic gradient descent approach is easy to implement, it is relatively inefficient, especially if there are many "easy" training examples which do not make much progress using gradient descent. A significant improvement is achieved in [60] by adopting the quasi-Newton method L-BFGS [93], which employs second-order derivatives to accelerate the training process. L-BFGS becomes the default training method in the latest DPM implementation [58].

Cutting plane methods are widely-used for training structural SVMs [69]. The main idea of cutting plane algorithms is to iteratively use the first-order Tyler approximation, *i.e.*, cutting plane, to bound the exact objective function; then find the optimal model parameters respect to all the previous added cutting planes. The algorithm terminates when the gap between the approximate and exact objective function falls below a given threshold. Furthermore, Teo *et al.* [136] proposed the Bundle Method for Regularized Risk Minimization (BMRM), which generalizes the cutting plane method and involves no parameters to tune. However, as argued in [136], the optimization process of bundle methods can be hindered by the "stalling" steps. For some steps, the bundle methods could not find a new optimal solution. As a result, the objective values are not strictly decreasing. As shown in [168], the BMRM algorithm does not properly decrease the function value and the norm of gradient for logistic regression on large datasets.

Most of the discussed methods are designed for SVMs and L2-regularizer. Due to the ability to obtain sparse models and thus be applied for feature selection, L1-regularized forms are also widely used in many areas. However, its non-differentiability causes more difficulties in training. Existing methods solving L1-regularizer for logistic regression include TRON (trust region Newton method) [88], CDN (coordinate descent using one-dimensional Newton steps) [23], active set methods [104] and quasi-Newton methods [3]. A thorough discussion can be found in [168], where extensive comparisons of the state-of-the-art software packages are presented in detail.

2.3 Weakly Supervised Learning

Weakly supervised learning is an effective way to scale learning algorithms by relieving the labeling burden by learning from simpler labels. In object recog-

nition tasks, weakly supervised algorithms can be employed in object localization and classification [32, 66, 130, 151, 177], co-segmentation and co-localization [70, 133, 146], semantic segmentation [108, 145, 162] and detection with mixture-component models [1, 43, 47].

Most of the algorithms solving weakly supervised learning problems employed multi-instance learning (MIL) methods [83, 100, 119]. First proposed in [39], MIL assumes that the labels are applied to "bags", in which a bag is labeled positive if at least one of the instances in this bag is positive, and negative when all of its instances are negative. In object recognition, this framework allows localization and classification to benefit from each other. Specifically, the Latent SVM used in DPM framework [47] is in fact identical to multi-instance SVM [4].

Different from supervised learning where the training labels are complete, weakly labeled learning needs to infer the integer labels of training examples, resulting in difficult mixed-integer programming (MIP). Most MIL algorithms start from an initialization and perform some form of local minimization. Examples include alternating optimization [170], in which the optimization is conducted alternatively on one variable when the others are keeping fixed; and constrained convex-concave procedure (CCCP) [169], in which the non-convex objective function is decomposed into a difference of two convex functions. Although the non-convex optimization approaches are usually efficient, they are relatively sensible to initialization; thus they are easily to be get stuck in local minima.

Considering the importance of initialization in non-convex optimization approaches, early attempts [31, 51] focused on datasets with strong object-in-the-center biases. Since images in such datasets usually contain only one object that is centered and fills much space of the image, weakly supervised learning in this scenario allows the detected object to move around flexibly, albeit less freely than that in an object detector. Recent work [128, 129] attempted to learn classifiers from much more challenging datasets such as PASCAL VOC [45]. Some efforts have been made to carefully design the initialization heuristics, such as [116]. Nonetheless, a better initialization strategy still remains an open issue.

Another idea to solve the non-convex objective function is to produce convex relaxations. Li *et al.* [87] proposed a WEakly LabeLed SVM (WellSVM) via a label generation strategy. WellSVM maximized the margin by generating

the most violated label vectors iteratively, and then combined them via efficient multiple kernel learning techniques. As a result, WellSVM was formulated as a convex relaxation of the original non-convex objective, and involved a series of SVM sub-problems which can be solved by efficient standalone SVM solvers.

Although most of the existing weakly supervised learning algorithms rely on multi-instance learning methods, there are some novel attempts to solve this problem in other ways. Cabral *et al.* [19] employed the additive nature of histogram features and formulated weakly supervised image classification as a low-rank matrix completion problem. The method was convex and robust to labeling errors, background noise and partial occlusions.

Mid-level visual element discovery [41] is an unsupervised method that automatically discovers patches with higher semantic levels than "visual words". They are both representative, *i.e.*, frequently occurring within a visual dataset, and visually discriminative. This idea is widely used in FGVC methods to discover object parts.

2.4 Webly Supervised Learning Approaches

Web data have long been a main supplier to acquire object recognition databases. The construction of modern datasets, such as ImageNet [36] and Microsoft COCO [91], usually involves a manually cleaning process after collecting a large number of web images to ensure the correctness of image labels. Considering the extensive effort required to label even larger numbers of images, it is drawing an increasing interest to learn directly from labels and images acquired from the Internet.

Due to the complexity of learning tasks requiring higher supervision in web scale. Recently, several works [29, 40] have been proposed to perform attribute learning on specific concepts, which aims to extract significant or interesting patterns. We mainly focus on methods on classification in this paper.

For previous work on image classification [14, 28, 120], web-scale learning can be roughly classified into two categories: the "filtering" approaches and the "grouping" approaches. In particular, the filtering approaches [50, 63, 86, 153] first obtain a large pool of images from image search engines, then perform a filtering operation to remove noises and discover visual concepts. Usually im-

plemented as an unsupervised learning method, the main issue of this kind of methods is to provide clusters pruned from outliers [63], *i.e.*, to model both the intra-class variance and irrelevant images returned by search engines. However, the learned concepts are highly dependent on the gathered data, which is easily to be biased [98].

On the contrary, the "growing" approaches [33, 48, 84, 161] first generate a small group of labeled seed images, then grow the dataset from the starting seeds by performing some kinds of iterative "self training" [159]. Although the semi-supervised problem setting provides a way to combat both noise and data bias, in practice it is tricky to achieve these two goals simultaneously. Intuitively, one needs to prevent over specialised results biased towards the seed images. However, too much generalization will possibly lead to semantic drift which forces the learned concepts to move too far away from the initial seeds.

Recently, several algorithms [29, 40] have been proposed to perform attribute learning on specific concepts, which aims to extract significant or interesting patterns. However, these methods are out of our scope as we mainly focus on methods on classification in this thesis.

2.5 Fine-Grained Visual Categorization

Fine-grained visual categorization is one of the latest and most challenging object recognition tasks. Different from basic-level object classification where the objects belong to highly discriminative concepts such as dogs, chairs, and cars, fine-grained categorization aims to distinguish objects in the subordinate level. Typical examples include different subspecies of animals [13, 73, 103, 150], plants [78, 101], and man-made objects [97, 144, 164]. We consider fine-grained visual categorization as one of the most important applications of the proposed MLLR in this thesis. The most predominant reason is that there are rich inter-class relationships between subordinate categories, making it preferable for conducting multi-class predictions and explaining the results probabilistically.

In the last few years, the performance of fine-grained visual categorization has been increasing steadily, where the developments mainly come from the following aspects.

2.5.1 Feature representation.

Early attempts on FGVC mainly aimed at learning more discriminative feature descriptors. Using traditional feature representations such as SIFT [95] and HOG [35], Bo *et al.* [15] introduced three types of matching kernels to measure similarities between image patches, and showed that kernel descriptors outperformed sophisticated features including SIFT and deep belief networks by turning any type of pixel attributes into patch-level features. Sanchez *et al.* [117] showed that by adopting the Fisher Vector (FV) [105] which included higher-order information than standard bag-of-visual-words (BOV) as the feature embedding method, superior results were achieved for FGVC tasks.

Recently, the traditional object recognition framework is greatly challenged by the rise of deep learning algorithms [57, 76, 122], which have achieved superior performance on varied visual understanding tasks. It has been proved that deep feature representations trained on large-scale datasets can be easily generalized to other object recognition tasks [114]. Not surprisingly, the latest FGVC methods mostly adopted the highest performing CNN architectures up-to-date as the feature representations, such as AlexNet [76], VGGNet [127] and GoogleNet [132]. A more promising choice is to incorporate the features, models and learning of CNNs in a unified end-to-end system specifically designed for FGVC problems. A lot of recent works exemplified this strategy [92, 174].

2.5.2 Model design

Except for the rather generic feature engineering approaches, researchers also developed a set of specific strategies for solving FGVC problems based on its unique characteristics. Examples include segmentation based methods, part based methods, and alignment based methods.

Foreground/background segmentation is a fundamental problem in object recognition. For fine-grained visual categorization, corroborative evidences are presented to show that exploiting foreground/background segmentation can improve the accuracy considerably [20, 101], since the discriminative power of subordinate categories mostly lie on detailed information of foreground objects.

Part-based methods are arguably the most widely used strategy in FGVC

approaches [62, 139, 172]. For example, Zhang *et al.* [172] proposed a part-based R-CNN method that trained a CNN model specifically on each of the object parts. As a result, the resultant CNN representations carried much higher discriminative power than features trained on the whole object bounding boxes. Meanwhile, by adopting strong supervision including object part landmarks, Berg *et al.* [12] presented a method that learned a set of part-based one-vs-one features (POOFs) that each of which specialized in discrimination between two particular classes based on the appearance of a particular part.

Motivated by face recognition methods, pose alignment [17,46,55] is a common trick used in many FGVC approaches. In particular, pose alignment methods either refined the objects to adjust the overall shape of a template image [55], or proposed a pose-normalized appearance model [46] that enabled a fair comparison of features extracted from various poses.

It is worth to be noted that the strategies discussed above are not mutual exclusive. Very recently, Lin *et al.* [89] put all three strategies together and proposed a framework called Deep LAC (Localization, Alignment and Classification).

2.5.3 Training supervision

Most of the aforementioned FGVC methods depended on strongly supervision, which means that the methods used detailed annotations during training to support the explicit extraction of powerful features. However, the requirement of strong supervision poses a problem for scaling up fine-grained recognition to an increasing number of domains.

Human-in-the-loop methods [18, 37, 142, 149] provided an effective alternative for removing the requirement of strongly supervised datasets. Nonetheless, due to the need of human interaction, these methods also could not scale up to larger systems with thousands or millions of training samples.

Weakly supervised FGVC algorithms [74, 92, 126, 160] were proposed recently to cope with the scaling problem. Most of them adopted part discovery approaches that enabled the algorithm to select object parts using data driven methods. Krause *et al.* [74] presented an FGVC method without using part annotations given the observation that objects in a fine-grained class shared a high

degree of shape similarity, allowing them to be aligned via segmentation alone. Similarly, Simon & Rodner [126] presented an approach that learned part models in a completely unsupervised manner by finding constellations of neural activation patterns computed using convolutional neural networks. Our weakly supervised FGVC method in this thesis, on the contrary, is based on the standard multi-instance learning methods [119] which are widely used in weakly supervised object recognition approaches, instead of performing object part discovery as discussed in the aforementioned works.

Another interest of the FGVC community currently is to employ webly supervised methods [28, 40] that rely on the Internet which offers a nearly endless supply of images with human generated labels or tags. Although labels crawled from the Internet would be very noisy, Krause *et al.* [75] proved that by training on publicly-available noisy web image search results, even higher accuracies could be achieved than current state-of-the-art algorithms without using any expert-annotated training data, while scaling to over ten thousand fine-grained categories. We also investigate webly supervised methods for FGVC in this thesis. However, our observation is that existing strongly supervised datasets can offer a reliable initialization strategy to remove noise in web images and also introduce detailed part-level annotations. This results in better performance than using noisy web images only for training.

Chapter 3

Multinomial Latent Logistic Regression

In this chapter, we will first introduce the proposed paradigm **Multinomial Latent Logistic Regression** (MLLR), including *what is MLLR* (the derivation of its objective function), *how to solve MLLR* (analysis on its concave and convex property and optimization methods), and *what MLLR can do* (possible forms of latent variables and several typical weakly supervised problems). Meanwhile, we will provide a thorough investigation of the difference and connection between the proposed algorithm and existing paradigms of latent variable models with structured outputs, which could be helpful for researchers to choose a proper latent variable algorithm under various occasions. Based on these analysis, further issues for conducting MLLR in practical applications will be discussed in the sections later.

3.1 Introduction

Latent variable models are widely used in diverse research fields such as computer vision, natural language processing, speech analysis and public health. By modeling hidden information that plays an important role in statistical modeling as latent variables and inferring them from the observed information, latent variable models extract additional information than standard supervised learning

paradigms and thus have the potential to achieve superior performance.

Among various kinds of latent variable models, algorithms that are designed for multi-class outputs or structured predictions are a special series particularly effective in object recognition and natural language processing domains. For example, in object recognition applications, very often one needs to distinguish an object class from a large set of concepts, such as face recognition and generic object detection, which inheritably results in multi-class problems. Meanwhile, modeling the context between multiple words in a sentence can be naturally regarded as a structural problem. In the literature, hidden conditional random fields (HCRFs) [113] and latent structural SVMs (LSSVMs) [167] are perhaps the most notable algorithms that attempt to address latent variable problems with structured predictions, with successful applications including gesture recognition [152], object recognition [118], object detection [178] and link prediction [163].

Approaches such as HCRFs and LSSVMs have their own advantages and disadvantages. Specifically, generalizing the support vector machines (SVMs), LSSVMs adopt a jointly maximum a *posteriori* procedure by treating the inference on latent variables as a parameter selection problem and using a max-margin strategy to formulate the objective function. Such algorithms are robust to problems posed by insufficient training data, especially when the latent variables have a complicated structure and cannot be represented in a graph model effectively [155]. In contrast, from the perspective of probabilistic learning, HCRFs treat both the possible class labels and latent variables marginally. As a result, they perform well when a large uncertainty exists over the training data [107]. However, marginalization is much more difficult in applications in which a well-defined probabilistic graphical model for latent variables is not available. In such applications, it would be of benefit to maximize rather than marginalize over the latent variables. Therefore, one should pick up a proper algorithm based on the their practical needs in real-world applications.

In this thesis, we present a new paradigm termed Multinomial Latent Logistic Regression (MLLR) by introducing latent variables to the Regularized Multinomial Logistic Regression (RMLR). In particular, MLLR performs "averaging" over possible class labels and "maximizing" over possible latent assignments; thus enjoys several merits in certain scenarios inheriting the advantages of lo-

gistic regression, including effective probabilistic analysis and natural multi-class extension. In addition to multi-class classification applications, MLLR can also be generalized to general structured output prediction, specializing the extended ϵ-extension model described in Section 2.1.6.

In the following sections, we will first derive MLLR from the regularized multinomial logistic regression (RMLR) to solve multi-class classification problems. Detailed analysis will be conducted on the objective function, optimization methods, convergence analysis of the proposed algorithm. Furthermore, by generalizing MLLR to structured outputs, we will present discussions of the connection and difference between MLLR and other latent variable models with structured outputs from both theoretical and experimental perspective.

3.2 Multinomial Latent Logistic Regression

The proposed multinomial latent logistic regression (MLLR) introduces latent variables into the regularized multinomial logistic regression algorithm. Specifically, MLLR performs a "maximization" operator over possible latent values and a "summation" operator over possible class labels. By modeling the optimization of MLLR as a concave-convex procedure, we introduce two approaches to optimize the convex part in the objective function, including a gradient descent approach similar to LSVMs, and a new version of coordinate descent method using one-dimensional Newton direction (CDN) [168], which involves a recomputing scheme for latent variables in the line search procedure.

3.2.1 Multinomial logistic regression

Given a training set of $S = \{(x_1, y_1), ..., (x_N, y_N)\}$ where $y_i \in \mathcal{Y} = [1, 2, ..., K]$. The multinomial logistic regression has been developed as a result of the desire to model the posterior probabilities of the K classes via linear functions in x. The posterior probability has the form

$$p_k(x; w) = Pr(Y = k|X = x) = \frac{\exp(w_k^T x)}{\sum_{l=1}^{K} \exp(w_l^T x)}, \tag{3.1}$$

where w_k are parameters of the model of the k-th class, with a dimension of D_k. We denote the entire parameter set $w = \{w_1, ..., w_K\}$.

Logistic regression models are typically fitted by maximizing the likelihood function, using gradient-based methods such as the Newton-Raphson algorithm. The objective function for N training examples is:

$$L(w) = -\sum_{i=1}^{N} \log p_{y_i}(x_i; w). \tag{3.2}$$

3.2.2 Latent variables

MLLR introduces latent variables into the logistic regression paradigm, where each input example x is associated with a latent variable h. Let $\phi(x, y, h)$ be the feature vector of an example x depending on the latent variable h and class y, where $h \in \mathcal{H}$ and the set $\mathcal{H}$ defines all the possible latent variable assignments.

Various applications can be explained under this assumption. For example, when detecting objects by parts, the latent variable h can be modeled as a possible location for each part. $\phi(x, y, h)$ becomes the feature vector of an example x given the location of each part fixed at h. In an application of language modeling such as parsing sentences, x is the input sentence, while $\mathcal{H}$ is the set of all possible parse trees and $\phi(x, y, h)$ is the feature vector for a sentence-tree pair.

Consider a score function of the form

$$\Psi(x; w_k) = \max_h w_k^T \phi(x, k, h). \tag{3.3}$$

The score is obtained by finding the optimal latent assignment h that gives the highest score to the example x given a model w_k.

We follow the terms of "positive" and "negative" examples in multi-class classification settings. An example x is called positive example with respect to model w_k if the label $y = k$, and called negative example with respect to model w_k if the label $y \neq k$. Therefore, an example becomes a positive example only when the classifier with respect to the example's true label is considered.

From (3.3), we can get

$$h(w_k) = \underset{h}{\operatorname{argmax}}[w_k^T \cdot \phi(x, k, h)]. \tag{3.4}$$

Analogous to standard logistic regression, we rewrite the posterior probability of the k-th class given an example x as

$$p_k(x; w) = \frac{\exp[\Psi(x; w_k)]}{\sum_{l=1}^{K} \exp[\Psi(x; w_l)]}. \tag{3.5}$$

Since the parameter space has a large volume, MLLR will be easy to overfit to the training data. To avoid the overfitting problem, we introduce a lasso regularizer, which tends to gain sparse representation. Further dicussions on the choice of the regularizer will be demonstrated in Section 3.3. The log-likelihood function becomes

$$\begin{aligned} l(w) &= L(w) + R(w) \\ &= -C \sum_{i=1}^{N} \log p_{y_i}(x_i; w) + \sum_{l=1}^{K} |w_l| \\ &= -C \sum_{i=1}^{N} \log \frac{\exp[\Psi(x_i; w_{y_i})]}{\sum_{l=1}^{K} \exp[\Psi(x_i; w_l)]} + \sum_{l=1}^{K} |w_l| \\ &= \underbrace{-C \sum_{i=1}^{N} \Psi(x_i; w_{y_i})}_{A} \underbrace{+C \sum_{i=1}^{N} \log \sum_{l=1}^{K} e^{\Psi(x_i; w_l)}}_{B} \underbrace{+ \sum_{l=1}^{K} |w_l|}_{C}, \end{aligned} \tag{3.6}$$

where part A measures the effect of positive examples; part B is a divisor considering all examples in all models; part C is the lasso regularizer term. The constant C controls the relative weight of the regularization term. In practice, C will be set empirically or according to cross validation.

The training process can be explained from two perspectives. In the local view, each example eagerly finds the K optimized latent configurations for all models regardless of which class the example belongs to. In the global view, the

models are refined to give higher scores to positive examples and lower scores to negative examples, *i.e.,* maximize the conditional probability.

3.2.3 Concave-convex procedure

Minimizing the objective function of the standard logistic regression model in (3.2) leads to a convex optimization problem. Note that if we discard the latent variable setting in (3.6) and redefine $\Psi(x, w_k) = \Psi^{lr}(x; w_k) = w_k^T x$, the likelihood function (3.2) can be written in the same form as (3.6).

However, when latent variables are introduced to the objective function, the function is no longer convex. In particular, part C remains unchanged. Part B is still convex since it involves only convex functions (log-sum-exp). However, part A becomes the negative of a maximum of a set of linear function with respect to w, which is a concave function.

A similar situation arises in relation to Latent Structural SVM [167] and LSVM [47]. The former algorithm uses the Concave-Convex Procedure (CCCP) to optimize loss function [169], while the latter introduces a semiconvexity property and solves the problem by coordinate gradient descent.

MLLR can be also solved in a CCCP framework. We define an auxiliary function that bounds the exact objective function by fixing the latent variable for each positive example. In particular, the auxiliary function is defined as:

$$\begin{aligned} l(w, H_p) &= C \sum_{i=1}^{N} \log \sum_{l=1}^{K} \exp[\Psi'(x_i; w_l)] \\ &- C \sum_{i=1}^{N} \Psi'(x_i; w_{y_i}) + \sum_{l=1}^{K} |w_l|, \end{aligned} \tag{3.7}$$

where

$$\Psi'(x_i; w_l) = \begin{cases} w_l^T \phi(x_i, y, h_i^*), & y_i = l. \\ \Psi(x_i; w_l), & y_i \neq l. \end{cases}$$

and $H_p = \{h_i^*, i = 1, ..., N\}$ is a set of optimal latent assignments for all the N positive training examples.

It is noted that

$$l(w) = \max_{H_p} l(w, H_p). \tag{3.8}$$

The auxiliary function bounds the exact likelihood function by linearlarizing the non-convex part A into a linear function in w. In [167], the procedure is called a "latent variable completion" problem. Since $l(w, H_p)$ is a convex function, we minimize $l(w)$ using CCCP as follows (Algorithm 1).

1. *Optimize positive examples*: Optimize $l(w, H_p)$ over H_p. For each example x_i, find the optimized latent value for its respective model w_{y_i} by (3.3), $h_i^* = \text{argmax}_h\, w_{y_i} \cdot \phi(x_i, y_i, h)$ and set $H_p = \{h_i^*, i = 1, ..., N\}$.

2. *Optimize model parameters* w. Optimize the convex function $l(w, H_p)$ over w and all possible latent values for negative examples.

Algorithm 1 Concave-convex procedure for training MLLR

Input: Training examples $\{(x_1, y_1), ..., (x_N, y_N)\}$,
Initial latent variables for all training examples.
Output: Model parameters w.
for outerLoop:=1 **to** numOuterLoop **do**
{Solve the axillary function $l(w, H_p)$ with $H_p = \{h_i^*\}$ fixed}
Choice 1. gradient descent (Section 3.2.4)
Choice 2. latent-CDN (Section 3.2.5)
{Relabel latent variables for positive examples}
for i:=1 **to** N **do**
Optimize $h_i^* = \text{argmax}_h\, w_{y_i} \cdot \phi(x_i, y_i, h)$.
end for
end for

Recall that in Section 3.2.2, we argue that the training process needs to iteratively find optimal latent values and updates model parameters according to logistic likelihood function. However, the CCCP here has to fix the latent values for positive examples to preserve the convexity of $l(w, H_p)$ when updating model parameters. Fortunately, in classification tasks, the number of positive examples is much less than the number of negative examples. After convergence, the algorithm has reached a strong local optimum considering an exponentially large space of model parameters and latent values for negative examples. Still the initialization of w should be carefully designed to avoid bad local minima.

3.2.4 Gradient descent

Similar to [47], the axillary objective function $l(w, H_p)$ can be optimized using a gradient descent method. Although the lasso regularizer term is non-differentiable, we can compute a subgradient of (3.7) with respect to $w_k, k = 1, ..., K$ as:

$$\begin{aligned} \nabla l(w_k) &= C \sum_{i=1}^{N} \phi(x_i, k, h_i(w_k)) \frac{\exp[\Psi'(x_i; w_k)]}{\sum_{l=1}^{K} \exp[\Psi'(x_i; w_l)]} \\ &- C \sum_{y_i=k} \phi(x_i, k, h_i(w_k)) + sgn(w_k), \end{aligned} \tag{3.9}$$

where $sgn(\cdot) \in \{-1, 1\}$ is a sign operator.

In fact, (3.9) can be rewritten as:

$$\nabla l(w_k) = C \sum_{i=1}^{N} \phi(x_i, k, h_i(w_k)) \cdot q(x_i, w_k) + sgn(w_k), \tag{3.10}$$

where

$$q(x_i, w_k) = \begin{cases} p_k(x_i; w) - 1 & , x_i \text{ is positive for class } k. \\ p_k(x_i; w) & , x_i \text{ is negative for class } k. \end{cases}$$

The gradient descent procedure iteratively updates model parameters and latent variables for negative examples as follows,

1. In the $(t+1)$-th iteration, for all training examples x_i and all class models w_k, let $h_i(w_k^{(t)}) = \text{argmax}_h\, w_k^{(t)} \cdot \phi(x_i, k, h)$, if $y_i \neq k$, and $h_i(w_k^{(t)}) = h_i^*$ if $y_i = k$, where $h_i^* \in H_p$.

2. For all class models, set $w_k^{(t+1)} = w_k^{(t)} - \alpha_t \cdot [C \sum_{i=1}^{N} \phi(x_i, k, h_i(w_k^{(t)})) \cdot q(x_i, w_k^{(t)}) + sgn(w_k^{(t)})]$.

The form of $q(x_i, w_k)$ has a clear probabilistic explanation. Similar to the perceptron algorithm, the gradient descent method repeatedly pushes the model w_k towards positive examples and far away from negative examples. By adding a probabilistic multiplier, the algorithm assigns a bigger penalization on "hard negative" where $p_k(x_i; w)$ is large. For positive examples, "hard positive" indicates an example that has a smaller probability regarding the current model, which

Algorithm 2 Gradient Descent Algorithm

Input: Training examples $\{(x_1, y_1), ..., (x_N, y_N)\}$,
Latent variables for positive examples $H_p = \{h_i^*\}$.
Output: Model parameters w.
for t:=1 **to** numGradientDescentLoop **do**
{Relabel latent variables}
for i:=1 **to** N **and** k:=1 **to** K **and** $y_i \neq k$ **do**
Optimize $h_i(w_k^{(t)}) = \text{argmax}_h \, w_k^{(t)} \cdot \phi(x_i, k, h)$.
end for
{Update model parameters}
for k:=1 **to** K **do**
Update $w_k^{(t+1)} = w_k^{(t)} - \alpha_t \cdot \nabla l(w_k^{(t)})$.
end for
end for

plays a more important role in updating the model parameters.

The training procedure is outlined in Algorithm 2.

3.2.5 Coordinate descent using one-dimensional Newton directions with latent variables

Different from the gradient descent method discussed before, Yu *et al.* [69] used an improved version of cutting plane algorithm to solve the objective function of LSSVMs. The main idea of cutting plane algorithms is to iteratively use the first order Tyler approximation, *i.e.,* cutting plane, to bound the exact objective function; then find the optimal model parameters with respect to all the previous added cutting planes. The algorithm terminates when the gap between the approximate and exact objective function falls below a given threshold. Furthermore, Teo *et al.* [136] proposed the Bundle Method for Regularized Risk Minimization (BMRM), which generalizes the cutting plane method and involves no parameters to tune.

However, as argued in [136], the optimization process of bundle methods can be hindered by the "stalling" steps, which means that for some steps, the bundle methods could not find a new optimal solution. As a result, the objective values are not strictly decreasing. As shown in [168], the BMRM algorithm fails to prop-

erly decrease the function value and the norm of gradient for logistic regression on large datasets.

We propose a new algorithm that efficiently solves the axillary objective function $l(w, H_p)$ in MLLR using second-order derivatives. Our optimization algorithm is based on the coordinate descent method using one-dimensional Newton directions (CDN) proposed by Yuan *et al.* [168]. The CDN algorithm is effective regarding the characteristics of logistic functions and the L1-regularization term. To deal with the latent variables, we add a step of recomputing optimal latent assignments in the line search process of CDN. As a result, the new algorithm termed latent-CDN ensures the objective value to be strictly decreasing, while not imposing excessive computational efforts.

In detail, given current model parameters $w^{(t)}$, a coordinate descent method updates one variable $w_{k,j}^{(t)}$ at a time, where $w_{k,j}^{(t)}$ stands for the j-th variable in the k-th class model. CDN uses one-dimensional Newton direction to accelerate the local convergence. However, since the L1-regularization term is not differentiable, we employ the second-order approximation of the loss term $L(w)$, and find the optimal step z by solving

$$\min_z g_{k,j}(z) = |w_{k,j}^t + z| - |w_{k,j}^t| + L'_{k,j}(0)z + L''_{k,j}(0)z^2, \tag{3.11}$$

where

$$L_{k,j}(z) \triangleq L(w^{(t)} + ze_{k,j}). \tag{3.12}$$

The problem has a closed-form solution:

$$d = \begin{cases} -\frac{L'_{k,j}(0)+1}{L''_{k,j}(0)} & \text{, if } L'_{k,j}(0) + 1 \leq L''_{k,j}(0)w_j^{(t)} \\ -\frac{L'_{k,j}(0)-1}{L''_{k,j}(0)} & \text{, if } L'_{k,j}(0) - 1 \geq L''_{k,j}(0)w_j^{(t)} \\ -w_{k,j}^{(t)}, & \text{, otherwise.} \end{cases} \tag{3.13}$$

For MLLR, the first-order and second-order derivatives are:

$$L'_{k,j}(0) = C \sum_{i=1}^{N} q(x_i, w_k)\phi_j(x_i, k, h_i(w_k)),$$
$$L''_{k,j}(0) = C \sum_{i=1}^{N} p_k(x_i; w)(1 - p_k(x_i; w))\phi_j^2(x_i, k, h_i(w_k)), \tag{3.14}$$

where $\phi_j(x, k, h) = \phi(x, k, h)e_j$.

Since (3.11) is a quadratic approximation of $l(w^{(t)} + ze_{k,j}) - l(w^{(t)})$, and due to the non-smooth regularization term, the Newton direction d does not guarantee to decrease the objective value. Instead, Tseng *et al.* [138] used a line search procedure to find a shrinking parameter $\lambda \in (0, 1)$, such that the step λd satisfies a modified sufficient decrease condition:

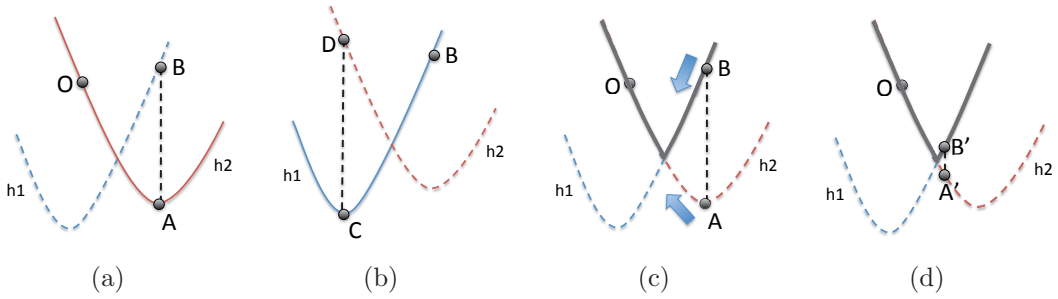
$$\begin{aligned} l(w^{(t)} + ze_{k,j}) - l(w^{(t)}) &= g_{k,j}(\lambda d) - g_{k,j}(0) \\ &\leq \sigma\lambda(L'_{k,j}(0)d + |w^{(t)}_{k,j} + d| - |w^{(t)}_{k,j}|). \end{aligned} \tag{3.15}$$

To find λ, a backtrack line search is adopted in CDN, which checks $\lambda = 1, \beta, \beta^2, ...$, where $\beta \in (0, 1)$, until the condition (3.15) is reached.

The previous steps are nearly identical to the CDN algorithm in [168]. However, a significant property of MLLR is that once the model parameters w are changed, the algorithm needs to recompute all the free latent variable assignments. Since the line search procedure involves updating w, it is problematic if we keep the previous latent assignments unchanged.

Considering the following toy example where two simulated objective functions f_{h_1} and f_{h_2} for two latent values h_1, h_2 are shown in Figure 3.1. The goal is to find the minimum value of the final objective function f_h, which is denoted as $f_h = \max(f_{h_1}, f_{h_2})$. Here the line search procedure finds optimal objective value with respect to f_{h^*}, while the latent updating process selects h^* as h_1 or h_2 according to the current objective value.

Suppose that point O indicates the initial state, where function f_{h_2} is activated. As shown in Figure 3.1(a)(b), if the latent assignments are not updated in the line search process, the algorithm will find the optimal solution according to



(a) (b) (c) (d)

Figure 3.1: Illustration for the importance of updating latent assignments when performing line search in CDN. Figure (a)(b) shows the situation where no updating process is performed, while (c)(d) shows the situation after updating latent assignments.

f_{h_2}, resulting in point A. However, the actual objective value now should be B, since $f_{h_1}(B) > f_{h_2}(B)$. After line search, the optimized value is updated to B in the latent assignment process, and then starts the next line search session. Again the algorithm computes the Newton direction and finds an optimal solution, indicated by C. This time the respective latent value for C turns to h_2, and the actual objective value is D. Following this kind of iteration, the algorithm gets stuck in a bistable state that repeatably finds B and D as the optimal objective value. Apparently, they are far away from the global optimal.

In the latent-CDN algorithm, we solve this problem by introducing an updating process of latent assignments to line search procedure. As shown in Figure 3.1(c)(d), when point A is reached, the modified algorithm notices that the actual objective is B, which does not satisfy the sufficient decrease condition. Thus the algorithm iteratively shrinks the step λd until a point B' satisfies the sufficient descent condition.

One possible drawback of the latent re-assigning process is that it could introduce a high computational effort. To update the latent assignments, one needs to compute all the score functions $s(x_i; w_k)$ over possible latent variables. However, since CDN only updates one variable $w_{k,j}$ at a time, the computation can be drastically decrease by re-using the previous scores. The new latent assignments

Algorithm 3 Latent CDN Algorithm

Input: Training examples $\{(x_1, y_1), ..., (x_N, y_N)\}$.
Latent variables for positive examples $H_p = \{h_i^*\}$.
The initial model parameters w^0.
Select $\beta \in (0, 1)$ and a random permutation P.
Output: Model parameters w.
for t:=1 **to** numCDNLoop **do**
for i:=1 **to** $\sum_k D_k$ **do**
Pick up a random variable $w_{k,j}$ according to P_i.
Calculate the Newton direction by (3.13).
{Line search procedure}
for λ:=$1, \beta, \beta^2, ...$ **do**
Check if the sufficient decreasing condition (3.15) is satisfied.
Update all latent variable assignments $h_i(w_k^{(t)})$ by (3.16).
If (3.15) is satisfied, terminate line search.
end for
Update $w_{k,j}^{(t+1)} = w_{k,j}^{(t)} + \lambda d$.
end for
end for

have the form of:

$$\begin{aligned} h_i^{new}(w_k^{(t)}) &= \underset{h}{\text{argmax}}[\Psi^{new}(x_i; w_k^{(t)})] \\ &= \underset{h}{\text{argmax}}\{\Psi^{old}(x_i; w_k^{(t)}) + w_{k,j}[\phi_j(x_i, k, h) - \phi_j(x_i, k, h_i^{old}(w_k^{(t)}))]\}, \end{aligned} \tag{3.16}$$

where only one numerical multiplication is required for a basic latent variable updating step.

Two implementation techniques are conducted in CDN to improve the convergence speed, including a random permutation on the one-dimensional subproblem, and a shrinking technique that heuristically remove redundant model variables. If $w_{k,j} = 0$ and $-1 \leq L'_{k,j}(0) \leq 1$, then the new $w_{k,j}^* = 0$. We follow the same techniques, and the resulting latent-CDN algorithm is demonstrated in Algorithm 3. We will show in the experiments that the latent-CDN algorithm can converge faster than BMRM by exploiting second-order Newton directions.

3.2.6 Generalization to Structured Outputs

MLLR can be generalized to solve structured output predictions. Consider the same problem statements for structured predictions in Section 2.1, where w denotes the model parameters. The objective function of MLLR involves a maximization over h, a log-likelihood function, and a lasso regularizer; it can be therefore be rewritten as:

$$\begin{aligned} l(w) = |w| \quad &+ \quad C\sum_{i=1}^{N} \log \sum_{y} \exp\big\{ \max_{h}[w^T\phi(x_i, y, h)]\big\} \\ &- \quad C\sum_{i=1}^{N} \max_{h}[w^T\phi(x_i, y_i, h)]. \end{aligned} \tag{3.17}$$

The multi-class problem methods discussed in the previous sections can be adopted for structured outputs with only minor modifications. Specifically, the calculation of derivatives in (3.10) and (3.13) involves belief propagation and results in more complex formulations.

Consider the first- and second-order derivatives of the logistic function $L(w)$ in structured output formulation. The calculation of derivatives follows the formulation of belief propagation:

$$\begin{aligned} \nabla L_i(w) &= \sum_{y} p_y(x_i; w)[\phi(x_i, y, h^*_{i,y}) - \phi(x_i, y_i, h_i)], \\ \nabla^2 L_i(w) &= \sum_{y} \Big\{ [\phi(x_i, y_i, h^*_{i,y}) - \phi(x_i, y, h_i)] p_y(x_i; w) \\ &\cdot \sum_{y'} [\phi(x_i, y, h^*_{i,y}) - \phi(x_i, y', h^*_{i,y'})] p_{y'}(x_i; w) \Big\}, \end{aligned} \tag{3.18}$$

where $p_y(x_i; w)$ is computed by (3.21), optimal latent assignments are $h^*_{i,y} = \text{argmax}_h[w^T\phi(x_i, y, h)]$, and $\nabla^m L(w) = \sum_{i=1}^{N} \nabla^m L_i(w), m = 1, 2$. The remaining computational issues are similar to that of multi-class problems.

3.3 Connection and Difference between MLLR and Existing Methods

The objective function (3.17) can be regarded as a special case of the generalized ϵ-extension model, where $\epsilon_h \to 0$ indicates a maximization over h, and $\epsilon_y = 1$ indicates an "averaging' over y. Taking into account the differences between existing latent variable models, in this section, we provide insights into appropriate model selection for practical applications. Interested readers are also referred to a more detailed discussion on the difference between probabilistic and max-margin approaches in [134, 155].

3.3.1 Maximization vs. marginalization over h

To solve the inference problem on each training example, MLLR adopts a maximization operator over all possible latent variables h, which is the same as in LSSVMs:

$$\Psi_{MLLR}(x, \hat{y}, w) = \underset{h \in \mathcal{H}}{\operatorname{argmax}}[w^T \phi(x, \hat{y}, h)]. \tag{3.19}$$

In contrast, HCRFs and MSSVMs adopt a marginalization strategy over latent variables when computing the score function:

$$\Psi_{HCRF}(x_i, \hat{y}, w) = \log \sum_h \exp[w^T \phi(x_i, \hat{y}, h)] \tag{3.20}$$

The maximization strategy has certain advantages. In the computational perspective, although dynamic programming and belief propagation can solve each inference strategy (max *vs.* sum), the maximization strategy requires less computational effort. Meanwhile, the marginalization strategy requires a defined probabilistic graphical model, which is difficult to acquire in certain scenarios. For example, in object localization, the positions of objects in the images are represented as latent variables. This may lead to a large volume of $|\mathcal{H}|$ and does not require definition of a graphical model. In the maximization strategy, since we are only concerned with the best-scored latent value, latent values that obviously conflict with the model hypothesis can be discarded, resulting in a much

smaller latent value space for each example. Such an approach was adopted in the DPM-LSVM framework [47] and significantly reduced the time and space complexity.

The choice of maximization or marginalization also depends on the characteristics of latent variable h. Specifically, if there is a probabilistic graphical model that efficiently models the interaction between multiple latent variables, such as neighbourhood superpixels in image segmentation or the star-structured part-based model in object detection, marginalization should be the natural choice since it is Bayesian optimal. Meanwhile, for applications with a large uncertainty in latent variables, the maximization strategy may be "overly optimistic" on the most likely states and overlook the influence of uncertainty. Nevertheless, if a graphical model is hard to obtain or a graphical relationship does not exist between latent variables, maximization over latent variable h is a better choice. When there is a large volume of $|\mathcal{H}|$, the sum of an exponentially large number of "incorrect" hs, could be larger than that of the small number of "correct" hs and eventually dominate the inference process, even though each only carries a very small probability.

3.3.2 Max-margin vs. log-likelihood over y

The formulation of objective function in MLLR follows multinomial logistic regression. The posterior probability has the form:

$$p_{\hat{y}}(x; w) = P(Y = \hat{y}|X = x) = \frac{\exp[\Psi(x, \hat{y}, w)]}{\sum_y \exp[\Psi(x, y, w)]}. \tag{3.21}$$

Based on the posterior probabilities, the loss function is defined as a log-likelihood function:

$$L(w) = -\sum_{i=1}^{N} \log p_{y_i}(x_i; w). \tag{3.22}$$

Compared to the max-margin criterion and the hinge loss adopted in LSSVMs and MSSVMs, the log-likelihood function enables high-order optimization methods such as Newton methods. Meanwhile, the log-likelihood function produces probabilistic models. The output posterior probabilities are valuable for building

cascades of classifiers or in applications in which soft assignment outputs are preferred. However, the performance of logistic functions tends to drop significantly when the training data are insufficient or the noise in the training data affect the model hypothesis. SVM-based methods also require less training time than logistic methods due to the absence of the time-consuming log-sum-exp operators.

3.3.3 Regularizer

We add a L1-regularizer in MLLR to avoid overfitting. Of a wide range of regularizers, such as L1-norm, L2-norm and group norm, the most notable characteristics of the L1-regularizer is that it produces sparse model parameters. For MLLR, as discussed previously, the maximization operator over h and the log-likelihood loss function exploit their full advantages in large-scale datasets with complicated latent variables. From an optimization perspective, complex latent variable models require larger computational effort and are more prone to getting stuck in a poor local minimum. To address this problem, sparse models are preferred due to their ability to prune the redundant model parameters, providing robustness in noisy datasets and with less computational effort. Nevertheless, the lost of smoothness in the L1-regularizer may always lead to problems during optimizing.

Other than the advantages of MLLR, we also note that there are some potential problems for the algorithm. The first issue is that MLLR is relatively heavy weighted as it considers data from all the classes at the same time when performing optimization, leading to quite a large burden of memory cost and computational cost. Secondly, inheriting from the nature of logistic regression, MLLR may perform poorly when training data is strongly unbalanced or extremely noisy.

From the theoretical and experimental analysis above, we summarize the application scenarios of MLLRs as follows:

1. *The number of training examples is sufficient, and the model assumptions are not highly violated.* On such occasions, logistic regression usually performs better than SVM.

2. *A probabilistic graphical model for latent variables is unavailable, or no graphical relationship between latent variables exists.* In other words, the

latent values are independently distributed or mutually exclusive. The marginalization over latent variables in HCRFs is inappropriate in such scenarios.

3. *Applications in which probabilistic analysis is preferred.* SVMs result in hard-margin classification assignments. They do not provide reasonable probabilistic outputs.

3.4 Experiment

Here we evaluate MLLR on four different visual recognition applications, and compare the performance of MLLR with LSSVM [167] and LSVM [47] under the same definition of latent variables and employing the same visual features.

3.4.1 Handwritten digit recognition

Handwritten digit recognition is a traditional computer vision task. It is well-known that the accuracy of digit recognition can be greatly improved by explicitly modeling the deformations present in each image, such as arbitrary rotations. Therefore, latent variable models such as LSSVM and MLLR can improve the recognition accuracy by modeling the rotation angle as a latent variable.

For fair comparison, we follow the experimental setup and use the same feature representation of Kumar *et al.* [77] and Chen *et al.* [26], who both adopted LSSVM approaches for recognizing digits. Specifically, [77] used uniform angle selection and proposed a self-paced learning algorithm that outperformed the CCCP algorithm in LSSVMs, while [26] proposed a group norm scheme to explicitly select a subset of discriminative rotation angles that controls the complexity of latent space.

In the experiment, we use the MNIST dataset [80] and perform binary classification on four digit-pairs (1-7,2-7,3-8,8-9) following [77] and [26]. We obtain the feature vector $\phi_h(x)$ by rotating an image x by an angle h, where $h \in \mathcal{H} = \{-60°, 48°, ..., 60°\}$; then projectinto a ten-dimension representation by PCA. Finally, a joint feature vector is specified regarding all 11 angles as

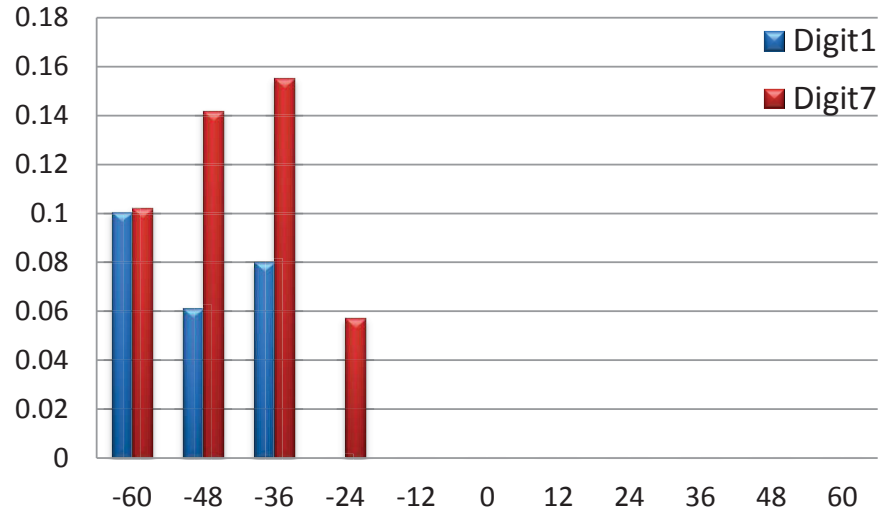
Table 3.1: Prediction accuracies for four digit pairs. N stands for SVM or LR methods without using latent variable models. GD and CDN stand for the two optimization methods in Section 3.2. MLLR-CDN algorithm consistently outperforms two LSSVM-based algorithms.

Digit pair	LSSVM			MLLR		
	N	[77]	[26]	N	GD	CDN
1 vs 7	0.976	0.945	0.988	0.978	0.987	**0.994**
2 vs 7	0.813	0.941	0.956	0.902	0.970	**0.977**
3 vs 8	0.784	0.916	0.923	0.836	0.938	**0.973**
8 vs 9	0.868	0.933	0.954	0.913	0.956	**0.977**

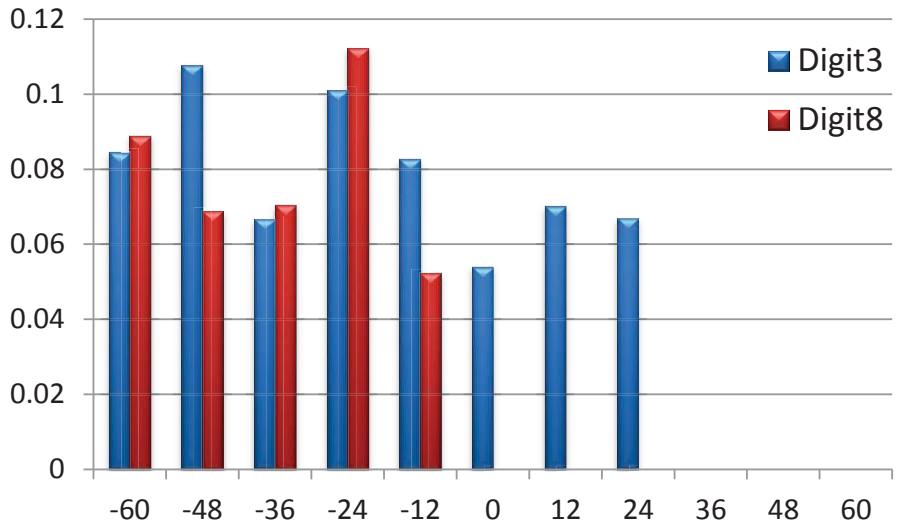
$\phi(x, y, h) = \{0, ..., \phi_h(x), ..., 0\}$. We run 50 iterations for each digit pair and each algorithm. The results are averaged using 10 random trials.

Table 3.1 reveals the resulting prediction accuracies. In general, MLLR outperforms LSSVM-based methods by a margin. That is partially because the MNIST dataset has a relatively large number of training examples (about 6000 for each digit pair), which makes logistic functions more effective. Compared to the general setting of LSSVM adopted in [77], a boost of performance was achieved in [26], where a L1-L2 regularizer was adopted to select a subset of latent variables h. It can be explained that redundant rotation angles may perform as noise, and eventually degrade the accuracy. The L1-regularizer in MLLR provides sparse models and reduces the effect of noise. Meanwhile, the resulting sparse models accelerate the testing process by approximately 3 times. Results show that the models learned by MLLR can also obtain a similar group sparsity as in [26]. Figure 3.2 shows the L1-norm of the parameter vectors for different angles.

Compared to gradient descent approaches, the latent-CDN algorithm converges significantly faster and obtains better performance. The phenomenon advocates the use of second-order gradients, which is impossible for SVM-based methods due to the non-smoothness in objective functions. In practice, the latent-CDN algorithm usually converges within 10 iterations.



(a)



(b)

Figure 3.2: L1-norm of the model parameter vectors for different angles learned by MLLR. x-axis stands for angles and y-axis reveals model responses. Although the L1-regularizer is not specified to produce group sparse models, the resulting model parameters follow a similar pattern.

Table 3.2: Average Precision on the PASCAL VOC 2011 Action Classification task.

Method	Jump	Use Pho.	Play Inst.	Read	Ride Bike	Ride Hor.	Run	Take Pho.	Use Comp.	Walk	Ove.
LSSVM	38.5	29.2	30.8	26.0	49.9	58.4	47.4	18.0	33.4	40.0	37.1
MLLR	38.1	25.8	28.1	25.2	59.4	70.6	60.4	24.0	52.7	40.2	**42.4**

3.4.2 PASCAL action classification

Considering the characteristics of the maximization inference for latent variables, we argue that the most important application scenario for MLLR lies in weakly supervised classification. In weakly supervised problems such as object recognition, the ground-truth only contains image-level labels. The exact object location is regarded as a "semi-structured" latent variable, for which it is impossible to do formal graphical model inference and marginalize them out in a principled way as done in HCRFs.

Human action classification can be regarded as a weakly supervised learning task. We use the PASCAL VOC 2011 [45] action classification dataset, and follow the experimental setting of [10][1], who used the DPM person detector [47] and the standard poselet-based feature vector [96], resulting in a 2405 dimensional feature vector. The classification results are obtained via a 5-fold cross validation on the 'trainval' set, including 1940 training images and 484 testing images.

Table 3.2 shows the Average Precision (AP) of LSSVM using CCCP optimization and MLLR with latent CDN optimization. MLLR significantly outperforms LSSVM in this task by five percents in overall. Moreover, due to the use of second-order derivatives, the computation of MLLR-CDN is around 2 to 10 times faster compared to LSSVM-CCCP, and reach better solutions after convergence. Experimental results on different parameter settings obtain similar results on the comparison between LSSVM and MLLR.

[1]We use the implementation by Kumar *et al.* in their tutorial in CVPR2013, available on http://cvn.ecp.fr/tutorials/cvpr2013/.

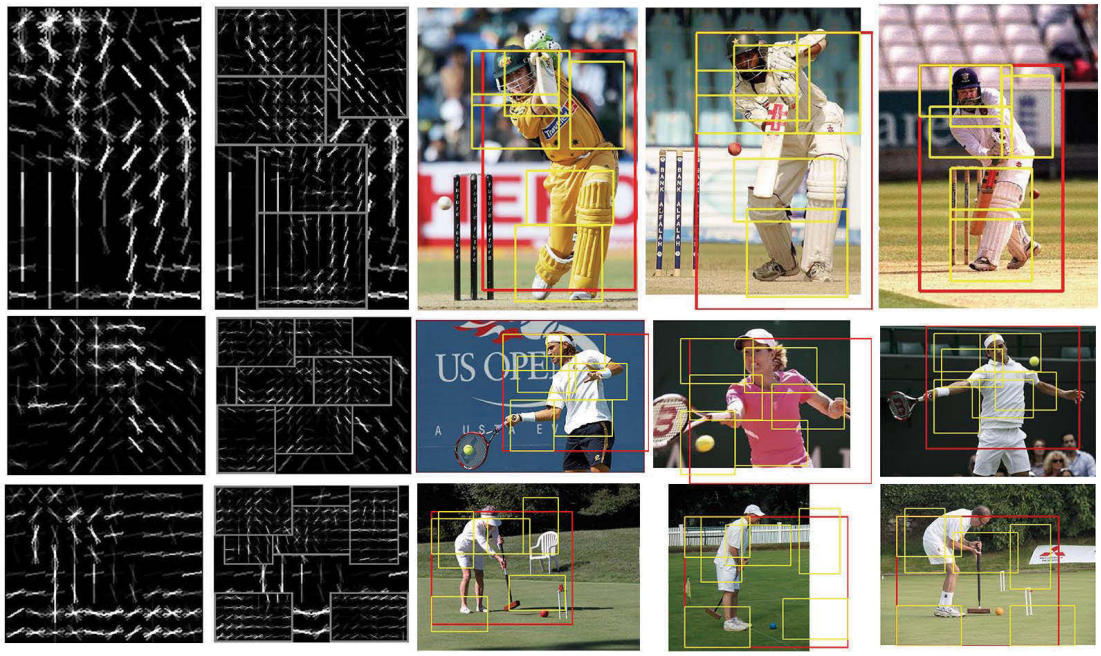


Figure 3.3: Visualization of the result of MLLR for 3 human actions (*cricket-defensive battling*, *tennis-forehand* and *croquet*). Detected root filters are displayed in red, and part filters are shown in yellow. Note that for the images of the class *croquet*, people usually have strong interactions with the background, which degrades the performance.

3.4.3 Sport action recognition

We show another application of MLLR on the task of sport action recognition. In this experiment, we use the sports action dataset, which has six possible actions [64]. The classes are selected so that they have significant confusion due to scene (*tennis-serve* and *tennis-forehand*) and pose (*volleyball-smash* and *tennis-serve*). Here we utilize the DPM framework [47], *i.e.*, regard the exact object location as a latent variable and use the entire image as the training instance. The aspect-ratio of the root-filter in DPM is set as the average of all training images. Detailed analysis of the DPM-MLLR framework will be discussed in the next chapter.

We use 30 images per class for training and 20 per class for testing. MLLR gets a classification accuracy of 78.33%, outperforms LSVM of 74.17%. It is noticeable that the accuracy is comparable with the performance with the fully supervised

model in [64] of 78.67%, indicating that without using the supervision of human body segmentation, the proposed method can achieve comparable performance with the fully supervised approaches. The confusion matrix (Figure 3.4(b)) shows that the proposed algorithm achieves better performance on actions that only allow deformation of a small amount, such as *cricket-defensive battling*. Most of the misses occur when the aspect-ratio of the DPM model is not suitable for the testing images, *e.g.*, images of *croquet* usually involve a large area of background, making it hard to model the action of human subject. Figure 3.3 shows typical results for human action recognition using MLLR.

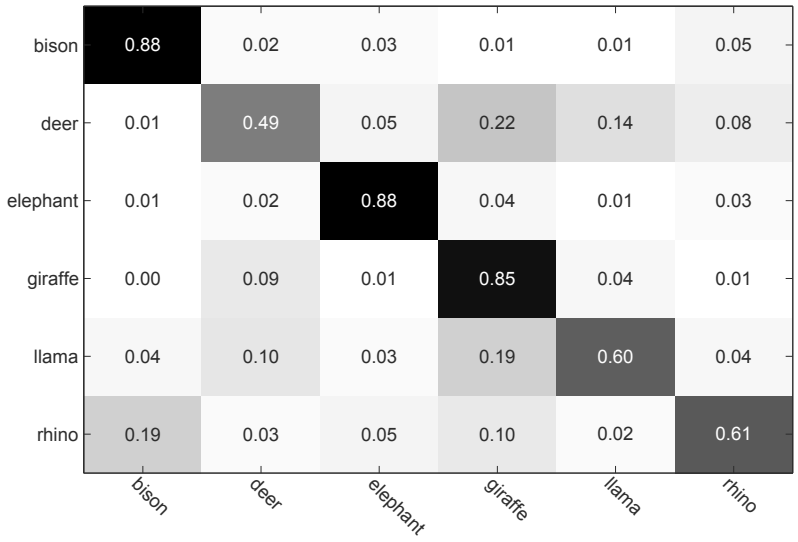
3.4.4 Animal classification

In this section, we show another object classification task with image level supervision for classifying mammals. Analagously, the training data only indicates whether or not an object appears in the image. The exact location of the object is not given, thus we consider the location of the object to be a latent variable z. By enumerating all possible locations in different scales, the latent variable has a huge value space.

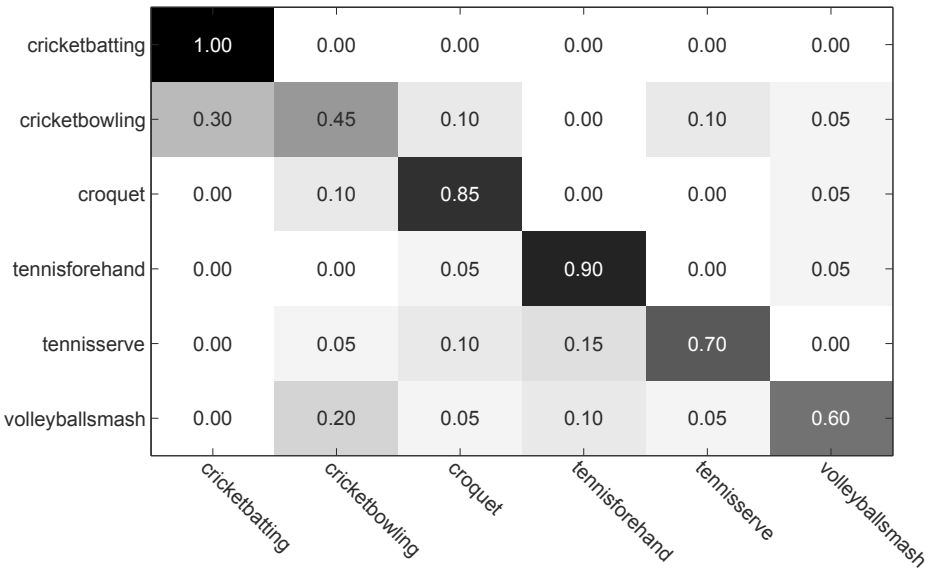
We evaluate the performance of non-Latent SVM, LSVM and MLLR on the mammal dataset [77]. The dataset contains 6 mammal classes, with about 50 images per class. We randomly choose half of the images for training and run 10 rounds of experiments. For all the algorithms, the template HOG features are extracted to indicate the approximate sketch of each object category [47]. The shape of the HOG template is determined as the average aspect ratio of all images in an object category.

First, a linear SVM classifier is constructed for each object category considering the binary classification problem. As shown in Figure 3.5, the template HOG models learned by SVM carry vague semantic information since the training images share similar background scenes.

The model parameters of non-Latent SVM are used to initialize LSVM and MLLR. These two algorithms consider the exact object localization as a latent variable and define the feature vectors $f(x, z)$ as the HOG feature from image x at location z. Therefore, the process to optimize a latent variable here means to find



(a)



(b)

Figure 3.4: Confusion matrices of MLLR in the task of Mammal dataset (a) and Sports dataset (b).

the highest scored position for the foreground object. Figure 3.6 visualizes the change of latent variable (object location) during learning of MLLR. The training process iteratively switches between the two procedures of finding the best

Table 3.3: Classification results for the mammal dataset. Linear SVM is trained without latent variables. All algorithms use the same feature extraction method. We show the mean/std of classification accuracies over 10 rounds of experiments.

Method	linear SVM	LSVM	MLLR
ACC(%)	64.23 ± 2.06	69.59 ± 4.38	73.31 ± 2.77

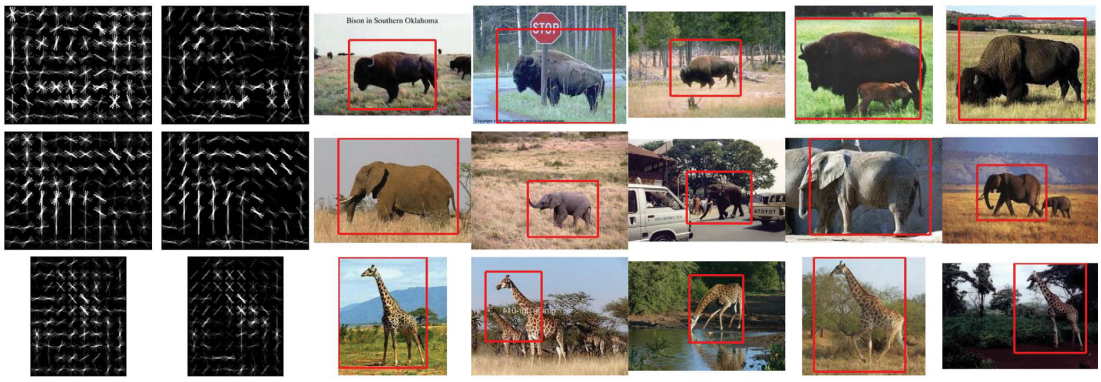


Figure 3.5: Visualization of the result of distributed MLLR on the mammal dataset. The first column contains the HOG models trained by non-latent linear SVM. Given the non-latent linear SVM model as the initialization status, MLLR models remove some of the noise data in the models, as shown in the second column. The last five columns visualize typical results of the latent position found by MLLR. The rows show three of the object categories, which are *bison*, *elephant* and *giraffe* respectively.

bounding box for the object and optimizing the model parameters. As shown in Table 3.3, by introducing latent variables, MLLR and LSVM significantly outperform non-Latent SVM. In most cases, the resulting latent values can accurately locate the objects. On the other hand, MLLR consistently outperforms LSVM by a slight margin. Figure 3.7 shows a typical example in which LSVM fails to achieve correct result due to the calibration problem, while MLLR obtains the correct result.

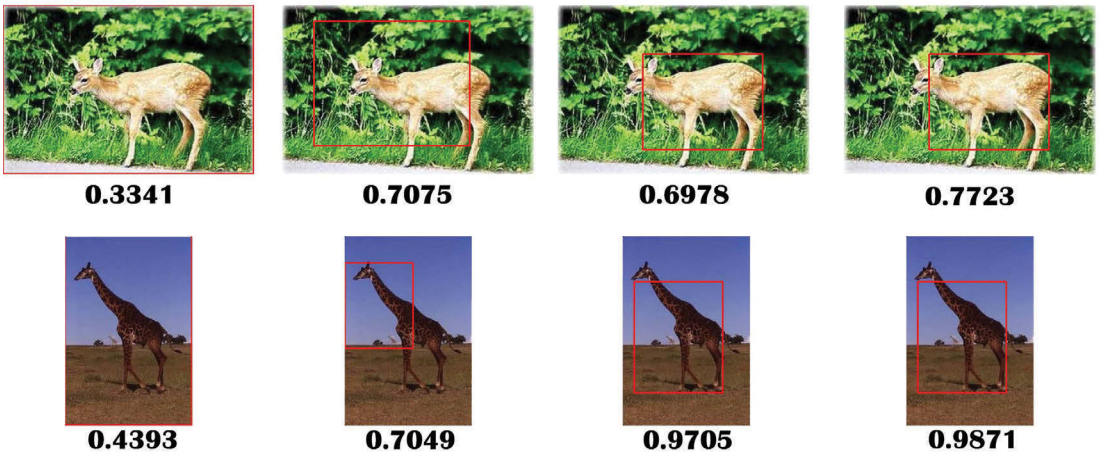
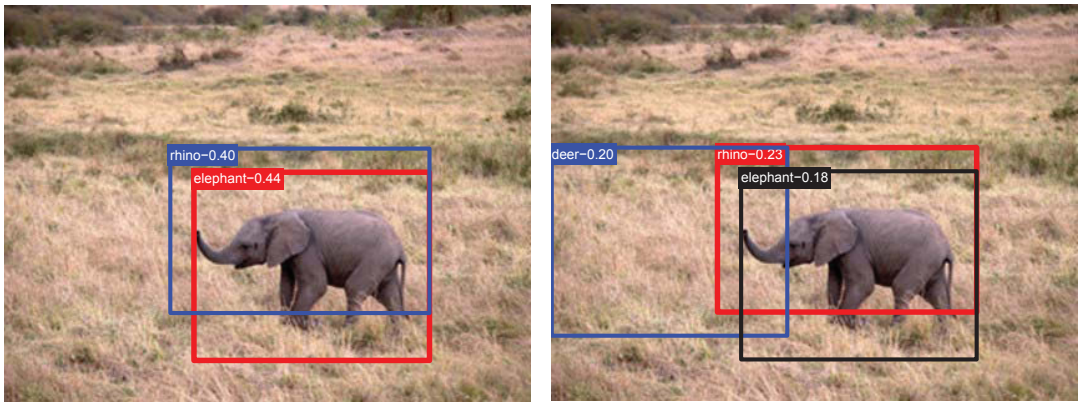


Figure 3.6: Visualization of how the latent variable (object location) changes during learning. Starting from the full bounding boxes, the algorithm iteratively finds the highest scored location of the object. The numbers underneath indicate the output probabilities of MLLR at various stages.



(a) (b)

Figure 3.7: Visualization of the comparison between MLLR (left) and LSVM (right). The text on the bounding box indicates the prediction label and the number shows its probability. We use the sigmoid function to turn the decision values of LSVM into probabilities. Although both algorithms find the same bounding boxes for the classes "elephant" and "rhino", MLLR correctly classifies the object due to a better calibration process.

3.5 Summary

This chapter has introduced theoretical analysis of the proposed latent variable model MLLR. By introducing latent variables into the objective function of logistic regression, MLLR provides efficient latent variable inference and effective probabilistic analysis. We have presented two optimization methods in the framework of concave-convex procedure to solve the objective function of MLLR. Experimental results reveal that MLLR outperforms LSVM in multi-class classification problems, and further beats LSSVMs when the learned hypothesis is strong enough. Weakly supervised object recognition, which introduces a large value set of latent variables (*e.g.*, object location) and performs multi-class classification, could be considered as an important application scenario for MLLR.

By conducting a detailed comparison between MLLR and existing latent variable models with structured output, we have provided suggestions of how to select a proper model in real-world applications. In the following chapters, a set of novel applications will be presented, by which we will study several practical issues for conducting MLLR and also other latent variable models in weakly supervised tasks.

Publications Related to This Chapter

1. **Zhe Xu**, Zhibin Hong, Junjie Wu, Ah Chung Tsoi, Dacheng Tao. Multinomial Latent Logistic Regression for Image Understanding. *IEEE Transactions on Image Processing (TIP)*, 25(2): 973-987, 2016.

Chapter 4

MLLR for Architectural Style Classification

In this chapter, we will study one distinguishable property of the proposed MLLR - the ability to produce probabilistic outputs. For this purpose, we focus on architectural style classification, a new application as a typical example which involves complicated inter-class relationships including re-interpretation, revival, and territoriality. We conduct the proposed MLLR together with deformable part-based model (DPM) and solve this task in a weakly supervised setting. Except for the standard classification results, we will also investigate the proposed DPM-MLLR framework on producing additional discoveries on architectural styles using probabilistic analysis.

4.1 Introduction

Object recognition has been extensively studied in the history of computer vision. In recent years, the research objective has evolved drastically, from highly artificial experimental settings with small datasets and restricted range of categories [49], to more challenging recognition tasks involving an increasing number of object categories and more real-world experimental settings [45, 112]. Except for the increase of the number of recognizing object categories, such evolution also raises new challenges to researchers due to the underlying inter-class relationships

between categories.

Buildings can be classified according to architectural styles, where each style possesses a set of unique and distinguishing features [44]. Some features, especially the façade and its decorations, enable automatic classification using computer vision methods. Architectural style classification has an important property that styles are not independently and identically distributed. The generation of architectural styles evolves as a gradual process over time, where characteristics such as territoriality and re-interpretation lead to complicated relationships between different architectural styles. Therefore, architectural style classification is a typical example where inter-class relationships play an important role.

Most of existing architectural style classification algorithms focus on efficient extraction of discriminative local-based patches or patterns [11,30,42,61,106]. In a four-style classification problem, Chu *et al.* [30] extracted visual patterns by modeling spatial configurations to address object scaling, rotation, and deformation. Goel *et al.* [61] achieved nearly perfect results on published datasets by mining word pairs and semantic patterns, and therefore tested this approach further on a more challenging five-class dataset collected from the internet. Zhang *et al.* [171] used "blocklets" to represent basic architectural components and adopted hierarchical sparse coding to model these blocklets. However, as argued in [148], some patches that look totally different can be very close in the feature space, which degrades the performance of local patches in understanding detail-rich architecture images. One recent study showed the possibility of cross-domain matching from sketches to building images [125], and this inspired us to employ sketch-like features to represent the building façades.

The Deformable Part-based Model (DPM) [47] is a popular scheme that employs sketch-like Histogram of Oriented Gradient (HOG) features. DPM models both global and local cues and enables flexible configuration of local parts by introducing so-called deformation costs. By adopting a latent SVM (LSVM) algorithm for training, the DPM-LSVM framework produces class labels effectively via part-based modeling. However, the predicted labels are deterministic results; in order to enable rational explanation of the gradual transition and mixture of architectural styles, it would be preferable to provide soft assignments and introduce the concept of probability into the model.

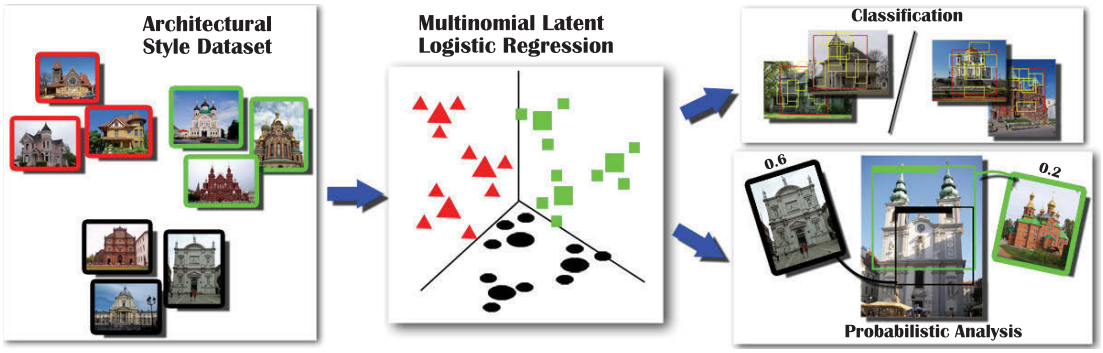


Figure 4.1: Schematic illustration of architectural style classification using Multinomial Latent Logistic Regression (MLLR). Given a new large-scale architectural style dataset, we model the façade of buildings using deformable part-based models. The resulting classifiers can provide probabilistic analysis along with the standard classification results.

Therefore, for architectural style classification task, we employ DPM as the feature representation method and adopt the proposed MLLR as the learning algorithm. MLLR is supposed to enjoy great advantages in this problem setting. By regarding object location as latent variables and producing probabilistic outputs, MLLR can produce a set of interesting discoveries together with traditional classification accuracy, such as inter-class relationship modeling, and style analysis for individual buildings (Figure 4.1). Experimental results reveal that the proposed method can not only achieve state-of-the-art classification performance, but also presents effective probabilistic analysis aware of the rich inter-class relationships between architectural styles.

We have also noticed that one reason why few previous studies focus on architectural style analysis lies in the lack of a well-organized, large-scale dataset. Therefore, we collect a new and challenging dataset containing 25 architectural styles. The dataset possesses several preferred properties enclosing in architectural style classification, including multiple classes, inter-class relationships, hierarchical structure, change of views and scales. Our new dataset provides an improved platform for evaluating the performance of existing classification algorithms, and encourages the design of new ones.

4.2 Architectural Style Dataset

An architectural style is a specific construction, characterized by its notable features. For instance, unique features, such as pointed arches, rib vaults, rose windows and ornate façades, make it possible to distinguish the *Gothic* style from other styles. Architectural history has dictated that there are complicated inter-relationships between different styles, including rebellion, special territoriality, revivals, and re-interpretations. As a consequence, it is difficult to strictly classify two styles using a standard criterion.

In order to study architectural styles and model their underlying relationships, we collected a new architectural style dataset from Wikimedia[1]. We obtained the initial list by querying with the keyword *"Architecture_by_style"*, and downloaded images from subcategories following Wikimedia's hierarchy using the depth-first search strategy. The crawled images were manually filtered to exclude images of non-buildings, interior decorations, or part of a building. Therefore, the remaining images contained only the exterior façade of buildings. Styles with too few images were discarded, resulting in a total of 25 styles. The number of images in each style varies from 60 to 300, and altogether the dataset contains approximately $5,000$ images[2].

We propose several challenges to extensively exploit the data size and rich relationships between different architectural styles in this dataset. Figure 4.2 illustrates the dataset.

- **Multi-class classification**. To the best of our knowledge, this dataset is the largest publicly available dataset for architectural style classification. Other popular datasets related to buildings do exist, such as the Oxford Landmark dataset [106]. However, their main purpose is for the retrieval of individual landmark buildings rather than classification of architectural styles. A discussion of the difference between "style" and "content" can be found in [52]. There are some researches on different type of art styles, such as painting [156, 179] and car designing [82], which may also provide

[1] From Wikimedia commons.
http://commons.wikimedia.org/wiki/Category:Architecture_by_style.

[2] https://sites.google.com/site/zhexuutssjtu/projects/arch

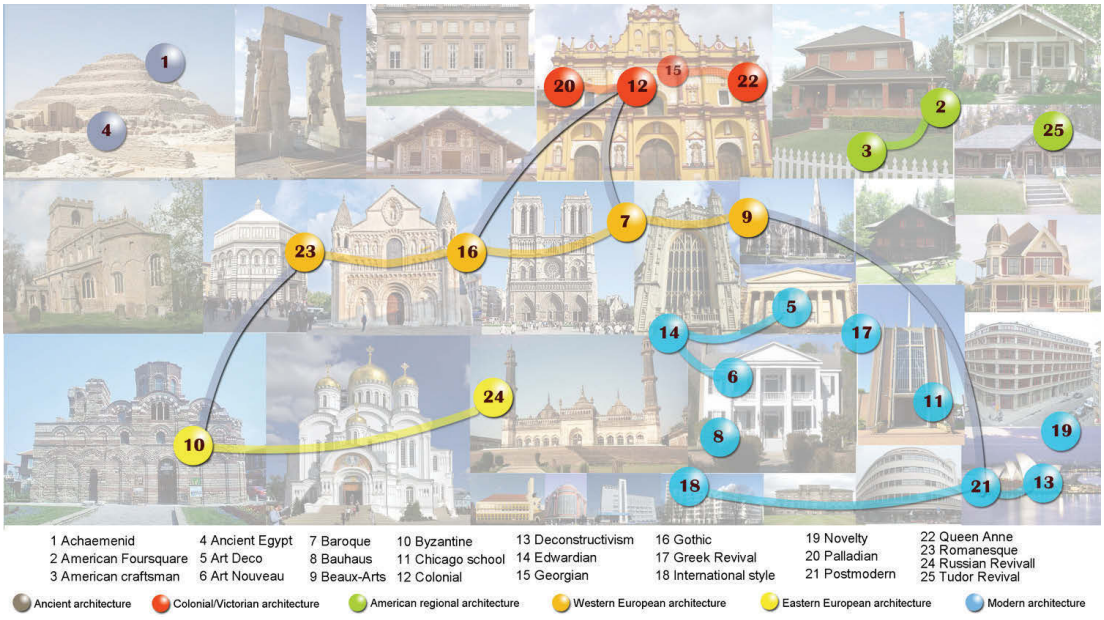


Figure 4.2: Illustration of the architectural style dataset. Each of the 25 styles is represented by a circle with the respective number in the middle, where different colors indicate broad concepts, such as modern architecture and medieval architecture. The styles are arranged according to time order, where newer ones are placed in the right of ancient ones. Various inter-class relationships exist between the styles, *e.g.*, lines between circles stand for following relationships; smaller circles around large ones indicate sub-categories. Typical images of the styles are shown in the background. Better viewed in color.

cross-domain knowledge from other aspects of art styles.

- **Modeling inter-class relationships between styles**. Various relationships exist between the 25 architectural styles, *e.g.*, following, revival, and against. Styles can be roughly classified into broad concepts, such as ancient architecture, medieval architecture and modern architecture, and thus further be arranged in a hierarchical structure. For reference, we summarize the relationships between different styles verified by Wikipedia. It is of interest to explore whether computer vision algorithms can efficiently extract the underlying inter-class relationships.

- **Modeling intra-class variance within a style**. The establishment of

an architectural style is a gradual process. When styles spread to other locations, each location develops its own unique characteristics. On the other hand, each building is unique due the personalities of different architects. Therefore, it is challenging to find common features within a style, as well highlighting the specific design of an individual building.

- **Style analysis for an individual building**. When designing a building, an architect sometimes integrates several different style elements. The building can therefore be represented as a mixture of styles. An algorithm should be able to model this phenomenon, *e.g.*, show that the window is inspired by style I and the arch by style II.

4.3 Model Description

In general, buildings are constructed by a set of basic elements, such as doors, windows, arches, and towers. Therefore, in order to recognize multiple architecture styles, it is reasonable to model buildings through part-based feature representations. The Deformable Part-based Model (DPM) [47] is perhaps the most popular part-based method in the field of object recognition. The definition of a DPM involves multiple forms of latent variables, and thus requires an effective latent variable paradigm to train the models. Here, we adopt DPMs to represent architecture styles, and show that the proposed MLLR is a better solution to solve multi-class problems than the latent SVM algorithm used in the original implementation of DPM.

4.3.1 Deformable part-based model

The Deformable Part-based Model (DPM) aims to model non-rigid deformations for recognizing generic objects such as people and cars. The key assumption is that objects can be represented by a combination of multiple object parts, while the parts can be displaced flexibly around an anchor position with some restrictions denoting by a deformable configuration. For example, to recognize a pedestrian in still images, a DPM models people by several parts, such as head, torso, arms and legs. Different parts have varied characteristics: a head should

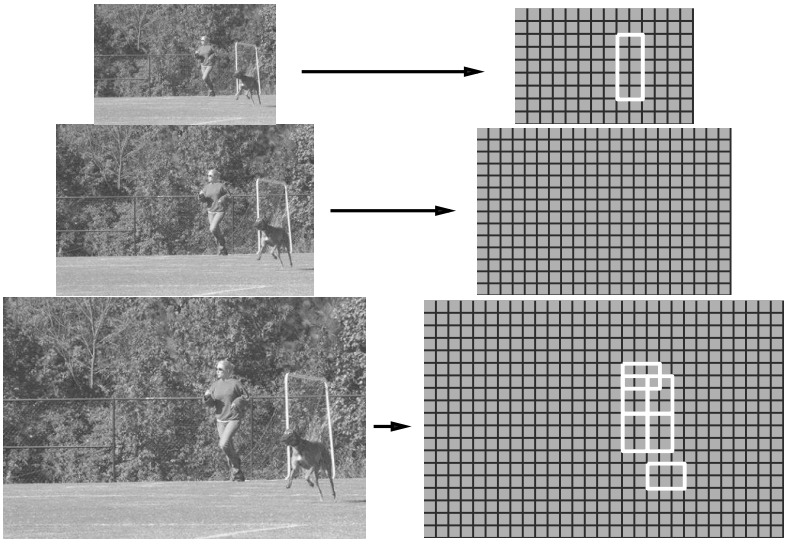


Figure 4.3: Illustration of the feature pyramid in a DPM. Part filters are placed at twice the spatial resolution than the root filter. Original figure can be found in [47].

be right above the torso with little displacement, while arms can be deformable within a rather large region around the torso. These properties are then modeled via the deformable configurations in DPMs.

Based on this assumption, DPM describes an object by a star-structured part-based model defined by a "root" filter plus a set of parts filters and associated deformation models. The score at a particular position and scale is computed as the score of the root filter at given location, plus the sum of part filters by computing the maximum score over placements of each part through the part filter score and a deformation cost. As shown in Figure 4.3, by representing an image in a multi-scale HOG feature pyramid, DPM models visual appearance at multiple scales, where the part filters resides at twice the spatial resolution relative to the root filter.

Formally, a model for an object with n parts is defined by a $(n + 2)$-tuple $(F_0, P_1, ..., P_n, b)$ where F_0 is a root filter, P_i is a model for the i-th part and b is a real-valued bias term. Each part model is defined by a 3-tuple (F_i, v_i, d_i) for the part filter, anchor point, and deformation cost respectively. An object hypothesis

specifies the location of each filter in the feature pyramid, $z = (p_0, ..., p_n)$, where $p_i = (x_i, y_i, l_i)$ is the position and scale of the i-th filter respectively. The score of a hypothesis is given by:

$$\text{score}(p_0, ..., p_n) = \sum_{i=0}^{n} F_i' \phi(H, p_i) - \sum_{i=1}^{n} d_i \cdot \phi_d(dx_i, dy_i) + b, \tag{4.1}$$

where the first term are filter scores; part displacements are denoted by

$$(dx_i, dy_i) = (x_i, y_i) - (2(x_0, y_0) + v_i),$$

and

$$\phi_d(dx, dy) = (dx, dy, dx^2, dy^2)$$

are deformation features; b is a bias.

The score function $s_\beta(x)$ can be simplified as a dot product $\beta \cdot \Psi(H, z)$ between a vector of model parameters β and a vector $\Psi(H, z)$,

$$\beta = (F_0', ..., F_n', d_1, ..., d_n, b), \tag{4.2}$$

$$\Psi(H, z) = (\phi(H, p_0), ...\phi(H, p_n), -\phi_d(dx_1, dy_1), ..., -\phi_d(dx_n, dy_n), 1). \tag{4.3}$$

A visualization of DPMs in architectural style classification is shown in Figure 4.4. Template HOG features model building façade by sketch-like representations. DPMs can discover distinguishable object parts automatically, and organize the discovered parts and the whole object as a star-structured framework that allows slight deformation. For *Gothic* style shown in the figure, the resultant root model shows typical façade outline of *Gothic* style buildings, and the part filters captures discriminative architectural elements such as rose windows.

Given various problem settings, the definition of DPMs can involve as much as three different kinds of latent variables. Consider the weakly supervised object recognition task in our application where only image-level labels are provided during training, the associated latent variables include:

- Root position. An object may appear at varied position, scale and in different aspect ratios in an image. Therefore, the position of the root filter

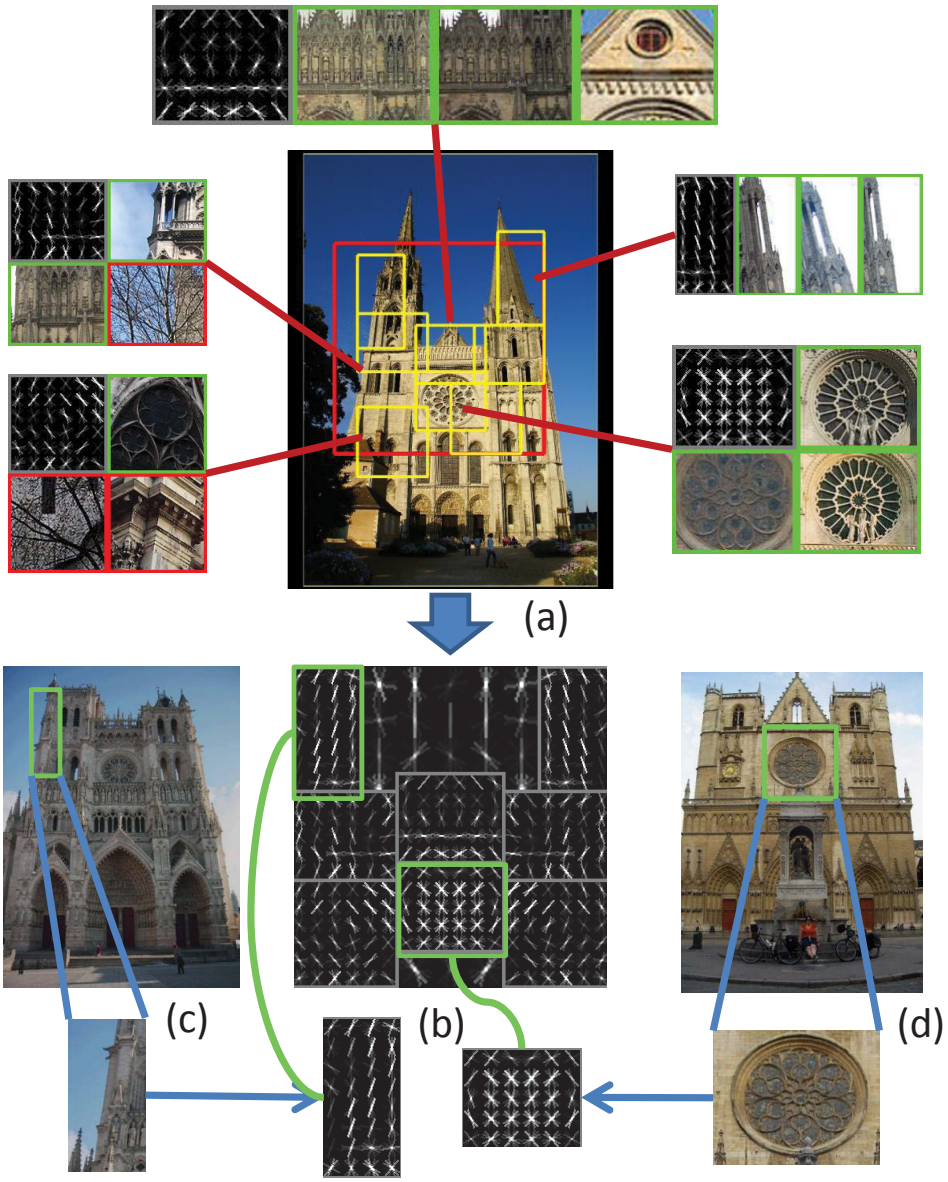


Figure 4.4: Visualization of the use of DPM in architectural style classification. (a)(c)(d) show detection results for different testing images. The trained model for *Gothic* architectural style is shown in (b).

in a HOG pyramid is considered as a latent variable in the model.

- Part position. In DPM, each part model is defined by a triplet including a part filter, an "anchor" position for the part relative to the root position,

and a deformation cost for each possible placement of the part relative to the anchor position. Part positions, therefore, are also regarded as a latent variable, as each part can be displaced with respect of the anchor position restricted by the deformation cost.

- Component label. DPM introduces a mixture-component model to account for intra-class variance. The score of a mixture model at a particular position and scale is then the maximum over components. Here, the latent variable specifies a component label and a configuration for that component.

4.3.2 Latent SVM

To train the model parameters β, DPM adopts a latent SVM algorithm described in Section 2.1.4, whose objective function is defined analogically to classical SVMs as:

$$L(D) = \frac{1}{2}||\beta||^2 + C\sum_{i=1}^{N}\max(0, 1 - y_i s_\beta(x_i)), \tag{4.4}$$

where $\max(0, 1 - y_i s_\beta(x_i))$ is the standard hinge loss and C is the soft margin parameter, which controls the weight of the regularization term. Due to the non-convex training objective function, latent SVM is solved using a coordinate descent framework.

As DPM is originally proposed for object detection [47], the latent SVM algorithm is mainly focused on solving binary output predictions such as a classifier for detecting people against the background. Following [47], Pandey *et al.* [102] use DPM in a scene recognition and a weakly supervised object localization task. They point out that scene recognition can also be viewed as a "part-based" problem, where the root captures the entire image and the parts encompass moveable "regions of interest" (ROIs). In their experiments, they find that when the model is trained using the entire image as the root, the resulting performance is not as good as expected. They remark that the root filter should be allowed to move, and regarding the position of the root filter as another latent variable.

However, there are some notably drawbacks of conducting this strategy to train latent SVMs for classifying architectural styles. First, it is argued that SVM tends to underperform when training data is highly imbalanced, *i.e.*, negative

examples far outnumber positive examples [158]. The vast "background" class in the object detection framework introduces a serious imbalance between positive and negative examples. Moreover, in LSVM, the training process needs to fix the latent value for positive examples, while keeping all possible latent values for negative examples. This process makes the imbalance problem even more severe.

Second, the dominant method for solving multi-class problems using SVM has been based on reducing a single multi-class problem to multiple binary problems. However, since each binary problem is trained independently, adopting this strategy is problematic because it cannot capture correlations between different classes. As a result, the output decision values are not comparable, and this is known as the "calibration" problem. MLLR trains all classes simultaneously by introducing a unified objective function and in this way does not suffer from the hazard of different biases occurring with the multi-class problem and imbalanced training data.

Finally, SVM does not provide an effective probabilistic analysis with a soft boundary. Given an input example, the corresponding output of SVM is called the decision value, which is the distance from the example to the decision boundary. A previous work [90], in which a normalization process was proposed to convert the decision values of SVM to probabilistic outputs, does not provide a genuine probabilistic explanation. MLLR produces comparable classification results for multiple classes, and more reasonably turns them into probabilities.

Therefore, in our implementation, we follow the setup of Pandey *et al.* for relaxing the position of root filter, *i.e.*, use a square root filter and restrict it to have at least 40% overlap with the image, and employ the proposed multinomial latent logistic regression to train the DPM models.

4.3.3 DPM-MLLR framework

The proposed method utilizes DPM to represent non-rigid objects and employs MLLR for multi-class predictions and probabilistic output analysis. Two forms of latent variables are used in the proposed method: root position and part position; mixture-component analysis is not introduced in our model in order to reduce the computational complexity. The latent variable is then defined as

$z = (p_0, p_1, ..., p_n)$.

Training procedure. The training procedure of DPM is relatively complicated due to the definition of root and part filters, and the non-convex objective function. We summarize the whole process as follows:

Initializing root filters. One predominant characteristic of architectural style classification is that objects to be recognized usually follow an object-in-the-center prior and are relatively large in scale. As a result, we adopt a rather simple initialization strategy that extracts features from the largest crop with squared shape in training images, and pre-train a set of classifiers via standard logistic regression as the initialization for root filters.

Root filter training. The second step is to relax the location of root filters and train classifiers using latent variable models. As parts are not yet introduced into the model, the only latent variable now is the position of the root filter in the feature pyramid. We adopt the gradient descent optimization method described in Section 3.2.4 and train models iteratively by updating latent variables or model parameters with the other one fixed.

Initializing part filters. After several iterations of outer loops in CCCP (typically 2-3), the resultant root filters can usually produce reasonable discriminative power for the recognizing object. Based on them, a set of part filters are defined to further model the details of objects. Following [47], part filters are defined greedily by placing parts to cover high-energy regions of the root filter. Note that for different object categories, the position and shape of part filters may be different, resulting in different feature representations.

Joint training. After introducing part filters, the DPM-MLLR framework jointly trains multiscale part-based models of the root filter and part filters on multiple classes simultaneously. We adopt a dynamic programming strategy similar to [47] to improve the efficiency of latent variable inference. The final DPMs gather the filter weight of the root and multiple parts, and the associated deformation costs.

Calibration between different subspaces. Compared to LSVM, a crucial advantage of MLLR in multi-class classification problems is that it can produce comparable results when generating scores for different categories. Here we present a detailed explanation of the definition of the "calibration" problem and

why MLLR can tackle this problem.

Consider the process of predicting the class label for an image. Assume that there are altogether K classes in the dataset, and there are Z possible values for the latent variable in this image. As discussed above, the extracted feature vectors by DPM for class A and class B are not always the same due to different shapes and scales of the root filter and part filters. Therefore, the feature representation is in fact defined by a triplet (x, y, h), where x is an input image, y is an output label, and h is a possible latent assignment.

Note that in LSVM, when the one-against-rest strategy is used, each time a classifier with binary outputs is learned for an object category. Therefore, for the k-th class, the training process in fact only utilizes a single form of feature representation $\phi(\cdot, y_k, \cdot)$ with y_k fixed. As a result, the classifiers $\beta = \{\beta_k\}$ are trained on multiple subspaces defined by $\phi(\cdot, y_k, \cdot)$ respectively. The scores generated on different subspaces are not directly comparable, leading to the "calibration" problem.

On the contrary, MLLR simultaneously trains the classifiers for multiple classes in a unified framework. Specifically, given a training image, MLLR extracts K sets of feature representations, each of which corresponds to a specific object category. An image is then associated with $K \times Z$ feature vectors, accounting for each latent assignments on each output label. Although different categories have their own set of feature representations lying in multiple subspaces, the generated scores for multiple categories are comparable by projecting the feature vectors into a single score and compute probabilities on them. The probability of the image belong to a category k is then given by:

$$p_k(x; \beta) = \frac{\exp(score(x, k))}{\sum_{l=1}^{K} \exp(score(x, l))}, \tag{4.5}$$

where $score(x, k) = \max_h \beta_k \phi(x, k, h)$. Figure 4.5 illustrates this process.

4.4 Experiment

In the experiments, we adopt the proposed DPM-MLLR framework to perform architectural style classification and compare the results with DPM-LSVM. The

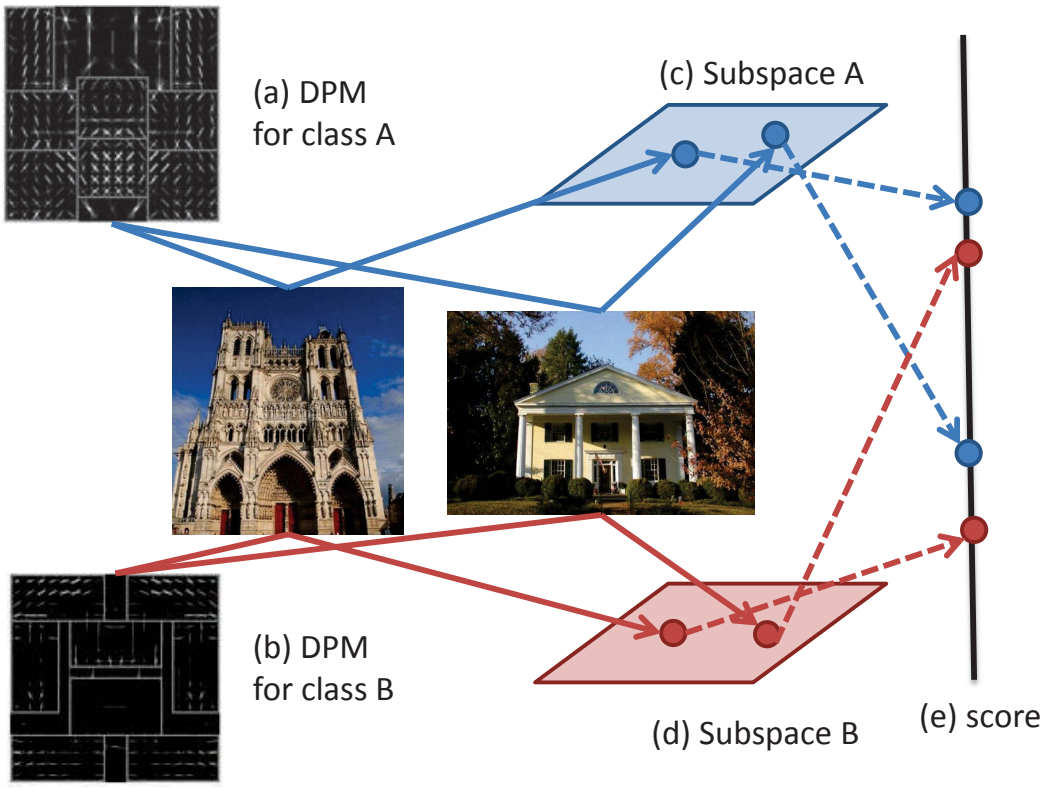


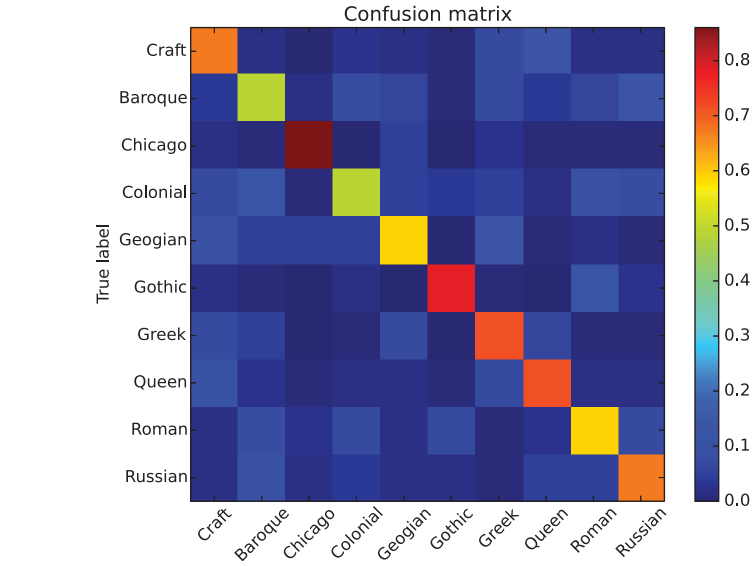
Figure 4.5: MLLR maps the classifier results of multiple classes to a unified score function. The resultant scores are directly comparable.

experiments are presented in three steps. In the first step, we choose ten architectural styles that are relatively distinguishable by their façades and have lower intra-class variance. As a result, using sketch-like HOG features, DPM can clearly demonstrate the characteristics of these styles. Second, we evaluate the effect of a more extensive multi-class problem and larger intra-class variance using the full dataset. Given the probabilistic results, we formulate inter-class relationships using a style relationship map. The third part illustrates individual building style analysis of MLLR.

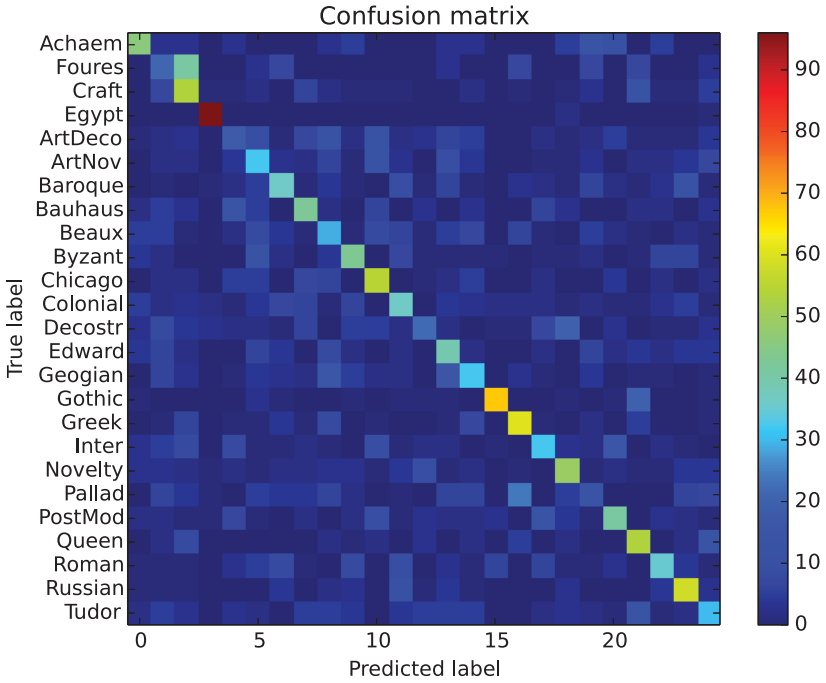
Table 4.1: Results on the architectural style classification dataset. MLLR consistently outperforms LSVM. Multiple features (*e.g.*, MLLR+SP) are combined by adopting a late fusion method using the softmax function on classifier outputs.

	GIST	SP	OB w/o. part	OB w. part	DPM LSVM	DPM MLLR	MLLR +SP
10 classes	30.74	60.08	62.26	63.76	65.67	67.80	**69.17**
25 classes	17.39	44.52	42.50	45.41	37.69	42.55	**46.21**

Figure 4.6: Testing results for the ten architectural styles. The first two columns visualize the result root and part filters for each model. From top to bottom: *Baroque, Chicago school, Gothic, Greek Revival, Queen Anne, Romanesque* and *Russian Revival* architecture. Detected root filters are displayed in red, and part filters are shown in yellow. Better viewed in color.



(a)



(b)

Figure 4.7: Confusion matrix for MLLR on the two experimental settings.

4.4.1 Classification task

A ten-class sub-dataset is exploited for the first classification task, most of which have prominent façade or decoration features, such as pointed arches, the ribbed vaults and the flying buttresses characteristics of *Gothic* architecture. For each class, 30 images are randomly chosen as training images and the remaining images are used for testing, $1,716$ in total. We run a ten-fold experiment. The proposed algorithm is denoted by DPM-MLLR. Table 4.1 compares the classification accuracy of DPM-MLLR with other algorithms, including GIST [137], Spatial Pyramid (SP) [79], Object Bank [85], and DPM-LSVM [102]. DPM-MLLR outperforms LSVM in terms of overall accuracy. It is noted that DPM and local patch-based algorithms, such as Spatial Pyramid, have complementary properties. We therefore combine their results using a naive softmax function and achieve the best result with nearly 70% accuracy.

Figure 4.6 shows the trained models and typical detection results of MLLR. Close inspection of the results reveals that the models capture discriminative features of the styles. For instance, the model representing *American Queen Anne* architecture detects twin gables and allows them to move within limits (in the top of the root). Thus, the model is robust to slight view changes and intra-class variance.

The confusion matrices of the proposed algorithm on the 10- and 25-class classification task are shown in Figure 4.7.

4.4.2 Inter-class relationships between styles

This part of the experiment is implemented on the full dataset. The 25-class dataset has stronger intra-class invariance and is harder to distinguish purely by the façades. The results show that algorithms that take the features of the entire image into consideration, *i.e.*, Spatial Pyramid and Object Bank, achieve superior performance (Table 4.1). DPM-MLLR has slightly lower accuracy. However, compared to the result of the ten-class problem, MLLR outperforms LSVM by a larger margin due to the increased number of classes. Again, the combined MLLR-SP algorithm achieves the best result.

Despite classification accuracies, the proposed algorithm provides a proba-

bilistic style distribution for each building image. By summing the probabilities, we obtain a probabilistic confusion matrix, which is further decomposed into a style inter-relationship network by assigning an edge between two styles whose confusion probability exceeds a given threshold. Figure 4.8 shows the resulting relationship map of the 25-class dataset. According to the set of relationships between styles collected from Wikipedia, the proposed algorithm gets a recall of 0.66, and the average precision $AP@10$ is 0.51.

Unlike hard-margin confusion matrices, large values can occur in the probabilistic confusion matrix under two occasions. The first is when two styles are similar to each other, making them hard to distinguish, and the second is when two styles appear on different parts of the same building, which is most likely to happen when the styles spread to a same place and start to mix. We try to distinguish these two scenarios by considering whether the optimized detecting bounding boxes of the two styles frequently appear at the same location. Experimental results show that the averaging bounding box intersection ratio of the *Queen Anne* and *American Craftsman* styles is higher than that of the *Baroque* and *Colonial* style (0.56 vs. 0.46), which means that the first two styles have a similar façade and should therefore be more dependent on local parts for classification. This phenomenon is in accordance with architectural history.

4.4.3 Individual building analysis

MLLR makes it possible to analyze the architectural style of a building probabilistically. Figure 4.9 shows two typical situations in which the algorithm gives comparable scores for at least two styles, which correspond to the two scenarios discussed in the previous subsection. The first is when different architectural styles share similar features, such as pear-shaped domes in both *Baroque* and *Russian Revival* architecture. The second scenario appears when architects design new buildings that combine several different architectural styles. For instance, Figure 4.9(b) shows a failed classification case in which the main body of the building follows the *Queen Anne* style, while the terrace shows a strong *Greek* sense. MLLR mistakenly classifies the building as *Greek Revival* style due to the unusual shooting angle, which places the main body in side view. However,

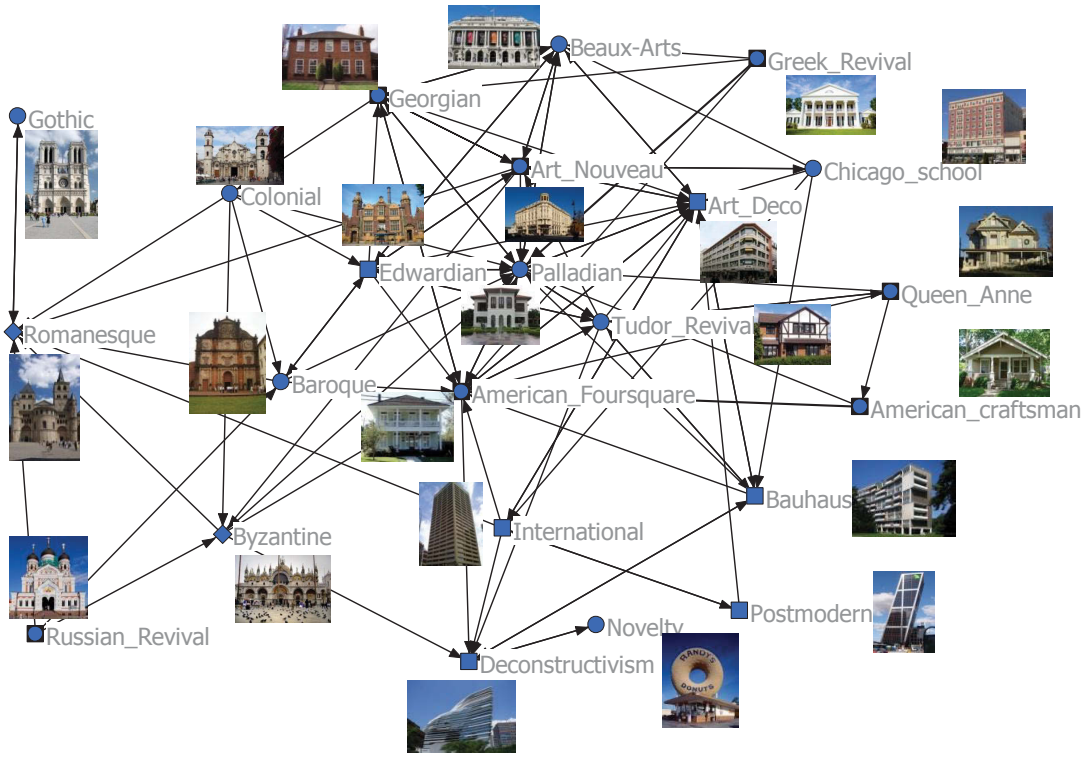


Figure 4.8: An architectural style relationship map generated by the proposed algorithm. The confusion probability between style A and B is obtained by summing the probabilities with regard to B for all images labeled by A. Only links whose weight exceeds a given threshold are shown in the figure. Modern styles, such as *Postmodern* and *International* style, are connected, while the links between modern and medieval styles are weak. The figure is drawn using NetDraw [16].

MLLR discovers interesting patterns in the building that indicates a combination of different styles, and assigns probabilities for each style according to the training set.

4.5 Summary

In this chapter, we focus on the multi-class classification task, where the proposed MLLR provides calibrated results on multiple categories and presents effective

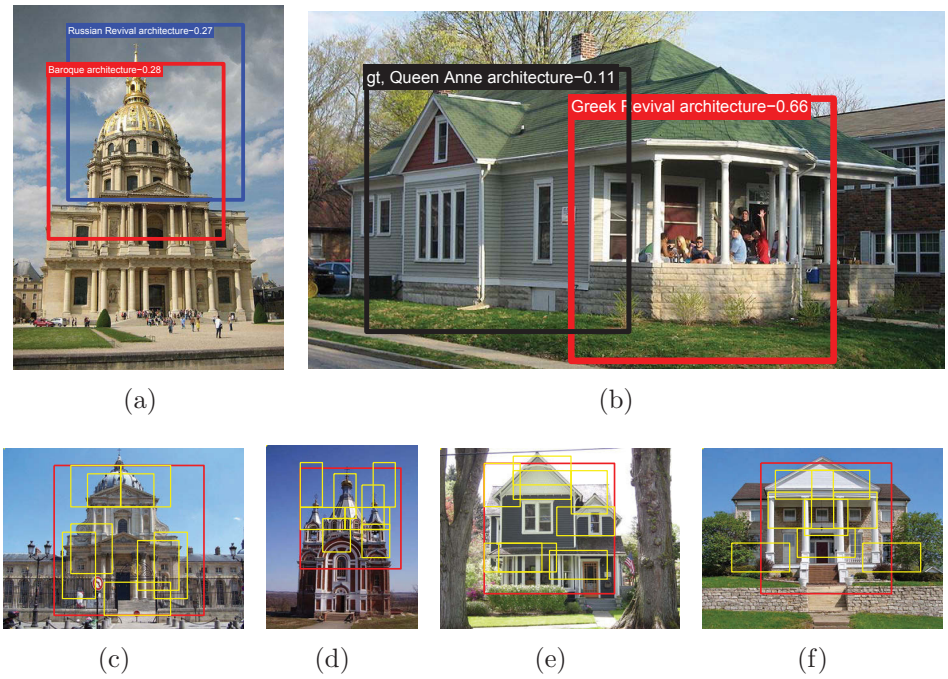


Figure 4.9: MLLR detects the optimized latent position for each class and outputs a global list of probabilities for each class. (a) Parts shared by different styles. (b) A building that combines several styles. (c)-(f) Typical detection results for the four styles appearing in (a) and (b), i.e., from left to right, *Baroque, Russian Revival, Queen Anne* and *Greek Revival*.

probabilistic analysis on output labels. Our study is conducted on a novel application of object recognition - architectural style classification, in which rich inter-class relationships exist between multiple categories. For this application, except for the traditional evaluation criterion of classification accuracy, MLLR also makes it possible for analyzing architectural styles in depth, including generating style relationship maps and performing style analysis on multiple elements of individual buildings.

Meanwhile, we have shown that MLLR is an effective approach to train the famous deformable part-based model in multi-class classification tasks. By simultaneously training classifiers for multiple categories, MLLR eliminates the calibration problem of latent SVM, leading to a reasonably performance improvement. In the next chapter, we will employ MLLR on a more challenging task - weakly supervised fine-grained visual categorization, and discuss the impact of various

initialization strategies on the performance of weakly supervised learning.

Publications Related to This Chapter

1. **Zhe Xu**, Dacheng Tao, Ya Zhang, Junjie Wu, Ah Chung Tsoi. Architectural Style Classification using Multinomial Latent Logistic Regression. *European Conference on Computer Vision (ECCV' 14)*, pp. 600-615, 2014.

Chapter 5

MLLR for Fine-grained Categorization

In this chapter, we will focus on an important aspect for training latent variable models - the initialization. As the objective function of the proposed MLLR is non-convex, a bad initialization could lead to poor local minima and thus result in suboptimal results. To study this issue, we investigate an extreme case of performing weakly supervised learning on fine-grained visual categorization which aims to distinguish object categories in the subordinate level. Considering that different subcategories in fact belong to a more generic concept, we propose a novel multi-task co-localization algorithm to perform initialization for the non-convex MLLR objective function, and incorporate MLLR into the framework of multi-instance learning (MIL) to solve the weakly supervised problem. Experimental results prove the effectiveness of the designed initialization scheme.

5.1 Introduction

One of the most intuitive applications of the proposed paradigm MLLR is for solving weakly supervised problems [38,51,53] in the form of multi-instance learning [2,4,39]. Multi-instance learning (MIL) is a variation on supervised learning, where the labels are provided on a set of bags (each contains several instances) instead of individual instances. In a simple binary classification problem, MIL

assumes that a bag is labeled as negative if all instances in this bag are negative; but labeled positive if at least one of the instances is positive. From the perspective of learning approaches, MIL can be solved using a latent variable model, by regarding the assignment of an optimized instance in a positive bag as a latent variable and performing a maximum a *posteriori* (MAP) procedure to infer the latent variable assignment. This property advocates the application of MLLR to solve MIL problems, especially for ones involving multi-class predictions.

In the context of object recognition, MIL provides an intuitive way to solve and explain the weakly supervised object recognition process, *i.e.*, iteratively answering the question of "where" the objects are and "what" the objects look like, leading to numerous successes in the literature [47, 67, 102]. However, as the objective function of latent variable models such as MLLR is usually non-convex, it is crucial to design initialization strategies carefully before conducting multi-instance learning extensively. In the chapter, we will study the effect of possible initialization strategies for multi-instance learning using the proposed MLLR as an example.

Here we focus on an extreme case of weakly supervised object recognition on Fine-Grained Visual Categorization (FGVC) [36, 157, 172]. FGVC is a highly challenging task which aims to classify categories at the subordinate level, for example different species of animals [13, 73, 103, 150], plants [6, 78, 101], and man-made objects [97, 131, 164]. Due to the subtle difference between subordinate categories and the large intra-class variance introduced by various object scales, poses and occlusions, the intra-class variance in fine-grained recognition could be even larger than inter-class variance. Most of existing FGVC algorithms relied on strong supervision [12, 172] or employed human-in-the-loop approaches [18, 37] to introduce rich annotations including object-level bounding boxes and part-level part landmarks. However, since the labeling process for rich annotations is extremely time-consuming and sometimes requires domain expertise, it is very important to investigate the possibility to recognize fine-grained categories without manually-labeled annotations, which is rarely studied in the literature [160].

To perform multi-instance learning on weakly supervised object recognition tasks, the standard approach is to regard images as bags with multiple instances denoting possible regions of interest for an object. The learning hypothesis relies

on the predicted object location. In particular, for FGVC, it has proved that by retrieving the location of objects and performing background removal, the performance could be significantly improved [22]. Therefore, MIL methods which iteratively update object locations and train classifiers based on the inferred object instances could conceptively enjoy good potential for solving weakly supervised FGVC problems.

However, we believe that standard routine for MIL cannot be applied to weakly supervised FGVC problems directly due to the following two outstanding concerns: 1) the lack of real "negative" training examples. Weakly supervised object recognition methods usually involve a localization step to detect foreground objects from the background, and a classification step to predict object labels. In FGVC, the visual appearance of subordinate categories could be very similar to each other; meanwhile, the intra-class variance could be large due to multiple factors including pose, scale and occlusion. Consequently, the key characteristic separates one category from another is not always able to distinguish it from the background, leading to a conflict between the goals of the localization step and the classification step. 2) Small amount of training data and high intra-class variance in FGVC significantly degrade the accuracy of object localization. Localization errors further propagate to the classification process. As a result, the non-convex MIL objective function is prone to be stuck in a bad local minimum.

Motivated by the above observations, we propose a new method for weakly supervised FGVC that aims to explore inter-class relationships to perform an effective initialization and thus generalize the traditional MIL framework. Based on the standard pipeline for weakly supervised object recognition, the problem is decomposed into two main phases: localization and classification. The key is that due to the lack of real "negative" examples in the dataset, we adopt an unsupervised co-localization method to locate objects, in which training examples in each category are regarded as a set of related images. The localization process of similar subordinate categories now contributes to each other through a novel multi-task (MTL) discriminative clustering algorithm that conducts co-localization on similar categories simultaneously. Afterwards, a multi-instance learning (MIL) algorithm is conducted in the classification stage to distinguish subordinate categories explicitly. Localization results in the previous phase are

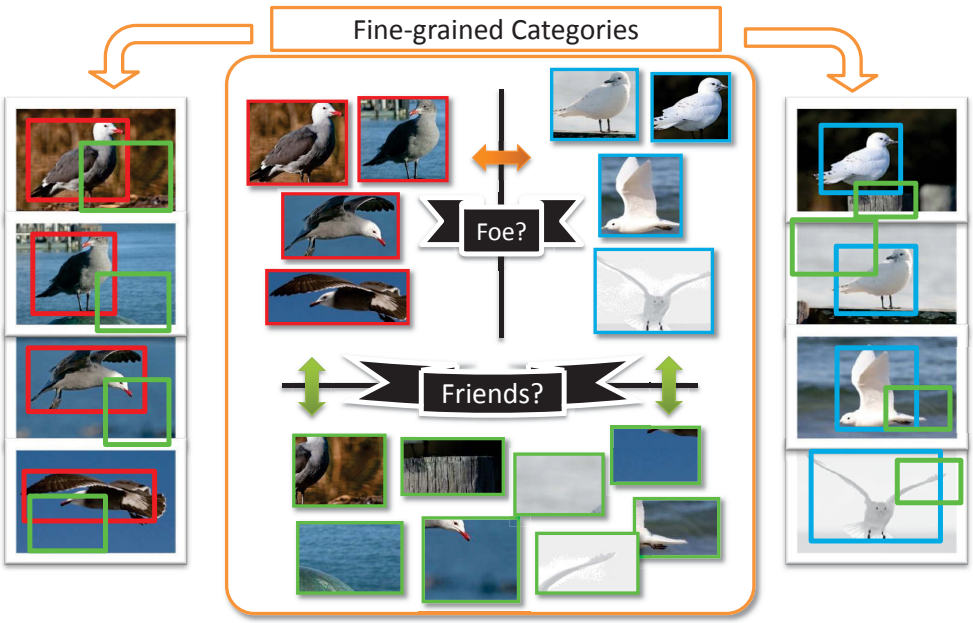


Figure 5.1: Illustration of the effect of inter-class relationships in fine-grained categorization under weakly supervised settings. In the proposed algorithm, fine-grained categories first act as "friends" in the localization phase against varied backgrounds, then turn back to "foes" in the following classification phase.

used as an effective initialization for the MIL algorithm to avoid bad local minima. In summary, as shown in Figure 5.1, the fine-grained subcategories first act as "friends" in the localization stage; then turn back as "foes" in the classification stage. Moreover, this process provides object-level cues that enables domain-specific fine-tuning of deep neural networks which significantly boosts the performance. Extensive results on three well-known FGVC datasets CUB-200-2010 [157], CUB-200-2011 [150], and Stanford Dogs [73] demonstrate the effectiveness of the proposed method in both classification and localization accuracies. We will present details about the proposed localization algorithm, classification method and experimental results in the following.

5.2 General Initialization Strategies for MLLR

We employ the proposed Multinomial Latent Logistic Regression (MLLR) as the learning algorithm for weakly supervised fine-grained visual categorization in this chapter. Recall that the objective function of MLLR is defined as:

$$l(W) = -C \sum_{j=1}^{M} \log \frac{\exp[\Psi(I_j; w_{y_j})]}{\sum_{k=1}^{K} \exp[\Psi(I_j; w_k)]} + R(W), \tag{5.1}$$

where $R(W)$ is an L1-norm regularizer, I_j is the j-th training image, y_j is its ground-truth label, and $\Psi(I; w)$ is a score function calculating the best score of all boxes in image I according to model parameters w:

$$\Psi(I_j; w_k) = \max_{b \in B_j} [w_k \phi(b)]. \tag{5.2}$$

The objective function in (5.1) is non-convex. As a result, MLLR has a crucial problem that it is sensitive to the initialization status. In general, there are several strategies to initialize latent variables, *i.e.*, to locate objects from a set of candidate regions.

The simplest approach is to assume an object-in-the-center prior and use a bounding box at the center to locate objects [115]. Based on this simple strategy, recent object proposal approaches provide an alternative by assigning category-independent objectness scores to the candidate regions. The top-scored region as the most salient patch could be selected to initialize MIL algorithms, as adopted by Deselaers *et al.* [38] in their object recognition method. However, this strategy is highly dependent on the effectiveness of object proposal methods; in most cases, it is impossible to achieve a high accuracy through the first top-scored object proposal only. As the goal of object proposal methods is to provide higher coverage on detecting objects with a smaller number of proposals, usually it is impossible to achieve a high accuracy using only one object proposal.

Another possible solution is to pre-train classifiers using original full images without detecting accurate object location; then use the learned classifiers to select the most discriminative region. Despite its success in scene recognizing tasks [102], it is argued that the strategy could be erroneous in FGVC. Consider-

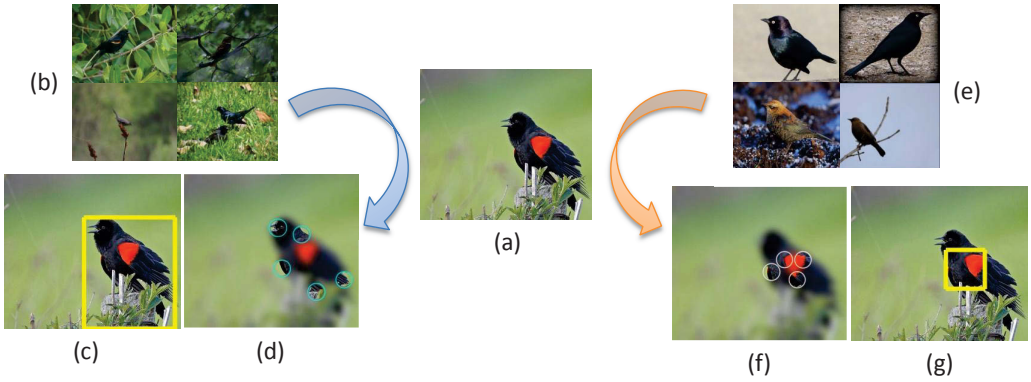


Figure 5.2: Illustration of the difference of discriminative regions for detecting objects from the background and classifying fine-grained categories. For an image of *Red winged Blackbird* in (a), object parts such as forehead, eyes, back and tail make it possible to detect the object, as shown in (c),(d). On the contrary, the subcategory different from other ones mainly from its red wing, resulting in a much smaller region of interest, as shown in (f),(g).

ing that fine-grained categories are very similar to each other, as shown in Figure 5.2, the most discriminative region for classifying fine-grained categories usually appears in a specific part, such as heads for a bird category and windows for an architectural style. As a result, categorical classifiers tend to fire on smaller part regions and cannot explicitly locate the whole object. Therefore, the strategy suffers on problems from varied view points and occlusion.

Instead, we want to seek a method that utilizes the high similarity between fine-grained categories to facilitate the localization process. This is achieved by a novel multi-task co-localization algorithm which will be detailed in the next section.

5.3 Initialization via Multi-task Co-localization

In order to prevent the multi-instance learning process from being stuck in a bad local minimum, we propose a method for initializing the latent variables, *i.e.*, generate object detectors from image-level labels, which can be modeled as a

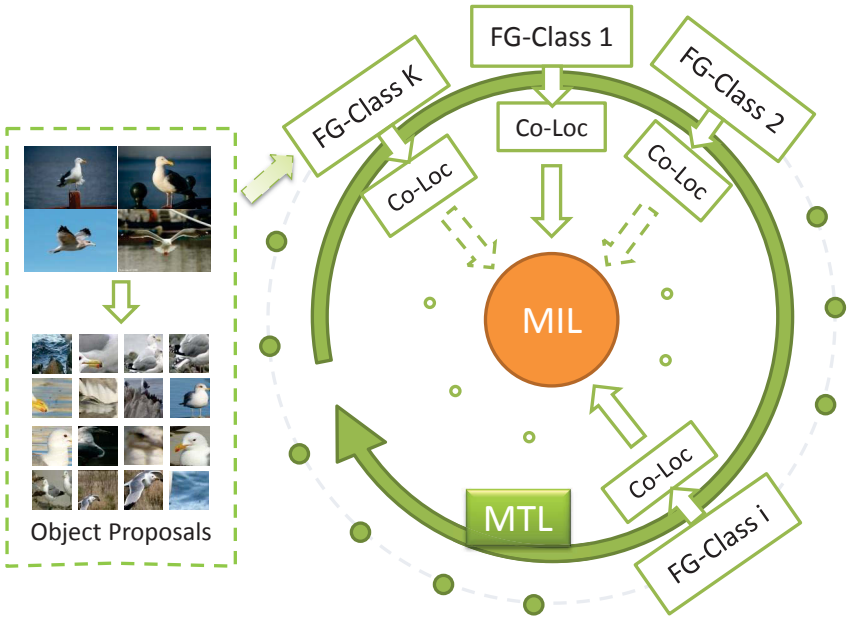


Figure 5.3: Illustration of the proposed method for weakly supervised fine-grained recognition. The first co-localization phase aims to detect foreground regions from the background. We propose a multi-task algorithm to perform co-localization on multiple subcategories simultaneously. The localization results are then employed to initialize a multi-instance learning process to learn the final object classifiers.

weakly supervised object localization problem. We solve this problem under a more relaxed scenario: the goal is to find and localize a common object within a set of images; however, there is no knowledge of what the object is, and no given negative images for which we know do not contain the common object. In fact, the relaxed setting is closer to FGVC where no real "negative" class exists. This problem is termed co-localization, which shares the same type of input as co-segmentation [147].

Our co-localization algorithm is inspired by discriminative clustering [70, 71, 133], in which a classifier is trained to distinguish the common object from the background, while enforcing consistency in appearance among foreground pixels or regions. Based on it, we propose a new multi-task learning algorithm that conducts co-localization on a set of similar categories. Figure 5.3 presents an overview of the proposed algorithm.

5.3.1 Preliminary

Given a set of N fine-grained training images $\mathcal{I} = \{I_1, I_2, ..., I_N\}$, our goal is to train fine-grained classifiers using only image-level training labels, *i.e.*, $\mathcal{Y} = \{y_1, y_2, ..., y_N\}$. Suppose there are K subordinate categories in the dataset, *i.e.*, $y_i \in [1, .., K]$. We decompose this weakly supervised learning problem into two phases: localization and classification. The first phase aims to obtain generic object detectors (such as a *bird* detector), while the detected objects are further classified in subordinate-level (such as *California gulls* and *Western gulls*) in the second phase.

Region proposals. We employ the multi-scale combinatorial grouping (MCG) [7] algorithm to extract category-independent proposals. Besides the need of objectness scores to initialize object locations, the proposed algorithm is agnostic to the particular region proposal method. Here we use bounding-box-level proposals and further reduce the number of object proposals by employing a non-maximum suppression (NMS) strategy and pruning small bounding boxes. The process results in averaging 15-20 bounding-box-level object proposals per image.

Feature representation. Convolutional neural networks (CNNs) recently delivered significant improvement over traditional methods in the ImageNet recognition challenge [76]. As shown by Razavian *et al.* [114], features extracted using ImageNet pre-trained CNNs can transfer to a variety of visual recognition tasks and achieve reasonable results. Inspired by this discovery, we employ CNN feature extractors in our work and extract a 4096-dimensional feature vector from each object proposal using the CNN implementation of [25]. In order to compute feature vectors from arbitrary-shaped object proposals, we warp the pixels in a region proposal into a bounding box with the required size (a fixed size of 227×227 in our implementation), regardless of the region size and aspect ratio. Features are then computed by forward propagating a mean-subtracted 227×227 RGB image through the CNN architecture.

In detail, for each image $I_j \in \mathcal{I}$, we generate a set of candidate regions $B_j = \{b_{j,1}, b_{j,2}, ..., b_{j,m_j}\}$. Let $B = \{B_j\}$ be the set of all candidate bounding boxes with the total number as $M = |B| = \sum_j m_j$. Each candidate bounding box $b_{j,l}$ is then associated with a 4096-dimensional feature vector $\phi(b_{j,l}) \in \mathbb{R}^d$ and a

category-independent objectness score $s_{j,l} \in \mathbb{R}$.

5.3.2 Co-localization by discriminative clustering

We briefly review discriminative clustering on terms and motivations in the objective function, and highlight several unique modifications we made in our implementation.

Optimization variable. Given a set of images $\mathcal{I}$ and bounding boxes B_j for each image $I_j \in \mathcal{I}$, each region bounding box $b_{j,l}$ is associated with a binary variable $z_{j,l}$, which is equal to 1 if $b_{j,l}$ is a positive box containing the common object, and 0 otherwise. Denote $z \in \{0,1\}^M$ by stacking $z_{j,l}$ for all the object proposals. The goal is to find z that minimize an energy function combining the following terms.

Objectness prior. A prior term conveying whether a box is an object or not is introduced using category-independent objectness scores in MCG. By stacking all $s_{j,l}$ into a vector s_{obj}, a linear term that penalizes less salient boxes is given as:

$$E_{obj} = -z^T \log s_{obj} \tag{5.3}$$

Box similarity. Boxes with similar appearance should have a large chance coming out with the same label. We define a similarity matrix S to represent local appearance similarities between boxes. For any pair (i,j) of boxes, $S_{i,j}$ is defined as:

$$S_{i,j} = 1 - \bar{d}(\phi(b_i), \phi(b_j)), \tag{5.4}$$

where $\bar{d}$ is the normalized Euclidean distance between two feature vectors. The box similarity term is then defined as:

$$E_{bs} = \frac{\mu}{M} z^T L z. \tag{5.5}$$

Here L is the normalized Laplacian matrix $L = I - D^{-1/2} S D^{-1/2}$, where I is the identity matrix and D is the diagonal matrix composed of the row sums of S; M is the number of all boxes, μ is a free parameter.

Box discriminability. Although co-localization is an unsupervised problem, [70] argued that a discriminative clustering algorithm which aims to classify

objects from the background would contribute to the localization performance. More precisely, a discriminative classifier finds the optimal parameters $\alpha \in \mathbb{R}^d$ and $\beta \in \mathbb{R}^d$ that minimize

$$E_{dc}(z, \alpha, \beta) = \frac{1}{M} \sum_{i=1}^{M} l(z_i, \alpha, \beta) + \lambda ||\alpha||. \tag{5.6}$$

Here we adopt the logistic loss function:

$$l(z_i, \alpha, \beta) = -\log \frac{\exp[z_i(\alpha^T \phi(b_i) + \beta)]}{1 + \exp(\alpha^T \phi(b_i) + \beta)}. \tag{5.7}$$

Cluster size balancing. Discriminative clustering approaches have a classical problem that assigning the same labels to all the boxes leads to perfect separation. [71] introduced a penalty term to encourage the proportion of points per class through entropy:

$$H(z) = -\frac{\sum_i z_i}{M} \log \frac{\sum_i z_i}{M} - (1 - \frac{\sum_i z_i}{M}) \log(1 - \frac{\sum_i z_i}{M}). \tag{5.8}$$

Joint formulation. Combining the terms in (5.3)(5.5)(5.6)(5.8), the joint formulation of the optimization problem can be denoted as:

$$\min_{z \in \{0,1\}^M} \left[\min_{\substack{\alpha \in \mathbb{R}^d \\ \beta \in \mathbb{R}}} E_{dc}(z, \alpha, \beta) \right] + E_{obj}(z) + E_{bs}(z) - H(z). \tag{5.9}$$

5.3.3 Multi-task discriminative clustering

The co-localization method described before is designed for detecting common objects from a set of related images. However, a slightly different scenario exists in weakly supervised FGVC - images are from a series of subordinate categories and we are aware of the category label of each image. In this situation, one could perform co-localization on each subordinate category independently, or assume that all images belong to one generic category and run co-localization on all of them. Nonetheless, both of the strategies have certain problems: the first one ignores underlying relationships between subordinate categories, while the latter one requires a large memory to compute and store the similarity matrix for all

pairs of boxes.

Considering that the localization process of all the subordinate categories share the same goal of classifying foreground objects from the background, multi-task learning (MTL) is a natural choice here to take advantage of the relationship between categories without extensive memory requirement. We thus introduce a new multi-task discriminative clustering (MTL-DC) term that regards the co-localization problem of each subordinate category as one single task. MTL is particular effective in our task due to the small number of images in each fine-grained category and the employment of high-dimensional CNN feature representations.

Specifically, assume that the input images $\mathcal{I}$ belong to K classes. Denote $\mathcal{I}_k$ the set of images and B_k the set of all object bounding boxes in class k, where $|B_k| = M_k$. Let $z = [z^{(1)}, z^{(2)}, ..., z^{(K)}]$ be the ensemble of z's of all classes. Altogether we have K sets of classifiers: $A = [\alpha_1, \alpha_2, ..., \alpha_K] \in \mathbb{R}^{d \times K}$, $B = [\beta_1, \beta_2, ..., \beta_K] \in \mathbb{R}^K$. The multi-task discriminative clustering term is then defined as:

$$E_{mtdc}(z, A, B) = \sum_{k=1}^{K} \frac{1}{M_k} \sum_{b_i \in B_k} l(z_i, \alpha_k, \beta_k) + \lambda ||A||_*. \tag{5.10}$$

Introducing a trace-norm form $||A||_*$ enforces a low-rank assumption on the model parameter matrix A. As a result, the model parameters are optimized simultaneously and a low-rank structure is achieved following underlying relationships between multiple subordinate categories. The final objective function is given as:

$$\min_{z \in \{0,1\}^M} \left[\min_{\substack{A \in \mathbb{R}^{d \times K} \\ B \in \mathbb{R}^K}} E_{mtdc}(z, A, B) \right] + \sum_{k=1}^{K} [E_{obj}(z^{(k)}) + E_{bs}(z^{(k)}) - H(z^{(k)})]. \tag{5.11}$$

5.3.4 Optimization

We optimize (5.11) using an expectation-maximization (EM) procedure following [71]. The first step is to obtain a linear relaxation of the objective function by relaxing the Boolean constraints on variable z to continuous, linear constraints between 0 and 1. Here $z_{j,l}$ can be interpreted as the probability of a bounding box $b_{j,l}$ being an object. Then the EM procedure is detailed as:

M-step. For some given values of z, minimize $E_{mtdc}(z, A, B)$ in terms of (A, B). It is a standard logistic regression problem with trace-norm regularizers, which can be efficiently optimized using an accelerated gradient method [27].

E-step. Given model parameters (A, B), the objective function is convex in z, and is thus minimized by a simple projected gradient descent method.

In practice, the EM algorithm usually converges within 10 iterations. We then obtain the optimal object location in each image as the box with the largest score according to the model parameters of the respective category.

As an initialization method, the proposed MTL-DC algorithm aims to distinguish common foreground objects from the background. Therefore, the fact that fine-grained categories are largely similar to each other becomes no longer a barrier; instead, it contributes to the co-localization process by sharing information through the multi-task learning strategy. Moreover, multi-task learning alleviates the problem caused by insufficient training data in each subcategory.

5.3.5 Fine-grained classifiers

Although the co-localization method discussed before is supposed to improve the localization accuracy, it is still not guaranteed that the learned object detectors could localize objects perfectly or are optimal for classifying fine-grained categories. The multi-instance learning strategy, therefore, is used to further refine co-localization results and train the final fine-grained object classifiers.

Suppose that the FGVC classifiers are denoted as $W = [w_1, w_2, ..., w_K] \in \mathbb{R}^{d \times K}$. For each training image I_j, the exact object location is modeled as a latent variable; the latent variable updating process thus equals to finding an optimal bounding box from candidate object proposals B_j. Note that these classifiers are different from those defined in Section 5.3: the goal here is to distinguish fine-grained categories, while co-localization focuses on extracting foreground regions from the background.

The weakly supervised learning problem here can be modeled as a multi-class classification problem with latent variables, where the latent variable (object location) is updated using a maximization inference procedure: an ideal setting for the proposed MLLR. As the objective function of MLLR is non-convex and

can be formulated as the difference of two convex functions, we solve MLLR using a convex-concave procedure by defining a convex auxiliary function that bounds the exact objective. The auxiliary function is constructed by fixing the latent variable assignments for all training examples with respect to their training labels. Optimization is then performed in an iterative way: model parameters and latent variables are updated iteratively with the other one fixed. The detailed optimization process can be found in Section 3.2.

5.4 Experiment

In this section, we will present experimental results and conduct comparison studies with state-of-the-art methods on three widely used FGVC datasets. Specifically, through quantitative studies, we will discuss the impact of each stage in the proposed method.

5.4.1 Dataset and implementation details

Dataset. Experiments are performed on three challenging FGVC datasets, the Caltech-UCSD Bird-200-2010 dataset (CUB-200-2010) [157], the CUB-200-2011 dataset [150], and the Stanford Dogs dataset [73]. CUB-200-2010 contains 200 bird species, with about 30 images per category; half of them are adopted for training. We also use a subset of 14 classes from the Vireo and Woodpecker family (CUB-14) following [46,111,166] to explicitly study the influence of model parameters. CUB-200-2011 dataset is an updated version of CUB-200-2010 with roughly 60 images per category; it is widely-used in recent works on FGVC. The Stanford Dogs dataset contains 120 bog breeds with around 100 train images and 70 test images per category. For all the datasets, we follow the original training/testing split provided by the authors. Our study is conducted on the scenario under the weakest supervision, *i.e.*, except for image-level labels, no additional annotations such as bounding boxes and part landmarks are used in both the training and testing stage.

Performance Measures. The performance of the proposed method is evaluated by the classification accuracy on the three FGVC datasets. Besides, we

also present localization results for reference, which are measured by the *CorLoc* evaluation metric defined as the percentage of correctly located objects compared to ground-truth ones according to the PASCAL criterion: $\frac{area(b_p \cap b_{gt})}{area(b_p \cup b_{gt})} \geq 0.5$.

Classification Process. Given training images, the proposed multitask co-localization algorithm is performed on the extracted object proposals to obtain localization results. They are further used to initialize the training of a multinomial latent logistic regression (MLLR) model to generate fine-grained classifiers. Testing stage involves no co-localization process; extracted region proposals are directly fed into the resultant MLLR model to obtain classification results.

Implementation Details. We use a single scaled version of MCG called SCG to extract object proposals [7]. Following the standard approach, CNN models pre-trained on ImageNet are employed to extract deep features. In particular, the standard seven-layer architecture is used for the CUB-200-2010 and the Stanford Dog dataset, for which we employ the implementation of the VGG-f model [25], and use the CNN's *fc6* and *fc7* layer for the bird and the dog dataset respectively. For the CUB-200-2011 dataset, a GoogleNet [132] is adopted in order to provide comparable results with latest methods. Typically the object proposal and feature extraction process require 10-20 seconds per image. After feature extraction, the co-localization phase requires ~ 5 hours, while the time cost for training the final classifiers is trivial.

5.4.2 Localization results

The proposed method regards the co-localization problem of each category as a single task, and implements a multi-task learning (*MTL*) approach to employ shared information among subordinate categories. Therefore, we compare our results with two baseline methods: 1) running co-localization on each category independently; 2) regarding all subcategories as a generic category and performing co-localization on it. We denote the two baselines as method *SINGLE* and method *ALL* for brief respectively. Since the latter strategy requires a large memory cost and thus is not practical on large datasets, this comparison is only conducted on the CUB-14 subset.

We run 5 times of random initialization and record the average results. All

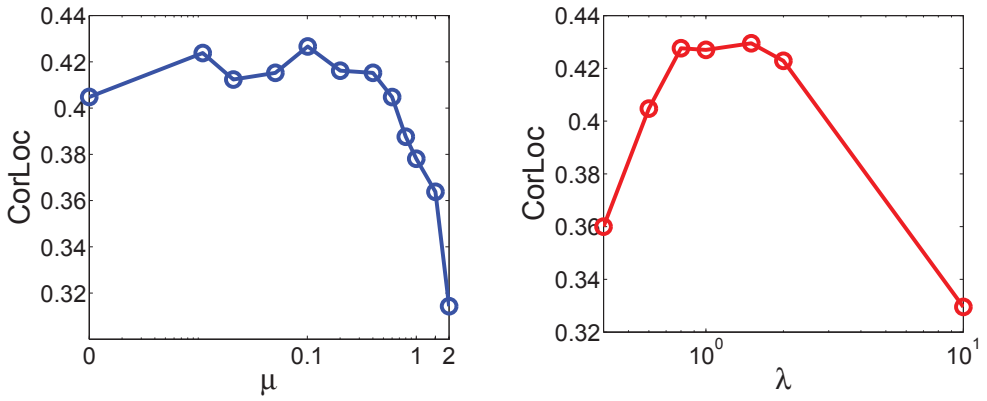
Table 5.1: *CorLoc* results for different co-localization strategies on the CUB-14 dataset. MCG denotes the baseline where boxes with the top objectness scores obtained by MCG were adopted without performing co-localization methods. "@n" denoted the best result among top-n candidates.

	MCG	*SINGLE*	*ALL*	MTL
Avg CorLoc@1	31.43	37.81	41.90	42.67
Avg CorLoc@5	60.00	65.52	67.24	68.57

methods are tuned within a range of model parameters μ and λ to gain the best performance. Table 5.1 shows comparison results of baseline methods. Compared to the strategy that simply assigns top-scored candidate regions given by MCG as the objects' location, co-localization methods achieve significantly higher localization accuracy. Method *ALL* further outperforms method *SINGLE* by a remarkable margin. This shows the fact that considering the small number of training images available in each subcategory, it is better to assume all subordinate categories as a generic object "bird" and perform co-localization on all of them than each of them independently. Multi-task learning shows its priority by obtaining even slightly higher accuracy than method *ALL* without extensive memory requirement, proving that the inter-class relationship between subordinate categories indeed contributes to the performance of object localization.

We have further investigated the influence of tuning model parameters in the co-localization algorithm, including the weight of box similarity μ, and the amount of regularization in discriminative clustering λ. In general, localization results are consistent when μ and λ are near 0.1 and 1, as shown in Figure 5.4. It is worth noting that a special case of $\mu = 0$ coincides with the category level method in [22], where a foreground/background classifier is trained based on the discriminative term only.

The co-localization approach significantly improves localization accuracy. However, localization results obtains from weakly supervised settings are still not reliable enough to detect foreground object regions perfectly. As shown in Table 5.1, only about 40 percent of the top-scored bounding boxes after co-localization could



(a) (b)

Figure 5.4: The impact of model parameters μ and λ. In the first figure, λ was fixed as 1, and the second figure set $\mu = 0.1$.

locate objects with a high confidence. A relaxation of returning five candidates after co-localization increases the number to 68%, indicating that it is still of need to allow the object locations to be updated in the classification stage. Therefore, in our implementation, the results of co-localization are used as an initialization of the following multi-instance learning (MIL) step to refine localization results, and to avoid bad local optima at the same time.

Figure 5.5 shows comparable studies of the object localization results in multiple steps of the proposed algorithm. As an initialization method, the resultant top scored candidates by MCG object proposal method usually appear in specific part of the target object with the highest saliency. After performing co-localization, the resultant bounding boxes tend to capture holistic information of the objects, resulting in larger detected regions than the top scored windows by MCG. The co-localization results are further refined by the following multi-instance learning step which aims to classify objects at the subordinate level. As shown in the third column in Figure 5.5, the MIL classifiers, on the contrary, captures detailed part information that is valuable for distinguishing fine-grained objects. As a result, performing MIL does not always guarantee a boost of localization results; in practice, localization results after MIL is approximately on par than that before MIL. The last column showes localization results by performing SVM classifiers on the whole images without running a previous localization process. Such base-

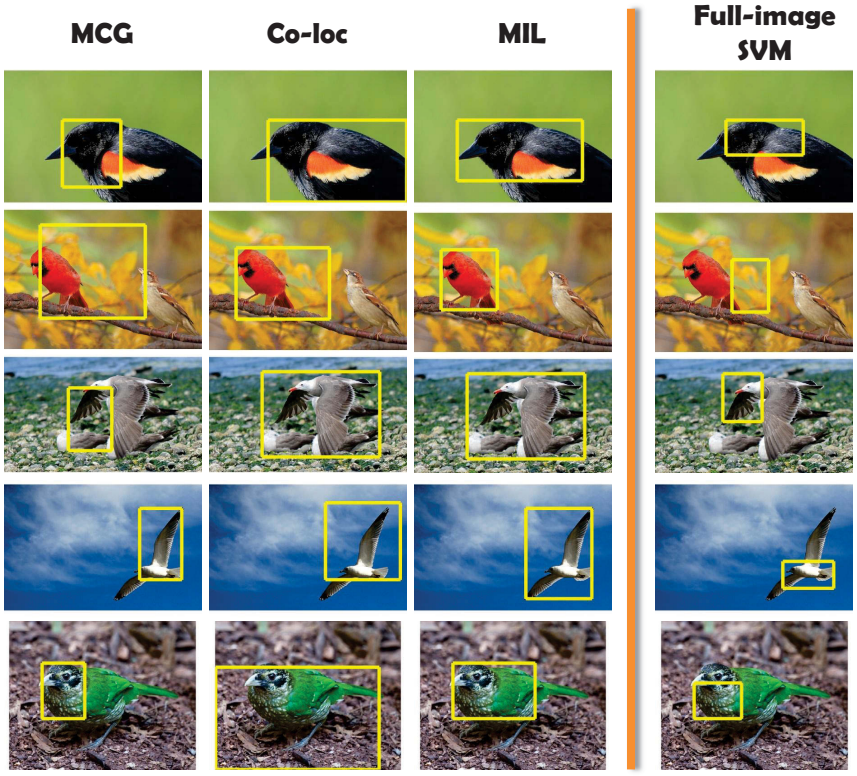


Figure 5.5: Localization results in different stages. Column 1 to 4: best-scored MCG candidate; results after performing co-localization; results after performing multi-instance classification; best scored bounding box according to SVMs trained using full images.

line fails to detect objects robustly, which proves the assumption that classifiers distinguishing whole images are not always able to capture the exact location of objects.

Table 5.2 summarizes the final localization results of our algorithm on the three datasets. Figure 5.6 shows examples of the final localization results after performing MIL.

5.4.3 Classification results

We hereafter analyze the classification results from three perspectives: comparison with baselines in which different localization strategies are employed, study-

Table 5.2: Localization results for CUB-200-2010, CUB-200-2011 and Stanford Dogs.

CorLoc@1 (%)	CUB-2010	CUB-2011	Standford Dogs
MCG	40.56	48.65	56.17
Co-localization	48.69	51.96	66.68

ing the effect of fine-tuning CNNs using varied strategies, and finally showing the comparison with state-of-the-art results.

Comparison with baselines. We first present an ablation study to analyze the importance of each step in the proposed method. Take the results of CUB-200-2010 as an example, as shown in Table 5.3, both the co-localization phase and the multi-instance learning phase contributed to the final performance. The simplest approach of training SVM classifiers on features extracted from the whole images obtains a 31% accuracy. A five-percent gain is then achieved by training SVMs on features extracted from foreground regions detected by co-localization. Multi-instance learning delivers an additional 4% boost by allowing re-assignment of object location according to discriminative information. On the other hand, classification results also reveal the importance of a well-designed initialization strategy for multi-instance learning methods. Models initialized using features from whole images or top-scored object proposals both result in lower accuracy than that using the proposed co-localization strategy.

The final co-localization initialized multi-instance method obtains a 40% accuracy under the weakest supervised setting. As a comparison, when the ground-truth bounding boxes are given at the training stage, classifiers using the same feature representation achieve a 48% accuracy. Hence, our weakly supervised algorithm can achieve half of the performance improvement brought by bounding box supervision. The number is impressing considering the difficulty of weakly supervised FGVC task and the challenging CUB-200-2010 dataset.

The effect of fine-tuning CNNs. Previous results are obtained by adopting pre-trained CNNs only as a feature extraction method; thus, other feature extraction methods such as SIFT [95] and KDES [15] could also be adopted in the

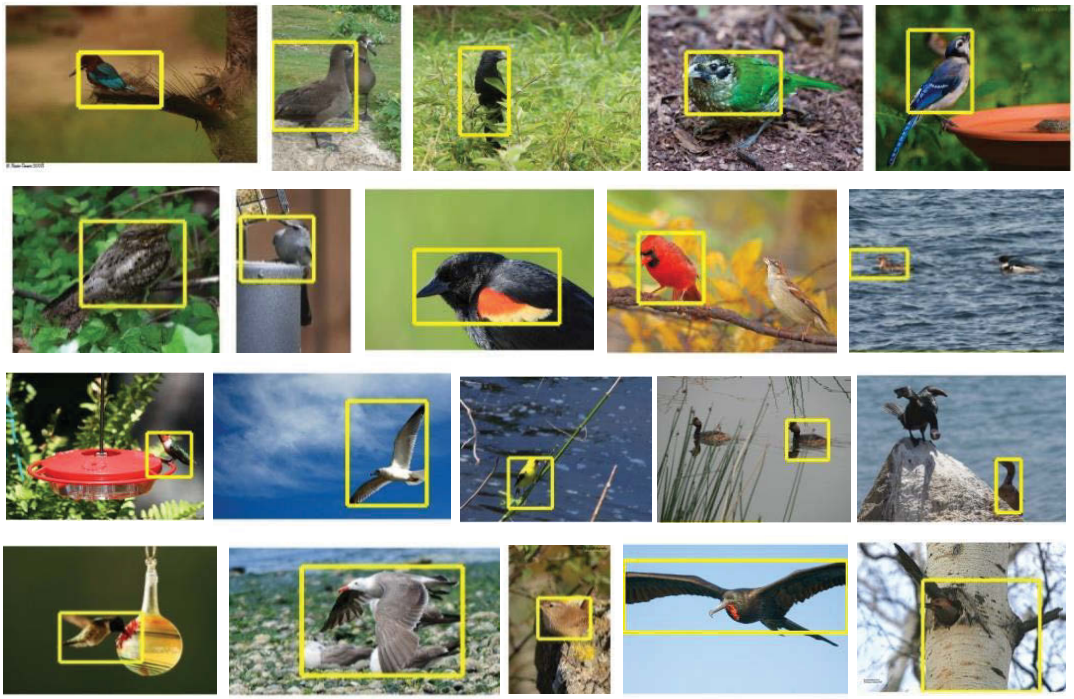


Figure 5.6: Example localization results for testing images after performing multi-instance learning. The rightmost column shows cases in which the proposed method failed to classify correctly due to multiple objects, uncommon object pose, and background clutter.

proposed method without any modifications. However, this also indicates that the performance could be further improved by performing specific designs regarding the characteristics of CNNs. It has been proved that fine-tuning CNN models could significantly boost the classification performance. Therefore, the standard approach of employing deep CNN features on object recognition tasks is to pre-train a CNN model on ImageNet, and then fine-tune the model on the target domain. However, under weakly supervised settings, there is no straightforward way to fine-tune CNNs to represent objects, which requires explicit object-level labeling information.

A naive solution is to fine-tune CNNs using whole images and their associated image-level labels. Such solution is definitely not optimal, since features learned using whole images contain redundant background information and thus are not

Table 5.3: A detailed comparison with baselines of different localization strategies and classification methods on the CUB-200-2010 dataset. Row 1-3 show results by training classifiers solely on the detected foreground regions. Row 4-6 show results by performing a multi-instance learning (MIL) approach initialized by the respective localization results. The final row presents an upper bound of our algorithm by using ground-truth bounding box supervision.

ID	Localization	Classification	Accuracy
1	whole image	SVM	31.42
2	MCG top objectness	SVM	25.02
3	Co-localization	SVM	36.00
4	whole image	MIL	35.97
5	MCG top objectness	MIL	35.21
6	Co-localization	MIL	**40.16**
7	GT bounding box	SVM	48.17

always able to distinguish foreground objects effectively. As an alternative, we propose to use the result of co-localization as the intermediate labels for fine-tuning CNNs. Specifically, in our co-localization process, each object proposal will output a score by discriminative clustering. A positive score indicates that the object proposal is classified as foreground object; thus it should be regarded as a positive example when fine-tuning CNNs. In practice, we labeled object proposals with positive localization scores as training samples for their respective classes, and fine-tuned CNNs using a 201-way classification layer for the CUB-200-2011 dataset accounting for the class "background". A similar approach is adopted in R-CNN [57]. However, R-CNN assumes that the ground-truth bounding boxes of training examples are given, and labels object proposals with a large overlap to ground-truth object bounding boxes as positive when fine-tuning CNNs. On the contrary, in this paper, we tackle the weakly supervised problem where only image-level labels are presented. As a result, the labels for fine-tuning CNNs are generated by the previous co-localization process.

Table 5.4 shows classification results on the CUB-200-2011 dataset by fine-tuning CNNs using various strategies. Strategies we investigate include using

Table 5.4: Effect of fine-tuning CNNs. We achieved an accuracy of 77.37% on the CUB-200-2011 dataset under the weakly supervised scenario.

localization classification	whole image SVM	co-localization MIL	GT BBox SVM
w/o ft	61.25	65.21	67.32
ft on img	72.67	75.72	77.20
ft on BBox	73.31	**77.37**	78.79

pre-trained models only, and fine-tuning on whole images or the predicted object bounding boxes. Regardless of the specific fine-tuning strategy, the proposed method consistently outperforms baselines that train SVM classifiers on features extracted from whole images by a large margin. The last column in which ground-truth bounding boxes are given in both training and testing phase is provided as an upper bound of the performance. For example, in the scenario similar to the first experiment where no fine-tuning process is conducted, the proposed method achieves 65.2% accuracy on the CUB-200-2011 dataset, while training SVMs on whole images obtains 61.2% and the upper bound accuracy using this feature representation is 67.3%.

Results also reveal that fine-tuning definitely contributes to the classification results - the simplest baseline strategy that directly relies on image-level labels boosted the classification performance by approximately ten percents to 75.7%. After generating object locations using the proposed co-localization method, fine-tuning CNNs on these predicted object-level labels further raises the accuracy to 77.4%, even outperforming the upperbound of the baseline fine-tuning strategy (77.2%). The results prove that fine-tuning CNNs under object-level supervision, even when the labels are not strictly accurate, could achieve better performance for classifying fine-grained categories.

Comparison with state-of-the-arts. Table 5.5 presents comparison results of our method and several state-of-the-art FGVC methods. As our method is agnostic to the particular CNN architecture, one can always employ stronger deep networks as the feature extractor without additional computational cost

Table 5.5: Performance comparison to the state-of-the-art results in the literature with or without the use of ground-truth bounding boxes at the training stage.

Approach	CUB-200-2010	CUB-200-2011	Stanford Dogs	GT BBox used?
Symbotic [21]	47.3	59.4	45.6	yes
Alignment [55]	-	67.0	57.0	yes
CNNaug-SVM [114]	-	61.8	-	yes
NoPart [74]	-	82.0	-	yes
BiCoS [20]	16.1	-	-	no
TriCoS [22]	25.5	-	26.9	no
Overfeat [114]	29.7	-	68.0	no
DeepCNN [9]	-	67.2	-	no
Attention [160]	-	77.9	-	no
Constellation [126]	-	81.0	68.6	no
Biliear [92]	-	84.1	-	no
Our Method	40.2	77.4	71.4	no

during the co-localization and classification phase. Our results are significantly better than previous state-of-the-arts, such as TriCoS [22] and Overfeat [114]. However, several latest weakly supervised FGVC methods [92, 126] have achieved better results than ours on the CUB-200-2011 dataset. It reveals the effectiveness of part discovery strategies that is particular important in the bird dataset to model object parts. Our algorithm that aims to locate object bounding boxes more accurately could be supplementary to these part-based methods.

Meanwhile, most state-of-the-art methods such as R-CNN [57] and Attention Model [160] employed selective search [141] to extract object proposals. Such process typically generated 1000-2000 object proposals for each image. It is remarkable that the proposed method could produce comparable results using far less object proposals (usually 15-20) than selective search; thus the efficiency of both feature extraction and CNN fine-tuning are largely improved.

We also present experimental results on the Stanford Dogs dataset. Images in this dataset show a strong object-in-the-center bias; in most cases, each image contains only one significant object. Under this more ideal setting, the proposed method achieves a 67% *CorLoc*, much higher than that of the bird dataset. As a result, the classification accuracy under weakly supervised settings (71.4%) is

nearly identical with that given the ground-truth bounding boxes (71.6%). This result reveals that bounding-box-level supervision is not indispensable for datasets that contain mostly large and significant objects such as the Stanford Dogs.

5.5 Summary

In this chapter, we have studied weakly supervised learning on an extremely challenging problem - fine-grained visual categorization. Considering that the objective function of the proposed MLLR has a semi-convex property, we carefully design an initialization strategy for the algorithm, by proposing a novel multi-task co-localization algorithm to localize a set of similar objects. Taking advantage of the fact that subordinate categories in fact belong to a more generic concept, the proposed method outperforms baseline strategies including modeling each subcategory independently or regarding them as a unified category, and achieves competitive results on fine-grained categorization benchmarks.

In the next chapter, we aim to further boost the classification performance of fine-grained visual categorization through the employment of a larger scale of training resources (web data) and more detailed annotations (strongly supervised datasets). The objective function of MLLR, meanwhile, will be slightly modified to support a combination of strongly and weakly supervised training data.

Publications Related to This Chapter

1. **Zhe Xu**, Dacheng Tao, Shaoli Huang, Ya Zhang. Friend or Foe: Fine-grained Categorization with Weak Supervision. *IEEE Transactions on Image Processing (TIP)*, 2015 (under review)

Chapter 6

MLLR for Webly Supervised Learning

It is widely acknowledged that one of the most significant advantages of weakly supervised learning appears in its scalability. In order to produce competitive results while being able to scale the algorithm to a larger scale of webly supervised learning, an intuitive solution is to use a small set of strongly supervised data as an initialization to guide the webly supervised learning process, especially considering the noisiness of labels acquired from the web. In this chapter, we still study the problem of fine-grained visual categorization, but using a sophisticated strongly supervised algorithm trained on existing datasets with rich annotations to introduce additional information in weakly supervised learning. The proposed MLLR, therefore, is generalized to train models on a combination of weakly supervised and strongly supervised examples, which leads to significant higher results.

6.1 Introduction

Webly supervised learning [28, 29, 72, 120], namely learning visual representations from web data, has drawn much attention in object recognition recently for its ability to scale learning paradigms to a higher level. It is also the ultimate goal of weakly supervised learning, which acquires data with nearly no cost and achieves

great scalability. Although webly supervised learning has seen monumental developments recently, when focusing on a particular task, in most cases webly supervised methods have struggled to match up against contemporary methods using extensive human supervision [28]; seldom have they reported significant higher results. So why is that?

We argue that a crucial problem comes from the inadequate employment of related knowledge available from existing manually labeled datasets. In general, due to the scale issue, webly supervised learning methods usually adopt simple and straightforward object recognition algorithms. More attention is paid on employing data mining algorithms such as bootstrapping [34] and query expansion [161] to reduce semantic bias and data noise in web data. Therefore, in order to further boost the performance, a sound solution is to extract more powerful perceptual representations from existing manually labeled datasets using sophisticated object recognition algorithms, and then exploit the learned knowledge to facilitate the webly supervised learning procedure.

Following this principle, our idea is to transfer knowledge from existing manually labeled datasets *as much as possible* to guide the learning process in web scale, then optionally transfer the learned representations back to further improve the performance on original datasets. Implementing as a semi-supervised framework, a unique design of the proposed method is that the labeled dataset *utilizes stronger annotations on individual images, but with a much smaller scale.* This is somewhat a more reasonable assumption than the standard label collection process which focuses on label cleanness and data scale; compared to computer algorithms, human beings are more favorable at doing explicit labeling job, instead of repetitive jobs that aim to improve scalability.

This strategy has several advantages. First, by exploiting knowledge using more sophisticated object recognition algorithms on stronger supervision, we can largely enrich the supervision associated with web images. As a result, each web image now carries more explicit knowledge and introduces a much higher information gain. The semi-supervised method in fact acts as a bridge between webly supervised paradigms and the counterpart object recognition algorithm using manually labeled datasets.

Second, as having been discussed extensively, the main problem of using exist-

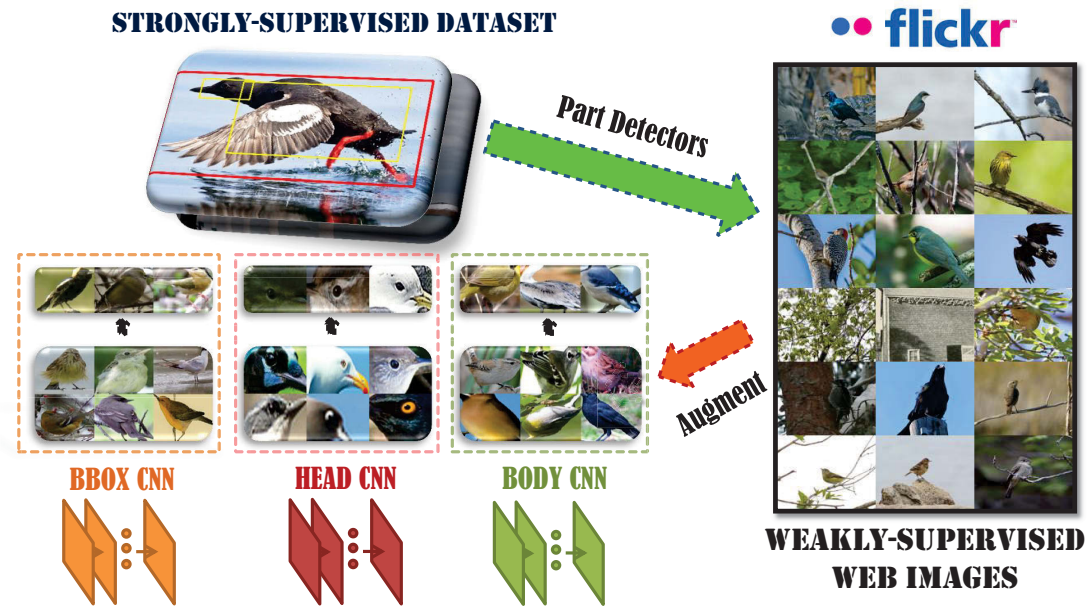


Figure 6.1: Illustration of the proposed semi-supervised method via web data. A strongly supervised dataset is introduced to "teach" web images how to learn properly.

ing datasets to initialize webly supervised learning is the data bias introduced in the construction of manually labeled datasets [110]. Here, the advantage of employing strong supervision is that the additional annotations, such as the definition of object parts or interpretable attributes, introduce a more reliable resource of transferred knowledge. These definitions are generally more closely related to common knowledge, while being inheritably shared among different categories. Therefore, as a standalone measure, they make it possible for the selected web images to be both diverse and precise at the same time.

To demonstrate the effectiveness of this strategy, we investigate fine-grained visual categorization (FGVC) [12, 74, 173], which is one of the problems that are appealing for extensive training data. With the goal of classifying objects in subordinate level, the data collection of FGVC problems is naturally harsh due to the requirement of expert knowledge and detailed part-level annotations. As a result, existing datasets [97, 150, 164] are relatively small in scale but richer in annotations, which well motivate the proposed semi-supervised strategy. In par-

ticular, as shown in Figure 6.1, we propose a warm starting scheme for performing FGVC with the help of web data, which learns robust deep convolutional feature representations and employs detailed object part annotations in a unified framework, and overcomes the lack of training data with the help of weakly supervised web images. The joint model is trained by a generalized version of the proposed MLLR paradigm which enables training on a combination of strongly and weakly supervised examples. Experimental results reveal a significant improvement over state-of-the-art methods on the FGVC benchmark CUB-200-2011 dataset [150], exemplifying the proposed strategy.

6.2 Webly Supervised Learning via Deep Domain Adaptation

The proposed method adopts a knowledge transferring strategy that utilizes domain specific knowledge extracted from existing strongly supervised datasets to guide webly supervised learning. Our key guideline here is to transfer *as much knowledge as possible* from existing strongly supervised datasets to weakly supervised web images, so that the proposed method enjoys the scalability brought by web images and meanwhile benefits from the effectiveness of sophisticated object recognition algorithms.

6.2.1 Preliminary

We start with a strongly supervised dataset $\mathcal{S}$, in which ground-truth bounding box annotations are provided not only for the entire objects p_0 but also for a set of n semantic parts $\{p_1, p_2, ..., p_n\}$. Assume that there are K fine-grained categories in the dataset.

Based on the strongly supervised dataset $\mathcal{S}$, an auxiliary dataset containing the same fine-grained categories is collected, but with only image-level labels. Images can be collected from search engines or online media sharing communities. Since the data acquirement process does not require human labeling, the weakly supervised dataset (termed $\mathcal{W}$) typically contains a larger number of images than $\mathcal{S}$. Denote the size of the datasets as $N_{\mathcal{S}}$ and $N_{\mathcal{W}}$.

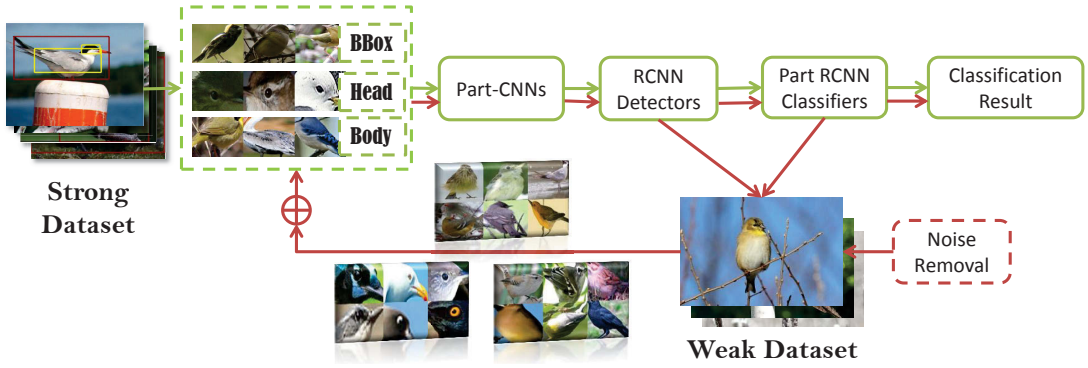


Figure 6.2: Flowchart of the proposed algorithm. Green lines show modules of strongly supervised method adopted in our framework, while red lines are additional operations of semi-supervised learning.

For each image in the dataset, selective search [141] is used to extract category-independent object proposals. Selective search typically generates 1000-2000 region proposals per image, significantly reducing the searching space for object detection. Meanwhile, compared to the method of directly using the ground-truth bounding boxes as positive samples, region proposals provide a relatively larger set of labeled image patches for training CNNs.

6.2.2 Objective function via generalized MLLR

We adopt a generalized version of the proposed MLLR to solve the webly supervised learning problem. While traditionally semi-supervised learning exploits a mixture of unlabeled data and labeled data, our method which uses weakly supervised data to augment existing strongly supervised dataset can be regarded as a generalization of the standard semi-supervised learning approaches. We consider a joint optimization algorithm that updates feature representations ϕ and model parameters w iteratively on a combination of strongly supervised dataset $\mathcal{S}$ and weakly labeled data $\mathcal{W}$. The overall objective function is defined as:

$$\min_{w,\phi} \sum_{p=0}^{n} l(w^{(p)}, \phi^{(p)}),$$

where

$$L(w^{(p)}, \phi^{(p)}) = \lambda \sum_{k} \Omega(w_k^{(p)}) + \frac{1}{N_{\mathcal{W}}} \sum_{I \in \mathcal{W}} q_I^{(p)} \cdot l(y_I, \max_{x_p \in X_I} {w_{y_I}^{(p)}}^T \phi^{(p)}(x_p)) + \frac{1}{N_{\mathcal{S}}} \sum_{I \in \mathcal{S}} l(y_I, {w_{y_I}^{(p)}}^T \phi^{(p)}(x_p)) \tag{6.1}$$

Here $y_I \in [1, ..., K]$ stands for the label of image I. For the p-th part, $\phi^{(p)}(\cdot)$ is the feature representation, x_p is the part location, and $w_k^{(p)}$ stands for classifier weights of the k-th category. For the auxiliary weak dataset, X_I is the set of candidate bounding boxes of image I. $q_I^{(p)}$ denotes an indicator of whether the detected region of p-th part in weakly supervised image I is selected to augment the training set, in order to account for label noise.

As shown in Figure 6.2, the proposed method contains an initialization step to extract robust perceptual representations from strongly supervised datasets and a model updating step by employing noisy web data via effective knowledge transfer.

6.2.3 Knowledge extraction on the strongly supervised dataset

The first step in the proposed method is to introduce domain specific knowledge via a strongly supervised algorithm. For the task of fine-grained visual categorization, we employ part-based methods (*e.g.* [12, 172, 173]) which have shown great success in the literature, and adopt deep convolutional networks (CNNs) as the feature representation [76, 127]. Whilst our method is agnostic to the specific form of part annotations and CNN architectures, here we study the same problem statement with part-based R-CNN [172], but make several significant modifications in our implementation to better deal with the lack of training data in strongly supervised datasets.

The resultant domain specific knowledge acquired from strong supervision are given in multiple forms, including deep convolutional feature representations, pre-

cise part detectors, and also robust object classifiers. They are further employed as a reliable initialization for the whole semi-supervised framework.

6.2.3.1 Feature representations

The core idea of part-based R-CNN [172] is to utilize deep convolutional features specifically trained on each object part, so that the resultant feature representations carry explicit information of distinguishing objects from the part level. To do so, in the original implementation of [172], a CNN model is trained on ground truth crops of the whole object bounding box and each of the part bounding boxes respectively. However, since the scale of the strongly supervised dataset is supposed to be relatively small, it is argued that training CNNs only on ground-truth crops will easily overfit the learning objective and thus could not achieve optimal results.

To overcome this problem, we augment the positive set by adding object proposals with high intersection-over-union (IoU) over the ground-truth bounding boxes, also known as a data jittering approach. Specifically, for each of the object parts p_i (or the whole object p_0), our goal is to train part-specific deep convolutional features $\phi^{(i)}(x)$ on the extracted region proposals.

Starting from a CNN pre-trained on ImageNet [76], we replace the CNN's ImageNet-specific 1000-way classification layer with a randomly initialized $(K+1)$-way layer that accounts for all the fine-grained categories and also a background class. Object proposals with $IoU \geq 0.5$ over the ground-truth bounding boxes are treated as positive examples for that box's class, while the others are regarded as the background. For each object proposal, the tight bounding box is dilated by m pixels (we use $m = 16$) to introduce contextual information, and all the pixels in the dilated region are warped into a fixed size of 227×227 pixels. The warped regions are then used as the input to fine-tune the network by stochastic gradient descent (SGD), starting at a learning rate of 0.001. As a result, the learned CNNs (we call them part-CNNs) carry specific domain knowledge of the fine-grained categorization, while not clobbering the initialization from large-scale ImageNet pre-training. The process described above is implemented as a fast R-CNN [56] with $(K+1)$ output categories for each part.

6.2.3.2 Object part detectors

For the case of testing with no available part-level annotations, the algorithm should be able to locate object parts automatically. Therefore, based on the fine-tuned CNNs, we further train a linear SVM with binary outputs to obtain the detector for each part. In order to achieve accurate detection results, only ground-truth boxes are used as positive samples. In our implementation, we train SVMs beyond features extracted from the *fc7* layer of CNNs and adopt a standard hard negative mining method [47] to fit the training data into memory. We also adopt bounding box regression [57] to further regularize the detected regions.

Denote $\{v_0, v_1, ..., v_n\}$ as the weights of detectors for whole-object p_0 and n parts $p_i|_{i=1}^n$. For a region proposal x, the corresponding detector scores $\{d_0, d_1, ..., d_n\}$ are computed as

$$d_i(x) = \sigma(v_i^T \phi^{(i)}(x)), \tag{6.2}$$

where $\sigma(\cdot)$ is the sigmoid function and $\phi^{(i)}(x)$ is the descriptor at location x according to the i-th part-CNN.

It is worth to be noted that the goals of Section 6.2.3.1 and Section 6.2.3.2 have a significant difference. The previous procedure aims to produce discriminative feature representations for classifying fine-grained categories; therefore, training is conducted with fine-grained output labels, with a "soft" IoU threshold that introduces more training samples to prevent overfitting. On the contrary, the latter step targets on obtaining accurate object detection results. Therefore, when training a specific part detector, only the ground-truth regions of this part are used as positive samples. Meanwhile, to introduce more robustness to factors including various object poses, object scales and occlusion, we want the part detector to be shared among all subordinate categories and thus ignore the object categorical labels in this procedure.

6.2.3.3 Fine-grained classifiers

The next step is to integrate the learned detectors and use them to train fine-grained classifiers. In part-based R-CNN, Zhang *et al.* [172] proposed three types of geometric constraint to ensure that the relative location of detected objects and their semantic parts follow a geometric prior. Here, however, the strength and

robustness of the resultant part detectors result in geometric constraints that only play a minor role in detection, especially considering that fine-grained datasets usually contain only a relatively limited number of training images. Therefore, in our implementation, we only conduct a simple box constraint to ensure object parts do not fall outside the root bounding box.

For an image I, let $X = \{x_0, x_1, ..., x_p\}$ be the predicted locations (bounding boxes) of an object and its parts, which are given during training, but unknown for both weakly supervised images and testing images. The final feature representation is then denoted as $\Phi(x) = [\phi^{(0)}(x_0), ..., \phi^{(n)}(x_n)]$, where $\phi^{(i)}(x_i)$ is the feature representation for part p_i as the output of the $fc7$ layer of the i-th part-CNN. Beyond them, a one-versus-all linear SVM is trained for each fine-grained category. The classification score for an image I being class k is then calculated as:

$$s(I; k) = \sum_{i=0}^{n} {w_k^{(i)}}^T \phi^{(i)}(x_i), \tag{6.3}$$

where $w_k^{(i)}$ is the classifier weights for class k on features extracted from the i-th object part.

As a summary, given n object parts and a root, and K fine-grained categories to be classified, the initialization step obtains:

- $n + 1$ independently fine-tuned part-CNNs with $(K + 1)$-way classification layers as the initialized feature extractors. We use the $fc7$ layer to obtain a 4096-dimensional feature vector $\phi^{(i)}$ for each part p_i.
- $n + 1$ part (or object) detectors. Each part (or root) p_i is associated with a detector d_i based on the respective CNN feature extractor $\phi^{(i)}$.
- $K(n + 1)$ sets of classification model weights, with each $w_k^{(i)} \in \mathbb{R}^{4096\times1}$.

6.2.4 Knowledge transfer to the weakly supervised dataset

The second part of the proposed method is a model updating step that transfers learned knowledge from the smaller strongly supervised dataset $\mathcal{S}$ to a larger collection of images acquired from the web (termed $\mathcal{W}$). Images in $\mathcal{W}$ are directly

collected from the Internet with no human labeling effort. It leads to two interesting facts: 1) the web dataset could contain a much larger number of images than the strongly supervised dataset, say $N_{\mathcal{W}}$ and $N_{\mathcal{S}}$ respectively, which largely scales the learning algorithm; 2) the web dataset is weakly supervised - images are associated with only image-level labels, which could be also incorrect due to label noises and outliers.

As shown in Figure 6.2, the domain transfer module (shown in red lines and arrows) involves several steps including object part detection in weak images, noise removal, re-fine-tuning CNNs, and final classifier training. We detail these steps below.

6.2.4.1 Part discovery

The first form of knowledge transfer comes from object part detectors. Since images in the web dataset are only associated with image-level labels, they are inheritably weakly supervised and carry less information. With the detectors trained on the strongly supervised dataset $\mathcal{S}$, we are able to introduce part-level "supervision" to web images by performing automatic part discovery.

The part detectors provide top-down messages to select relative patches with high discriminative power for classification. After obtaining detecting scores for all the parts, we adopt the box constraint restriction in part-based R-CNN to introduce geometric relations between the object and its parts. The detected locations $X^* = \{x_0, ..., x_n\}$ are given as:

$$X^* = \underset{X}{\operatorname{argmax}} \prod_{i=1}^{n} c_{x_0}(x_i) \prod_{i=0}^{n} d_i(x_i), \tag{6.4}$$

where

$$c_x(y) = \begin{cases} 1, \text{if region } y \text{ falls outside } x \text{ by at most 10 pixels} \\ 0, \text{otherwise} \end{cases}$$

6.2.4.2 Noise removal

As well as the lack of part-level annotations, web images are also "weakly supervised" due to label noises: it is not guaranteed that images in the auxiliary

dataset are all related to the fine-grained categories. Therefore, we introduce a noise removal process to clean up the detected part patches.

In the context of generating part patches from weakly supervised images, the strategy of selecting proper patches can be defined in two ways: (i) a sample should be selected if we are confident about the accuracy of detected localization; or (ii) a sample should be selected if it is easy to predict its true label. We argue that, in our task, adopting these strategies individually is unlikely to produce optimal results. As shown in Figure 6.3, an interesting finding here is that images which are correctly classified do not always generate valid part patches due to occlusion effects and the absence of a particular object part. Meanwhile, even when an object part is successfully detected, an image could be also incorrectly classified due to uncommon poses and imperfect object classifiers. Therefore, directly selecting samples according to classification results is error-prone. On the other hand, there is no clear boundary to perfectly separate "good" detections from "poor" detections with respect to detection scores.

We therefore propose a two-threshold strategy that combines two kinds of transferred knowledge, detection scores and classification results, to select valid part patches. The basic idea is to flexibly adjust the threshold of "good" detections by setting a loose condition on the correctly classified images and requiring harsher terms for misclassified images. Specifically, the criterion of whether a part patch x is selected to augment the training set is determined as an indicator $q_I^{(i)} = \mathcal{I}(d_i(x) > \lambda)$ where

$$\lambda = \begin{cases} \lambda_{pos}, \text{ if } \tilde{y_I} = y_I \\ \lambda_{neg}, \text{ if } \tilde{y_I} \neq y_I \end{cases}, \tag{6.5}$$

Here y_I is the label of image I and $\tilde{y_I}$ is the predicted label obtained by our object classifiers. We set two thresholds for detection scores $d_i(x)$, where $\lambda_{pos} < \lambda_{neg}$. The two thresholds are defined as:

$$\begin{aligned} \lambda_{pos} &= \sigma \bar{d}_i(neg) \\ \lambda_{neg} &= \sigma \bar{d}_i(pos), \end{aligned} \tag{6.6}$$

where $\bar{d}_i(\cdot)$ is the average detection score of part patches over correctly or in-

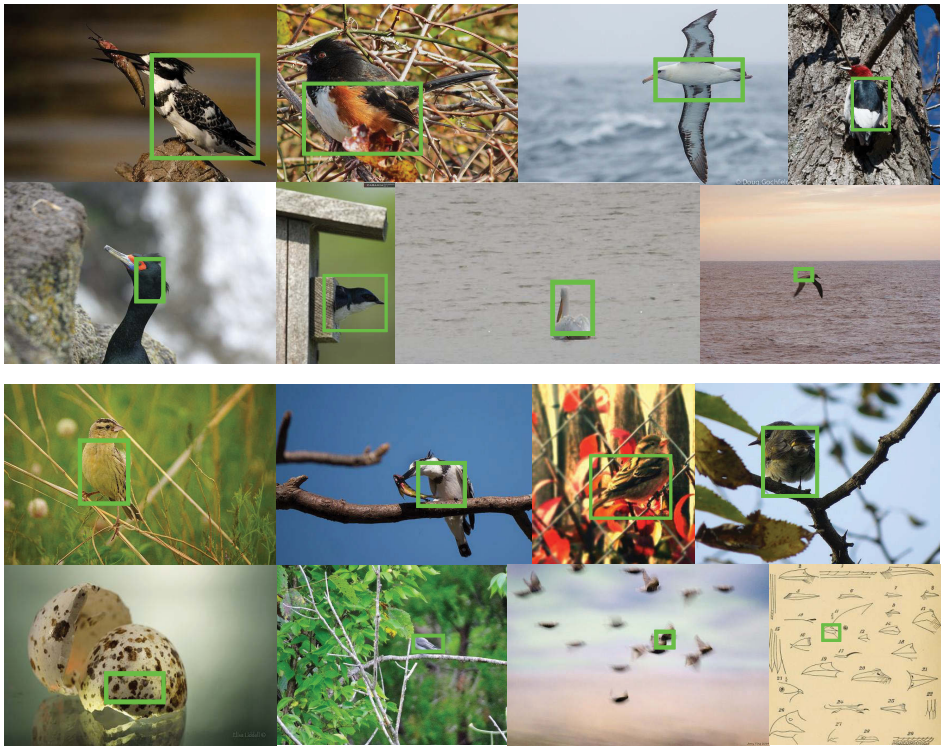


Figure 6.3: Detection results on weakly supervised images. Green frames indicate the detected bounding box for part "body". Image labels in the top two rows are correctly classified; the bottom two rows show cases in which classification has failed. Beyond the classification results, part patches in rows 1 and 3 are associated with high detection scores, while rows 2 and 4 have low detection scores.

correctly classified images, σ is a free parameter. The resultant threshold λ_{pos} is guaranteed to be lower than λ_{neg} because successfully detected part patches could always contribute to classification performance.

6.2.4.3 Re-fine-tuning CNNs

We employ part detectors trained using strong supervision and a two-threshold denoising process to generate discriminative part patches from the weakly supervised dataset. These part patches, in addition to the strongly supervised training data, are used to generate better feature representations by re-fine-tuning the

part-CNNs. We use the same CNN architecture as discussed in Section 6.2.3.2, and once again randomly initialize the $(K+1)$-way *fc8* layer with the filter weights of previous layers kept fixed. All region proposals that have ≥ 0.5 IoU over the detected part bounding boxes are cropped, dilated, warped and then fed into the CNN architecture as input. Re-fine-tuning the $n+1$ part-CNNs actually serves as an updating procedure of the feature representation ϕ in (6.1).

6.2.4.4 Final classifier

Having updated the feature representations and detected part locations on weakly supervised images, the model parameters w are jointly retrained on the strong and weak datasets to obtain the final object classifiers. Inspired by [67], we define a multi-instance learning (MIL) formulation that includes bags defined on both types of images. Specifically, for each image in the auxiliary set, the top 10 locations of the root bounding box are detected, each of which is regarded as an instance in MIL. The objective function (6.1) is rewritten as:

$$L(w) = \lambda\Omega(w) + \frac{1}{N_{\mathcal{S}}}\sum_{I\in\mathcal{S}} l(y_I, w_{y_I}^T\Phi(x)) + \frac{1}{N_{\mathcal{W}}}\sum_{I\in\mathcal{W}} l(y_I, \max_{x\in X_I} w_{y_I}^T\Psi(x)), \quad (6.7)$$

where $\Phi(x) = [\phi^{(0)}(x_0), ..., \phi^{(n)}(x_n)]$ is the part-CNN feature representation for a strongly supervised image; $w = [w^{(0)}, ..., w^{(n)}]$ denotes the joint model classifier; $\Psi(x) = [q_I^{(0)}\phi^{(0)}(x_0), ..., q_I^{(n)}\phi^{(n)}(x_n)]$ is the feature representation for a weakly supervised image, in which a part filter p is set to a zero vector if the indicator $q^{(p)}$ is zero.

As aforementioned, we employ a generalized version of the proposed MLLR to solve the objective function. Recall that for MLLR, a logistic function is used to estimate model parameters:

$$l(y_I, w_{y_I}\Phi(x)) = -\log\frac{\exp(w_{y_I}^T\Phi(x))}{\sum_{k=1}^{K}\exp[w_k\Phi(x)]}. \quad (6.8)$$

(6.7) can be regarded as a generalized version of MLLR, which simultaneously updates model parameters based on strongly supervised and weakly supervised data. MLLR adopts a Concave-Convex Procedure (CCCP) [169] to solve the non-

convex objective function. Analogously, we adopt the same paradigm to solve the objective function (6.7) by iteratively optimizing latent variable assignment for positive examples when fixing model parameters and optimizing model parameters when fixing positive latent assignments. The positive assignments is given as:

$$x_I^* = \underset{x \in X_I}{\operatorname{argmax}}[w_{y_I}^T \Psi(x)]. \tag{6.9}$$

The gradient of (6.7) is given as:

$$\frac{\partial L}{\partial w} = \lambda \frac{\partial \Omega(w)}{\partial w} + \frac{\partial l(\mathcal{S})}{\partial w} + \frac{\partial l(\mathcal{W})}{\partial w}. \tag{6.10}$$

Therefore, (6.7) can be solved following the same optimization method as Section 3.2.4, despite that the computation of gradients includes both a linear term and a multi-instance term.

Although the whole process can undergo several rounds of iteration, in practice a single updating feature representation and object classifier iteration already produces promising results. Due to the ensuing time complexity, further iterations of the whole pipeline are not performed.

At the testing stage, for a new test image, we apply the whole object and part detectors with the box geometric constraint to localize object parts; the features of all parts are then concatenated into the final feature vector for prediction. No additional annotations are required during testing.

6.3 Experiments

We present experimental results and analysis of the proposed method in this section. Specifically, we will describe the acquirement of weakly supervised web images, the effectiveness of part detectors, discuss factors on classification results, and visualize learned part-CNNs.

6.3.1 Dataset and implementation details

Experiments are conducted on two widely used fine-grained classification benchmarks: the Caltech-UCSD Birds dataset [150] (CUB200-2011) and the Oxford-

IIIT Pet Dataset [103] (PET). The CUB-200-2011 dataset contains 11,788 images of 200 types of bird, in which roughly 30 images per category are used for training. The dataset is strongly supervised, *i.e.*, images are associated with detailed annotations including image-level labels, object bounding boxes, and part landmarks. Following the protocol of [172, 173], we exploit the location annotation of two semantic parts, head and body, along with whole object bounding boxes to conduct part-based models. The PET dataset contains 37 cat and dog breeds, with roughly 200 images per category. Ground truth object and head bounding boxes are exploited as strong supervisions. We follow the provided train/test split in both datasets.

An auxiliary weakly supervised dataset is then collected to augment the strongly supervised data. Images are obtained from Flickr by conducting image searches using the names of the 200 bird species or 37 pet breeds as queries. For each category, the top 100 images for CUB and 200 images for PET are downloaded and sorted by upload time to ensure no overlap between the crawled images and test images in the datasets. No further manual filtering process is conducted on the web dataset. These downloaded images only have image-level labels, which are not always correct due to the ambiguity of query words and label noise.

We use the open-source package Caffe [68] to extract deep features and fine-tune part-CNNs from AlexNet [76]. The last fully connected layer *fc7* in the CNN architectures is used to train part detectors and in image representation for classification.

6.3.2 Detection results and analysis of discovered part patches

One of the key assumptions of our method is that the use of detectors learned from strongly supervised data can effectively detect and locate object part patches in the weakly supervised web images. Therefore, analysis commenced by evaluating detection results and studying the discovered part patches.

The quantitative detection results are measured in terms of the "Percentage of Correctly localized Parts" (PCP) metric on the test set. A part patch

Table 6.1: Part localization accuracy in terms of PCP on the CUB-200-2011 dataset.

	BBox	Head	Body
Strong DPM [8]	-	37.44%	47.08%
Part R-CNN [172]	-	61.94%	70.16%
Ours	92.84%	70.89%	75.79%

Figure 6.4: Examples of detected part patches from web images selected as valid training patches. From top to bottom: whole object, head, body. The left-most five columns show top-scoring detections, while the right two columns show patches with the lowest detection scores.

is marked as correctly localized if the predicted bounding box has $IoU \geq 0.5$ with the ground-truth bounding box. The learned part detectors produce reasonable results, achieving greater than 70% PCP for all parts (Table 6.1). The improvement over part-based R-CNN [172] is due to the additional negative mining process and from assigning the background as the $(K+1)$-th category for fine-tuning part-CNNs (as specified in [56]).

The high-performing part detectors ensure that a large number of part patches can be discovered on the web dataset. However, since the parameter-rich CNN architectures can easily overfit the training data, it is crucial to find a balance

Table 6.2: CUB-200-2011 Ablation study of different choices of fine-tuning, classifier, detector, and denoising.

Part Localization	Predict BBox		Oracle
feature\classifier	Train	Train+Weak	Train
w/o ft	68.58	71.19	74.14
ft on train	78.56	79.89	82.12
ft on train/weak	81.17	82.16	85.07
ft on train/weak-dn	83.24	84.59	86.57

between adding more training data and ensuring clean labels. Hence, we use the noise removal approach discussed in Section 6.2.4.2, with σ set to 0.5 empirically. We have experimented different σ, but resulted in similar performance. These part patches are then used to re-fine-tune part-CNNs. Example detected patches from the web dataset are shown in Figure 6.4.

6.3.3 Classification results

Since our method involves multiple steps to boost classification performance, we first analyze the effect of each step by detailed comparison with the baselines shown in Table 6.2.

Feature Perspective. The first set of comparisons reveal that improved feature representations by fine-tuning CNNs on domain-specific data significantly contribute to classification accuracy. Directly exploiting an ImageNet pre-trained CNN as the feature extractor achieves an accuracy of 68%. Fine-tuning part-CNNs on the bird training set improves this result by a large margin to 79%. Furthermore, by augmenting the part patches by performing part discovery on the weak dataset and re-fine-tuning CNNs, a further improvement to 81% classification accuracy is obtained. These results show that the larger amount of training data does indeed improve the discriminative power of the learned CNN representation. Denoising on the weak dataset further improves the accuracy by 2%.

Model Perspective. It is argued that even without using CNN features,

employing additional training data can boost classification results by increasing data diversity in training examples. We study this factor by re-training object classifiers on the augmented dataset and comparing the results to those trained on strongly supervised training data only. Results show that when the feature representations are fixed (as in traditional features such as SIFT), the performance improvement is trivial ($\sim$ 1%) compared to re-fine-tuning CNN features. This reveals an interesting phenomenon that feature representation plays a greater role in fine-grained object recognition than model training. Meanwhile, if the multi-instance learning step in weakly supervised web images is not performed, the performance will drop slightly of $\sim$ 1%. The proposed method of training classifiers on the re-fine-tuned part-CNN features finally delivers 84.6% accuracy.

Localization Accuracy. The accuracy of part localization also has a large impact on the final classification results. Although the learned detectors obtain reasonable detection accuracy for object parts, an average 3% gap still remains between classification results using predicted bounding boxes and the oracle method, which casts as an upper bound of classification performance by employing ground-truth part annotations during both training and testing. It is worth noting that our final classification result of 84.6% after introducing weakly supervised samples exceeds even the upper bound accuracy of 82.1% when using strongly supervised training data alone.

Role of Strong Initialization. Meanwhile, if the idea of part-based representations is not introduced in the model, fine-tuning on image-level labels only will obtain an accuracy of 74.2% even using additional web images, which is about 10 percent lower than our result. This proves that the knowledge learned from existing strongly supervised dataset can be effectively transferred to the weakly supervised web images and obtain significant stronger feature representations.

Comparison with state-of-the-arts. The comparison of accuracies between the proposed method and state-of-the-art methods on CUB-200-2011 is shown in Table 6.3. Unlike most of the literature on this dataset, we consider it more realistic that the birds' bounding boxes are unknown during testing. In this challenging setting, we achieve an accuracy of 84.6% using AlexNet, outperforming state-of-the-arts. It is worth to be noted that our strongly supervised method without web images already outperforms part-based R-CNN [172] by 5%, proving

Table 6.3: Accuracy comparison on the CUB-200-2011 dataset. To conduct fair comparison, we only list methods which use no annotation at testing time; for all the methods, we report their results using the same CNN architecture (AlexNet) if possible.

Method	Train Anno.	Accuracy(%)
Alignment [54]	n/a	53.6
Constellation [126]	n/a	68.5
Attention [160]	n/a	69.7
Weak-Sup [176]	n/a	75.0
Fused [175]	n/a	76.0
Bilinear CNN [92]	n/a	78.1
Without part [74]	bbox	73.7
Part R-CNN [172]	bbox+parts	73.9
PoseNorm CNN [17]	bbox+parts	75.7
Multiple-Proposal [124]	bbox+parts	78.3
Our Method	bbox+parts	78.6
(AlexNet)	bbox+parts+web	84.6
Our Method	bbox+parts	81.1
(VGG16)	bbox+parts+web	86.8

the significance of the new feature training design in our method.

Table 6.4 shows comparison results on the PET dataset. Again the proposed method obtains promising results, being comparable to [9] who used deeper network architectures. Although our method requires additional training data by collecting weakly supervised images from the web, this data acquisition process is easy to implement and requires no additional human labeling effort.

6.3.4 Visualization

Beyond the quantitative results presented above, here we present a more intuitive description of how our method works on practical examples. The procedure of classifying a fine-grained image using the proposed method is shown in Figure 6.5. Given a test image (a) belonging to *Green_Kingfisher*, detectors were used to localize the object and its semantic parts, detailed in (b) and (c). As shown

Table 6.4: Accuracy comparison on the Oxford-IIIT Pet Dataset.

Method	Accuracy(%)
Angelova *et al.* [5]	54.3
Murray *et al.* [99]	56.8
Azizpour *et al.* [9]	88.1
Simon *et al.* [126] (ALEXNET)	85.2
Simon *et al.* [126] (VGG19)	91.6
Our Method (Strong only)	86.1
Our Method (Strong+Weak)	**88.2**

in (d), the proposed strongly supervised method misclassified the image into a very similar subcategory *Pied_Kingfisher*. Closer inspections reveal that the bird in the test image indeed belongs to a rare occurring subclass in the category in which black and white spots decorate the chest. Unfortunately, the strongly supervised dataset does not include sufficient training data for this subclass.

We solved this problem by introducing an auxiliary dataset of weakly supervised images collected from the web to augment the training data. As shown in (e), the new feature representations obtained by re-fine-tuning part-CNNs on the augmented training set improved the discriminative power in this case, especially for the bird's head, even when only images in the strongly supervised dataset were employed to train the object classifiers. Naturally, inserting weakly supervised images into the training set also contributed to the classification process. Nearest neighbors shown in (f) indicated that in the web dataset, there were a larger number of images similar to the test image, making the classification result more convincing.

6.4 Summary

In this chapter, we further scale up the application of weakly supervised learning to the web scale, and present a semi-supervised strategy that transfers domain specific knowledge learned from strongly supervised data to boost the perfor-

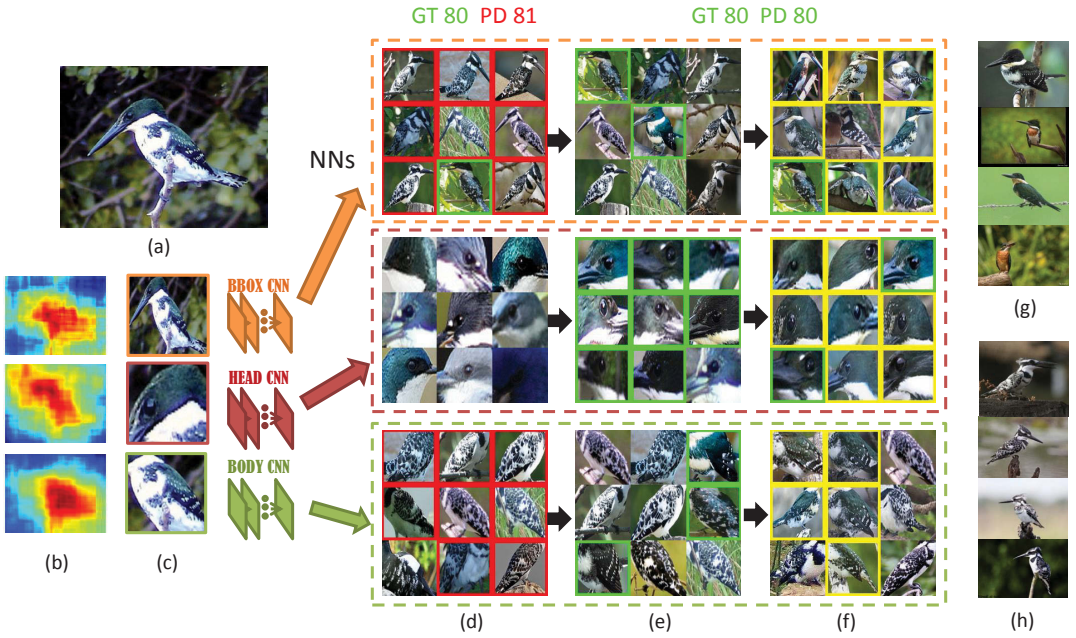


Figure 6.5: Visualization of the classification process using the proposed method with a root and two parts: head and body. (a) Test image with a ground-truth label of 80. (b) Activation map for the three detectors. (c) Located part bounding boxes. The top 9 nearest neighbours for the detected parts from the training images are shown in (d)-(f). The original strongly supervised method using training data only misclassified the test image into class 81, as shown in (d). Green boxes demonstrate the image patches of label 80, and red boxes for label 81. After re-fine-tuning part-CNNs with the augmented training set, the new feature representations guaranteed that the test image was correctly classified. (e) Nearest neighbours from the strongly supervised training set only using the new feature representations. (f) Results after putting weakly supervised images into the training set either. Yellow boxes indicate images in the weakly supervised dataset with label 80. (g) and (h) show typical training images from class 80 (*Green_Kingfisher*) and 81 (*Pied_Kingfisher*) respectively.

mance of webly supervised learning. Specifically, the proposed method is conducted as a multi-instance learning framework, in which a generalized MLLR is proposed to train strongly supervised and weakly supervised examples in a unified framework. Our method acts as a bridge between the requirement for extensive data to train deep representations and the difficulty in obtaining large-scale strongly annotated datasets. Experiments on two benchmark datasets show

that introducing additional weakly supervised images leads to an impressive improvement over baseline methods and achieves state-of-the-art results. Moreover, we believe the most important advantage of the proposed method is its potential usefulness in practice, especially considering the varied forms of part annotations in existing datasets, and the increasing complicity of CNN architectures over time.

Publications Related to This Chapter

1. **Zhe Xu**, Shaoli Huang, Ya Zhang, Dacheng Tao. Augmenting Strong Supervision Using Web Data for Fine-grained Categorization. *International Conference on Computer Vision (ICCV' 15)*, pp. 2524-2532, 2015.

2. **Zhe Xu**, Shaoli Huang, Ya Zhang, Dacheng Tao, Webly-Supervised Fine-Grained Visual Categorization via Deep Domain Adaptation, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2016 (under review).

Chapter 7

Conclusions

In this chapter, we will first conclude the entire thesis, and then elaborate possible future research directions based on this thesis.

7.1 Thesis Summarization

This dissertation studies the problem of weakly supervised learning, which enables the employment of highly discriminative feature representations when only limited forms of training supervision is provided. Modeling this problem into a latent variable framework, we first introduce a novel latent variable paradigm termed multinomial latent logistic regression (MLLR) that offers preferred characteristics including efficient latent variable inference and effective output probabilistic analysis. A set of applications, mostly in the context of object recognition, are then presented to show the effectiveness of the proposed MLLR. Meanwhile, in the experiments, several practical issues of employing latent variable models in weakly supervised scenarios are discussed extensively, including initialization, optimization, inter-class relationships, and scalability.

Specifically, theoretical analysis on MLLR is presented in Chapter 3. By introducing latent variables with a maximum a *posteriori* (MAP) inference procedure into the logistic regression framework, the proposed MLLR can be modeled as a latent variable model with structured outputs or multi-class predictions, which performs "maximization" over latent variables and "averaging" over output la-

bels. The objective function of MLLR has a semi-convexity property, which leads to effective solutions using the concave-convex procedure. We have proposed two optimization methods for solving the convex part, including a novel Newton approach using second-order derivatives. The proposed MLLR reveals superior results than existing latent variable models on multi-class classification tasks including hand-written digit recognition, human action classification, and animal object recognition in the experiments. Meanwhile, by studying the connection and difference between the proposed MLLR and other existing latent variable models both theoretically and experimentally, we present a practical manual on how to select a proper model under various application scenarios. This could be of great value for followers on this research area.

Based on the discussions above, in Chapter 4, we present a novel application of architectural style classification which is particularly suitable for the proposed MLLR. For this task, we employ the high-performing deformable part-based model to describe buildings, which introduces powerful part-based representations, while also leading to complicated latent variable definitions. Meanwhile, due to the rich inter-class relationships between multiple styles, it is preferred analyze styles using "soft" assignments instead of deterministic predictions. MLLR is particularly effective in this application for its efficient latent variable inference and the ability to produce probabilistic analysis on output predictions, leading to several interesting discoveries including an inter-class relationship map and style analysis for multiple regions of an individual building.

We further study the application of MLLR on an extremely challenging problem of weakly supervised fine-grained visual categorization (FGVC) in Chapter 5. Aiming to classify objects in the subordinate level, one critical challenge of FGVC is that the associated inter-class variance could be lower than the intra-class variance due to factors such as different object poses, scales and occlusions. We model this problem in a multi-instance learning framework and employ the proposed MLLR as the training algorithm. The unique design here is using a novel multi-task co-localization algorithm to initialize the non-convex objective function of MLLR. It is motivated by the fact that subordinate categories can be regarded as "friends" when foreground/background classification is conducted to find initialized object locations (also known as the latent variable here). The pro-

posed method achieves remarkable results on several FGVC benchmarks under the weakest supervision, where the robust initialization is shown to be crucial for the final classification performance.

Our last work (Chapter 6) aims to study weakly supervised learning in a larger scale using a generalized version of MLLR. It is achieved by employing the nearly endless supply of web data for scalability, together with a small but accurate strongly supervised set to introduce domain specific knowledge as an initialization. The proposed approach is implemented as a semi-supervised framework, for which the proposed MLLR is generalized to learn models on a joint training set of strongly and weakly supervised data. For the task of FGVC, this approach acts as a bridge between the extensive demand of data in training deep convolutional networks and the difficulty in acquiring detailed part-level annotations. As a result, the proposed method enjoys the performance improvement brought by discriminative deep feature representations and part-level classification cues in a unified framework, and thus delivers a significant improvement over state-of-the-art results which rely on the strongly supervised dataset only.

7.2 Future Work

In this thesis, we have proposed a novel latent variable model for weakly supervised learning, and studied its properties and application scenarios from both the theoretical and practical perspective. Certainly, there are still some issues that remain open and need to be further investigated.

- **Study applications of MLLR on structured outputs.** The proposed MLLR is derived from logistic regression in a multi-class formulation. However, we have also proved that MLLR can be applicable for problems with structured outputs, such as protein structure discovery and semantic segmentation. It is interesting to see the comparison between MLLR and other methods including latent structural SVM and hidden CRF in these applications.

- **Generalize MLLR to the third dimension of uncertainty.** In Sec-

tion 2.1.7, we have mentioned a generalized form of latent variable models that captures uncertainty from three dimensions, *i.e.*, latent variables h, ground-truth weak labels y, and strong predict labels s. By modeling the uncertainty in the third dimension between ground-truth labels and the predicted labels, it is possible to employ inter-class relationships in the training stage, instead of the testing stage as described in Chapter 4.

- **Multi-label problems.** MLLR can be adopted in multi-label problems, such as predicting the existence of a particular concept in contents from web-sharing medias. For example, for images in Flickr[1], users may upload several tags to describe the image content or his/her emotion for the image. Therefore, weakly supervised learning with structured outputs could be a reasonable solution for this problem.

- **Additional theoretical analysis of MLLR**. Possible extensions include multi-task objective functions, the analysis of prior/dual form solvers, distributed computation of multiple classes, *etc.*

- **Other optimization methods**. MLLR is currently solved in the framework of a convex-concave procedure using gradient methods. It would be of interest to consider using other optimizers, such as meta-heuristics that do not reply on the differentiability of the cost function, which will possibly overcome the drawbacks of the current method such as local minima.

- **Simplifying the form of MLLR**. One of the possible drawbacks of MLLR comes from the complexity of the framework. As MLLR itself is quite complicated by implementing a unified framework using data from all the training classes, it is tricky to extend the algorithm on more advanced methods such as part-based models and multi-task learning methods. Therefore, it is of interest to find an approach to simplify the algorithm and make it more flexible.

[1] www.flickr.com

References

[1] O. Aghazadeh, H. Azizpour, J. Sullivan, and S. Carlsson, "Mixture component identification and learning for visual recognition," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 115–128. 22

[2] J. Amores, "Multiple instance classification: Review, taxonomy and comparative study," *Artificial Intelligence*, vol. 201, pp. 81–105, 2013. 76

[3] G. Andrew and J. Gao, "Scalable training of l 1-regularized log-linear models," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 33–40. 21

[4] I. T. Andrews, Stuart and T. Hofmann, "Support vector machines for multiple-instance learning," in *Advances in Neural Information Processing Systems*, 2002, pp. 561–568. 15, 22, 76

[5] A. Angelova and S. Zhu, "Efficient object detection and segmentation for fine-grained recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 811–818. 118

[6] A. Angelova, S. Zhu, and Y. Lin, "Image segmentation for large-scale subcategory flower recognition," in *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*. IEEE, 2013, pp. 39–45. 77

[7] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 328–335. 83, 89

[8] H. Azizpour and I. Laptev, "Object detection using strongly-supervised deformable part models," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 836–849. 114

[9] H. Azizpour, A. Razavian, J. Sullivan, A. Maki, and S. Carlsson, "From generic to specific deep representations for visual recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 36–45. 97, 117, 118

[10] A. Behl, C. Jawahar, and M. Kumar, "Optimizing average precision using weakly supervised data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1011–1018. 48

[11] A. C. Berg, F. Grabler, and J. Malik, "Parsing images of architectural scenes," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8. 56

[12] T. Berg and P. Belhumeur, "Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 955–962. 26, 77, 101, 104

[13] T. Berg, J. Liu, S. W. Lee, M. L. Alexander, D. W. Jacobs, and P. N. Belhumeur, "Birdsnap: Large-scale fine-grained visual categorization of birds," in *Computer Vision and Pattern Recognition (CVPR), 2014*. IEEE, 2014, pp. 2019–2026. 24, 77

[14] A. Bergamo and L. Torresani, "Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach," in *Advances in Neural Information Processing Systems*, 2010, pp. 181–189. 23

[15] L. Bo, X. Ren, and D. Fox, "Kernel descriptors for visual recognition," in *Advances in Neural Information Processing Systems*, 2010, pp. 244–252. 25, 93

[16] S. Borgatti, "Netdraw software for network visualization," *Analytic Technologies*, 2002. xiv, 73

[17] S. Branson, G. Van Horn, S. Belongie, and P. Perona, "Bird species categorization using pose normalized deep convolutional nets," *arXiv preprint arXiv:1406.2952*, 2014. 26, 117

[18] S. Branson, G. Van Horn, C. Wah, P. Perona, and S. Belongie, "The ignorant led by the blind: A hybrid human–machine vision system for fine-grained categorization," *International Journal of Computer Vision*, vol. 108, no. 1-2, pp. 3–29, 2014. 26, 77

[19] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino, "Matrix completion for weakly-supervised multi-label image classification," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 37, no. 1, pp. 121–135, 2015. 23

[20] Y. Chai, V. Lempitsky, and A. Zisserman, "Bicos: A bi-level co-segmentation method for image classification," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2579–2586. 25, 97

[21] ——, "Symbiotic segmentation and part localization for fine-grained categorization," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*. IEEE Computer Society, 2013, pp. 321–328. 97

[22] Y. Chai, E. Rahtu, V. Lempitsky, L. Van Gool, and A. Zisserman, "Tricos: A tri-level class-discriminative co-segmentation method for image classification," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 794–807. 78, 90, 97

[23] K.-W. Chang, C.-J. Hsieh, and C.-J. Lin, "Coordinate descent method for large-scale l2-loss linear support vector machines," *The Journal of Machine Learning Research*, vol. 9, pp. 1369–1398, 2008. 21

[24] O. Chapelle, B. Schölkopf, A. Zien *et al.*, "Semi-supervised learning," 2006. 2

[25] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014. 83, 89

[26] D. Chen, D. Batra, and W. T. Freeman, "Group norm for learning structured svms with unstructured latent variables," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 409–416. 45, 46

[27] X. Chen, W. Pan, J. T. Kwok, and J. G. Carbonell, "Accelerated gradient method for multi-task sparse learning problem," in *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on*. IEEE, 2009, pp. 746–751. 87

[28] X. Chen and A. Gupta, "Webly supervised learning of convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1431–1439. 23, 27, 99, 100

[29] X. Chen, A. Shrivastava, and A. Gupta, "Neil: Extracting visual knowledge from web data," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1409–1416. 23, 24, 99

[30] W.-T. Chu and M.-H. Tsai, "Visual pattern discovery for architecture image classification and product image search," in *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*. ACM, 2012, p. 27. 56

[31] O. Chum and A. Zisserman, "An exemplar model for learning object classes," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8. 22

[32] R. G. Cinbis, J. Verbeek, and C. Schmid, "Multi-fold mil training for weakly supervised object localization," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 2409–2416. 22

[33] B. Collins, J. Deng, K. Li, and L. Fei-Fei, "Towards scalable dataset construction: An active learning approach," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 86–98. 24

[34] Y. Cui, F. Zhou, Y. Lin, and S. Belongie, "Fine-grained categorization and dataset bootstrapping using deep metric learning with humans in the loop," *arXiv preprint arXiv:1512.05227*, 2015. 100

[35] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893. 25

[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255. 23, 77

[37] J. Deng, J. Krause, and L. Fei-Fei, "Fine-grained crowdsourcing for fine-grained recognition," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 580–587. 26, 77

[38] T. Deselaers, B. Alexe, and V. Ferrari, "Weakly supervised localization and learning with generic knowledge," *International journal of computer vision*, vol. 100, no. 3, pp. 275–293, 2012. 76, 80

[39] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial intelligence*, vol. 89, no. 1, pp. 31–71, 1997. 22, 76

[40] S. K. Divvala, A. Farhadi, and C. Guestrin, "Learning everything about anything: Webly-supervised visual concept learning," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 3270–3277. 23, 24, 27

[41] C. Doersch, A. Gupta, and A. A. Efros, "Mid-level visual element discovery as discriminative mode seeking," in *Advances in Neural Information Processing Systems*, 2013, pp. 494–502. 23

[42] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros, "What makes paris look like paris?" *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, p. 101, 2012. 56

[43] J. Dong, W. Xia, Q. Chen, J. Feng, Z. Huang, and S. Yan, "Subcategory-aware object classification," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 827–834. 22

[44] C. Dunlop, *Architectural Styles*. Dearborn Real Estate, 2003. 56

[45] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010. 22, 48, 55

[46] R. Farrell, O. Oza, N. Zhang, V. I. Morariu, T. Darrell, and L. S. Davis, "Birdlets: Subordinate categorization using volumetric primitives and pose-normalized appearance," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 161–168. 26, 88

[47] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 9, pp. 1627–1645, 2010. xi, xiii, 4, 15, 20, 22, 33, 35, 43, 45, 48, 49, 50, 56, 60, 61, 64, 66, 77, 106

[48] H. Feng and T.-S. Chua, "A bootstrapping approach to annotating large image collection," in *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*. ACM, 2003, pp. 55–62. 24

[49] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2. IEEE, 2003, pp. II–264. 55

[50] ——, "A visual category filter for google images," in *Computer Vision-ECCV 2004*. Springer, 2004, pp. 242–256. 23

[51] ——, "Weakly supervised scale-invariant learning of models for visual recognition," *International journal of computer vision*, vol. 71, no. 3, pp. 273–303, 2007. 22, 76

[52] W. T. Freeman and J. B. Tenenbaum, "Learning bilinear models for two-factor problems in vision," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*. IEEE, 1997, pp. 554–560. 58

[53] C. Galleguillos, B. Babenko, A. Rabinovich, and S. Belongie, "Weakly supervised object localization with stable segmentations," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 193–207. 76

[54] E. Gavves, B. Fernando, C. G. Snoek, A. W. Smeulders, and T. Tuytelaars, "Fine-grained categorization by alignments," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1713–1720. 117

[55] ——, "Local alignments for fine-grained categorization," *International Journal of Computer Vision*, vol. 111, no. 2, pp. 191–212, 2014. 26, 97

[56] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1440–1448. 105, 114

[57] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587. 25, 95, 97, 106

[58] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester, "Discriminatively trained deformable part models, release 5," 2012. 21

[59] R. B. Girshick, P. F. Felzenszwalb, and D. A. Mcallester, "Object detection with grammar models," in *Advances in Neural Information Processing Systems*, 2011, pp. 442–450. 15

[60] R. B. Girshick, *From rigid templates to grammars: Object detection with structured models*. Citeseer, 2012. 21

[61] A. Goel, M. Juneja, and C. Jawahar, "Are buildings only instances?: exploration in architectural style categories," in *Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing*. ACM, 2012, p. 1. 56

[62] C. Goering, E. Rodner, A. Freytag, and J. Denzler, "Nonparametric part transfer for fine-grained recognition," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 2489–2496. 26

[63] E. Golge and P. Duygulu, "Conceptmap: Mining noisy web data for concept learning," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 439–455. 23, 24

[64] A. Gupta, A. Kembhavi, and L. S. Davis, "Observing human-object interactions: Using spatial and functional compatibility for recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 10, pp. 1775–1789, 2009. 49, 50

[65] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, "The elements of statistical learning: data mining, inference and prediction," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2005. 1

[66] M. Hoai, L. Torresani, F. De la Torre, and C. Rother, "Learning discriminative localization from weakly labeled data," *Pattern Recognition*, vol. 47, no. 3, pp. 1523–1534, 2014. 22

[67] J. Hoffman, D. Pathak, T. Darrell, and K. Saenko, "Detector discovery in the wild: Joint multiple instance and representation learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2883–2891. 77, 111

[68] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast

feature embedding," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678. 113

[69] T. Joachims, T. Finley, and C.-N. J. Yu, "Cutting-plane training of structural svms," *Machine Learning*, vol. 77, no. 1, pp. 27–59, 2009. 21, 36

[70] A. Joulin, F. Bach, and J. Ponce, "Discriminative clustering for image cosegmentation," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 1943–1950. 22, 82, 84

[71] ——, "Multi-class cosegmentation," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 542–549. 82, 85, 86

[72] A. Joulin, L. van der Maaten, A. Jabri, and N. Vasilache, "Learning visual features from large weakly supervised data," *arXiv preprint arXiv:1511.02251*, 2015. 99

[73] A. Khosla, N. Jayadevaprakash, B. Yao, and F.-f. Li, "L.: Novel dataset for fine-grained image categorization," in *First Workshop on Fine-Grained Visual Categorization, CVPR 2011*. Citeseer, 2011. 24, 77, 79, 88

[74] J. Krause, H. Jin, J. Yang, and L. Fei-Fei, "Fine-grained recognition without part annotations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5546–5555. 26, 97, 101, 117

[75] J. Krause, B. Sapp, A. Howard, H. Zhou, A. Toshev, T. Duerig, J. Philbin, and L. Fei-Fei, "The unreasonable effectiveness of noisy data for fine-grained recognition," *arXiv preprint arXiv:1511.06789*, 2015. 27

[76] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105. 25, 83, 104, 105, 113

[77] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," in *Advances in Neural Information Processing Systems*, 2010, pp. 1189–1197. 45, 46, 50

[78] N. Kumar, P. N. Belhumeur, A. Biswas, D. W. Jacobs, W. J. Kress, I. C. Lopez, and J. V. Soares, "Leafsnap: A computer vision system for automatic plant species identification," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 502–516. 24, 77

[79] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2. IEEE, 2006, pp. 2169–2178. 71

[80] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. 45

[81] Y. LeCun and C. Cortes, "Mnist handwritten digit database," *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2010. 2

[82] Y. J. Lee, A. A. Efros, and M. Hebert, "Style-aware mid-level representation for discovering visual connections in space and time," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1857–1864. 58

[83] F. Li and C. Sminchisescu, "Convex multiple-instance learning by estimating likelihood ratio," in *Advances in Neural Information Processing Systems*, 2010, pp. 1360–1368. 22

[84] L.-J. Li and L. Fei-Fei, "Optimol: automatic online picture collection via incremental model learning," *International journal of computer vision*, vol. 88, no. 2, pp. 147–168, 2010. 24

[85] L.-J. Li, H. Su, L. Fei-Fei, and E. P. Xing, "Object bank: A high-level image representation for scene classification & semantic feature sparsification," in *Advances in neural information processing systems*, 2010, pp. 1378–1386. 71

[86] Q. Li, J. Wu, and Z. Tu, "Harvesting mid-level visual concepts from large-scale internet images," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 851–858. 23

[87] Y.-F. Li, I. W. Tsang, J. T. Kwok, and Z.-H. Zhou, "Convex and scalable weakly labeled svms," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2151–2188, 2013. 22

[88] C.-J. Lin, R. C. Weng, and S. S. Keerthi, "Trust region newton method for logistic regression," *The Journal of Machine Learning Research*, vol. 9, pp. 627–650, 2008. 21

[89] D. Lin, X. Shen, C. Lu, and J. Jia, "Deep lac: Deep localization, alignment and classification for fine-grained recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1666–1674. 26

[90] H.-T. Lin, C.-J. Lin, and R. C. Weng, "A note on platts probabilistic outputs for support vector machines," *Machine learning*, vol. 68, no. 3, pp. 267–276, 2007. 65

[91] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 740–755. 23

[92] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear cnn models for fine-grained visual recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1449–1457. 25, 26, 97, 117

[93] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical programming*, vol. 45, no. 1-3, pp. 503–528, 1989. 21

[94] Q. Liu and A. Ihler, "Variational algorithms for marginal map," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 3165–3200, 2013. 20

[95] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157. 25, 93

[96] S. Maji, L. Bourdev, and J. Malik, "Action recognition from a distributed representation of pose and appearance," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 3177–3184. 48

[97] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," *arXiv preprint arXiv:1306.5151*, 2013. 24, 77, 101

[98] E. Mezuman and Y. Weiss, "Learning about canonical views from internet image collections," in *Advances in Neural Information Processing Systems*, 2012, pp. 719–727. 24

[99] N. Murray and F. Perronnin, "Generalized max pooling," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 2473–2480. 118

[100] M. H. Nguyen, L. Torresani, F. de la Torre, and C. Rother, "Weakly supervised discriminative localization and classification: a joint learning process," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1925–1932. 22

[101] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on*. IEEE, 2008, pp. 722–729. 24, 25, 77

[102] M. Pandey and S. Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based models," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1307–1314. 64, 71, 77, 80

[103] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar, "Cats and dogs," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3498–3505. 24, 77, 113

[104] S. Perkins, K. Lacker, and J. Theiler, "Grafting: Fast, incremental feature selection by gradient descent in function space," *The Journal of Machine Learning Research*, vol. 3, pp. 1333–1356, 2003. 21

[105] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Computer Vision–ECCV 2010*. Springer, 2010, pp. 143–156. 25

[106] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8. 56, 58

[107] W. Ping, Q. Liu, and A. Ihler, "Marginal structured svm with hidden variables," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 190–198. xvii, 8, 14, 16, 17, 20, 29

[108] P. O. Pinheiro and R. Collobert, "From image-level to pixel-level labeling with convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1713–1721. 22

[109] P. Pletscher, C. S. Ong, and J. M. Buhmann, "Entropy and margin maximization for structured output learning," in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2010, pp. 83–98. 16

[110] J. Ponce, T. L. Berg, M. Everingham, D. A. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. C. Russell, A. Torralba *et al.*, "Dataset issues in object recognition," in *Toward category-level object recognition*. Springer, 2006, pp. 29–48. 101

[111] J. Pu, Y.-G. Jiang, J. Wang, and X. Xue, "Which looks like which: Exploring inter-class relationships in fine-grained visual categorization," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 425–440. 88

[112] A. Quattoni and A. Torralba, "Recognizing indoor scenes," 2009. 55

[113] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell, "Hidden conditional random fields," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 10, pp. 1848–1852, 2007. 7, 8, 12, 29

[114] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*. IEEE, 2014, pp. 512–519. 25, 83, 97

[115] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3, pp. 309–314, 2004. 80

[116] O. Russakovsky, Y. Lin, K. Yu, and L. Fei-Fei, "Object-centric spatial pooling for image classification," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 1–15. 22

[117] J. Sánchez, F. Perronnin, and Z. Akata, "Fisher vectors for fine-grained visual categorization," in *FGVC Workshop in IEEE Computer Vision and Pattern Recognition (CVPR)*, 2011. 25

[118] P. Schnitzspan, S. Roth, and B. Schiele, "Automatic discovery of meaningful object parts with latent crfs," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 121–128. 13, 29

[119] B. Schölkopf, J. Platt, and T. Hofmann, "Multi-instance multi-label learning with application to scene classification." 22, 27

[120] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting image databases from the web," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 4, pp. 754–766, 2011. 23, 99

[121] A. Schwing, T. Hazan, M. Pollefeys, and R. Urtasun, "Efficient structured prediction with latent variables for general graphical models," *arXiv preprint arXiv:1206.6436*, 2012. 16, 17

[122] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013. 25

[123] J. Shen, G. Liu, J. Chen, Y. Fang, J. Xie, Y. Yu, and S. Yan, "Unified structured learning for simultaneous human pose estimation and garment attribute classification," *Image Processing, IEEE Transactions on*, vol. 23, no. 11, pp. 4786–4798, 2014. 14

[124] K. J. Shih, A. Mallya, S. Singh, and D. Hoiem, "Part localization using multi-proposal consensus for fine-grained categorization," *arXiv preprint arXiv:1507.06332*, 2015. 117

[125] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros, "Data-driven visual similarity for cross-domain image matching," in *ACM Transactions on Graphics (TOG)*, vol. 30, no. 6. ACM, 2011, p. 154. 56

[126] M. Simon and E. Rodner, "Neural activation constellations: Unsupervised part model discovery with convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1143–1151. 26, 27, 97, 117, 118

[127] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. 25, 104

[128] P. Siva, C. Russell, and T. Xiang, "In defence of negative mining for annotating weakly labelled data," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 594–608. 22

[129] P. Siva and T. Xiang, "Weakly supervised object detector learning with model drift detection," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 343–350. 22

[130] H. O. Song, R. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darrell, "On learning to localize objects with minimal supervision," in *Proceed-*

ings of the 31st International Conference on Machine Learning (ICML-14), 2014, pp. 1611–1619. 22

[131] M. Stark, J. Krause, B. Pepik, D. Meger, J. J. Little, B. Schiele, and D. Koller, "Fine-grained categorization for 3d scene understanding," *International Journal of Robotics Research*, vol. 30, no. 13, pp. 1543–1552, 2011. 77

[132] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *arXiv preprint arXiv:1409.4842*, 2014. 25, 89

[133] K. Tang, A. Joulin, L.-J. Li, and L. Fei-Fei, "Co-localization in real-world images," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 1464–1471. 22, 82

[134] B. Taskar, S. Lacoste-Julien, and M. I. Jordan, "Structured prediction, dual extragradient and bregman projections," *The Journal of Machine Learning Research*, vol. 7, pp. 1627–1653, 2006. 42

[135] M. Taylor, J. Guiver, S. Robertson, and T. Minka, "Softrank: optimizing non-smooth rank metrics," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*. ACM, 2008, pp. 77–86. 2

[136] C. H. Teo, S. Vishwanthan, A. J. Smola, and Q. V. Le, "Bundle methods for regularized risk minimization," *The Journal of Machine Learning Research*, vol. 11, pp. 311–365, 2010. 21, 36

[137] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 273–280. 71

[138] P. Tseng and S. Yun, "A coordinate gradient descent method for nonsmooth separable minimization," *Mathematical Programming*, vol. 117, no. 1-2, pp. 387–423, 2009. 38

[139] S. Tsogkas, I. Kokkinos, G. Papandreou, and A. Vedaldi, "Semantic part segmentation with deep learning," *arXiv preprint arXiv:1505.02438*, 2015. 26

[140] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on.* IEEE, 1991, pp. 586–591. 2

[141] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013. 97, 103

[142] G. Van Horn, S. Branson, R. Farrell, S. Haber, J. Barry, P. Ipeirotis, P. Perona, and S. Belongie, "Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 595–604. 26

[143] V. N. Vapnik and V. Vapnik, *Statistical learning theory.* Wiley New York, 1998, vol. 1. 2

[144] A. Vedaldi, S. Mahendran, S. Tsogkas, S. Maji, R. Girshick, J. Kannala, E. Rahtu, I. Kokkinos, M. B. Blaschko, D. Weiss *et al.*, "Understanding objects in detail with fine-grained attributes," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on.* IEEE, 2014, pp. 3622–3629. 24

[145] A. Vezhnevets and J. M. Buhmann, "Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE, 2010, pp. 3249–3256. 22

[146] A. Vezhnevets, V. Ferrari, and J. M. Buhmann, "Weakly supervised semantic segmentation with a multi-image model," in *Computer Vision (ICCV), 2011 IEEE International Conference on.* IEEE, 2011, pp. 643–650. 22

[147] S. Vicente, C. Rother, and V. Kolmogorov, "Object cosegmentation," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2217–2224. 82

[148] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba, "Hoggles: Visualizing object detection features," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1–8. 56

[149] C. Wah, S. Branson, P. Perona, and S. Belongie, "Multiclass recognition and part localization with humans in the loop," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2524–2531. 26

[150] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," 2011. 24, 77, 79, 88, 101, 102, 112

[151] C. Wang, W. Ren, K. Huang, and T. Tan, "Weakly supervised object localization with latent category learning," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 431–445. 22

[152] S. B. Wang, A. Quattoni, L.-P. Morency, D. Demirdjian, and T. Darrell, "Hidden conditional random fields for gesture recognition," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2. IEEE, 2006, pp. 1521–1527. 13, 29

[153] X.-J. Wang, L. Zhang, X. Li, and W.-Y. Ma, "Annotating images by mining image search results," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 11, pp. 1919–1932, 2008. 23

[154] Y. Wang and G. Mori, "Max-margin hidden conditional random fields for human action recognition," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 872–879. 14

[155] ——, "Hidden part models for human action recognition: Probabilistic versus max margin," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 7, pp. 1310–1323, 2011. 29, 42

[156] S. Watanabe, "Discrimination of painting style and quality: pigeons use different strategies for different tasks," *Animal cognition*, vol. 14, no. 6, pp. 797–808, 2011. 58

[157] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-ucsd birds 200," 2010. 77, 79, 88

[158] G. Wu and E. Y. Chang, "Class-boundary alignment for imbalanced dataset learning," in *ICML 2003 workshop on learning from imbalanced data sets II, Washington, DC*, 2003, pp. 49–56. 65

[159] Y. Xia, X. Cao, F. Wen, and J. Sun, "Well begun is half done: Generating high-quality seeds for automatic image dataset construction from web," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 387–400. 24

[160] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang, "The application of two-level attention models in deep convolutional neural network for fine-grained image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 842–850. 26, 77, 97, 117

[161] T. Xiao, J. Zhang, K. Yang, Y. Peng, and Z. Zhang, "Error-driven incremental learning in deep convolutional neural network for large-scale image classification," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 177–186. 24, 100

[162] J. Xu, A. G. Schwing, and R. Urtasun, "Tell me what you see and i will show you where it is," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 3190–3197. 22

[163] Y. Xu, D. Rockmore, and A. M. Kleinbaum, "Hyperlink prediction in hypernetworks using latent social features," in *Discovery Science*. Springer, 2013, pp. 324–339. 29

[164] L. Yang, P. Luo, C. C. Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proceedings of the IEEE*

Conference on Computer Vision and Pattern Recognition, 2015, pp. 3973–3981. 24, 77, 101

[165] W. Yang, Y. Wang, A. Vahdat, and G. Mori, "Kernel latent svm for visual recognition," in *Advances in Neural Information Processing Systems*, 2012, pp. 818–826. 15

[166] B. Yao, G. Bradski, and L. Fei-Fei, "A codebook-free and annotation-free approach for fine-grained image categorization," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3466–3473. 88

[167] C.-N. J. Yu and T. Joachims, "Learning structural svms with latent variables," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 1169–1176. 7, 8, 13, 14, 20, 29, 33, 34, 45

[168] G.-X. Yuan, K.-W. Chang, C.-J. Hsieh, and C.-J. Lin, "A comparison of optimization methods and software for large-scale l1-regularized linear classification," *The Journal of Machine Learning Research*, vol. 11, pp. 3183–3234, 2010. 21, 30, 36, 37, 38

[169] A. L. Yuille and A. Rangarajan, "The concave-convex procedure," *Neural computation*, vol. 15, no. 4, pp. 915–936, 2003. 19, 22, 33, 111

[170] K. Zhang, I. W. Tsang, and J. T. Kwok, "Maximum margin clustering made practical," *Neural Networks, IEEE Transactions on*, vol. 20, no. 4, pp. 583–596, 2009. 22

[171] L. Zhang, M. Song, X. Liu, L. Sun, C. Chen, and J. Bu, "Recognizing architecture styles by hierarchical sparse coding of blocklets," *Information Sciences*, vol. 254, pp. 141–154, 2014. 56

[172] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based r-cnns for fine-grained category detection," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 834–849. 26, 77, 104, 105, 106, 113, 114, 116, 117

[173] N. Zhang, R. Farrell, F. Iandola, and T. Darrell, "Deformable part descriptors for fine-grained recognition and attribute prediction," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 729–736. 101, 104, 113

[174] N. Zhang, E. Shelhamer, Y. Gao, and T. Darrell, "Fine-grained pose prediction, normalization, and recognition," *arXiv preprint arXiv:1511.07063*, 2015. 25

[175] X. Zhang, H. Xiong, W. Zhou, and Q. Tian, "Fused one-vs-all features with semantic alignments for fine-grained visual categorization." *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society*, 2015. 117

[176] Y. Zhang, X.-s. Wei, J. Wu, J. Cai, J. Lu, V.-A. Nguyen, and M. N. Do, "Weakly supervised fine-grained image categorization," *arXiv preprint arXiv:1504.04943*, 2015. 117

[177] Y. Zhang, X.-S. Wei, J. Wu, J. Cai, J. Lu, V.-A. Nguyen, and M. N. Do, "Weakly supervised fine-grained categorization with part-based image representation," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1713–1725, 2016. 22

[178] L. Zhu, Y. Chen, A. Yuille, and W. Freeman, "Latent hierarchical structural learning for object detection," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 1062–1069. 14, 29

[179] J. Zujovic, L. Gandy, S. Friedman, B. Pardo, and T. N. Pappas, "Classifying paintings by artistic genre: An analysis of features & classifiers," in *Multimedia Signal Processing, 2009. MMSP'09. IEEE International Workshop on*. IEEE, 2009, pp. 1–5. 58

Publication

Journal Paper

1. **Zhe Xu**, Zhibin Hong, Junjie Wu, Ah Chung Tsoi, Dacheng Tao. Multinomial Latent Logistic Regression for Image Understanding. *IEEE Transactions on Image Processing (TIP)*, 25(2): 973-987, 2016.

2. **Zhe Xu**, Ya Zhang, Longbing Cao. Social Image Analysis from a Non-IID Perspective. *IEEE Transactions on Multimedia (TMM)*, 16(7): 1986-1998, 2014.

3. **Zhe Xu**, Dacheng Tao, Shaoli Huang, Ya Zhang. Friend or Foe: Fine-grained Categorization with Weak Supervision. *IEEE Transactions on Image Processing (TIP)*, 2016 (accepted).

4. **Zhe Xu**, Shaoli Huang, Ya Zhang, Dacheng Tao, Webly-Supervised Fine-Grained Visual Categorization via Deep Domain Adaptation, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2016 (under review).

Conference Paper

5. Shaoli Huang*, **Zhe Xu***, Dacheng Tao, Ya Zhang. Part-Stacked CNN for Fine-grained Categorization. *Conference on Computer Vision and Pattern Recognition (CVPR '16)*, (* for equal contribution)

6. **Zhe Xu**, Shaoli Huang, Ya Zhang, Dacheng Tao. Augmenting Strong Supervision Using Web Data for Fine-grained Categorization. *International Con-*

ference on Computer Vision (ICCV' 15), pp. 2524-2532, 2015.

7. **Zhe Xu**, Dacheng Tao, Ya Zhang, Junjie Wu, Ah Chung Tsoi. Architectural Style Classification using Multinomial Latent Logistic Regression. *European Conference on Computer Vision (ECCV' 14)*, pp. 600-615, 2014.