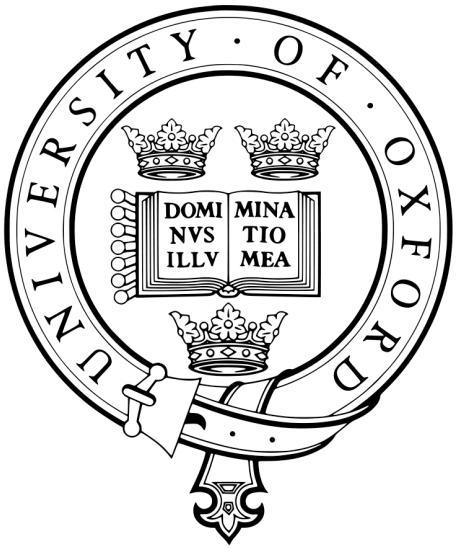
Transfer Learning for Object Category Detection

D.Phil Thesis

Robotics Research Group
Department of Engineering Science
University of Oxford



Supervisor:
Professor Andrew Zisserman

Yusuf Aytar
Brasenose College
2014

Yusuf Aytar
Brasenose College

Doctor of Philosophy
January 2014

Transfer Learning for Object Category Detection

Abstract

Object category detection, the task of determining if one or more instances of a category are present in an image with their corresponding locations, is one of the fundamental problems of computer vision. The task is very challenging because of the large variations in imaged object appearance, particularly due to the changes in viewpoint, illumination and intra-class variance. Although successful solutions exist for learning object category detectors, they require massive amounts of training data.

Transfer learning builds upon previously acquired knowledge and thus reduces training requirements. The objective of this work is to develop and apply novel transfer learning techniques specific to the object category detection problem.

This thesis proposes methods which not only address the challenges of performing transfer learning for object category detection such as finding relevant sources for transfer, handling aspect ratio mismatches and considering the geometric relations between the features; but also enable large scale object category detection by quickly learning from considerably fewer training samples and immediate evaluation of models on web scale data with the help of part-based indexing. Several novel transfer models are introduced such as: (a) *rigid transfer* for transferring knowledge between similar classes, (b) *deformable transfer* which tolerates small structural changes by deforming the source detector while performing the transfer, and (c) *part level transfer* particularly for the cases where full template transfer is not possible due to aspect ratio mismatches or not having adequately similar sources. Building upon the idea of using part-level transfer, instead of performing an exhaustive sliding window search, part-based indexing is proposed for efficient evaluation of templates enabling us to obtain immediate detection results in large scale image collections. Furthermore, easier and more robust optimization methods are developed with the help of feature maps defined between proposed transfer learning formulations and the "classical" SVM formulation.

This thesis is submitted to the Department of Engineering Science, University of Oxford, in fulfillment of the requirements for the degree of Doctor of Philosophy. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Yusuf Aytar, Brasenose College

Acknowledgements

I am very grateful to my supervisor, Professor Andrew Zisserman, for his guidance, support, and advice throughout my PhD. I would like to thank him for fruitful discussions on research directions, comforting positive views in the time of failures, motivating appreciations for achievements, and well-directed guidance with his extensive expertise in computer vision and machine learning. I found great inspirations in his character, full of passion, devotion, enthusiasm and optimism. It has been a great pleasure working with him for the past five years.

I want to thank my labmates from VGG for all the shared experiences and joyful VGG dinners. Special thanks go to the VGG-twins, Amr, Relja and Karen. Particularly, I would like to thank Relja for giving me relief by doing administrative things before I do. I want to thank Arpit, Varun, Patrick, Maria-Elena, James and Lyndsey for kindly welcoming us to the VGG lab; and Mircea, Chai, Max, Carlos, Ken, Omkar, Tomas, Meelis, Elliot, Victor, Lubor, Eric, Marcin, Alonzo, Minh, Matthew, Esa, Nataraj, Siddhartha, Mayank and Karel for making the PhD a joyful experience. I wish them all the best in life and hope to keep in touch.

Most importantly, I would like to thank my lovely fiancée Zeynep for making my rainy Oxford days feel sunny, my brother Cem and his wife Asiye, and my parents Sabriye and Celal to whom I dedicate this thesis.

Last but not the least, I would like to thank Dr. Aybars Uğur, Dr. Muhammed Cinsdikici and Professor Mubarak Shah not only because of their excellent supervision in the years before my PhD, but also for the great motivations, inspirations and experiences that they shared with me.

Contents

Chapter 1

Introduction

1.1 Objective, Motivation and Challenges

Tabula rasa, meaning blank slate in Latin, refers to the concept that learning starts without any prior knowledge, i.e. a blank slate to be filled in by time. However, learning doesn't necessarily start from scratch and it can be performed by building upon previously existing knowledge, i.e. *transfer of learning* (Haskell, 2001). As humans, we have an amazing capacity of learning progressively as we grow, and it is achieved by transferring the already acquired knowledge in order to solve novel tasks in new conditions we confront in life. Similar learning mechanisms apply to visual cognition as well. By the time we are 6 years old, we recognize tens of thousand of categories of objects (Biederman, 1987), actions, materials, etc. and we progressively keep learning throughout our life. Finding a way to replicate these abilities in machines will profoundly improve visual recognition. In machine learning,

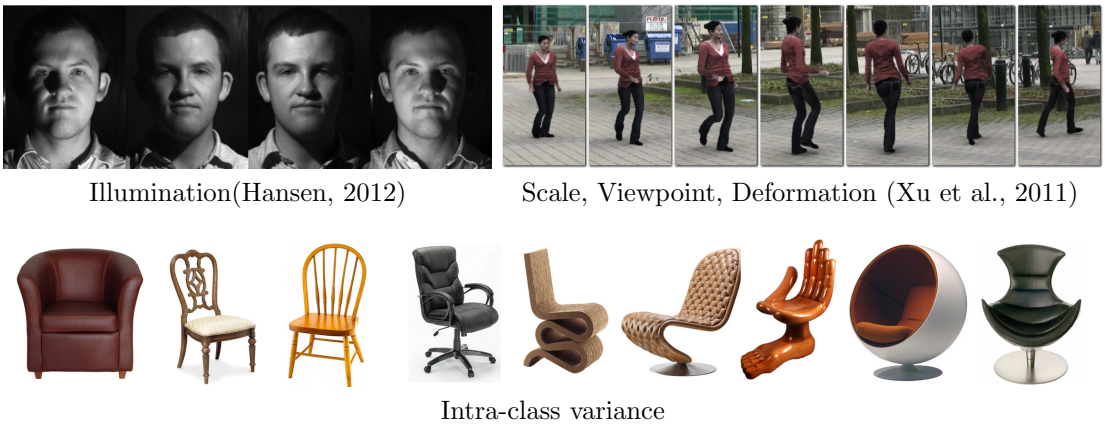


Figure 1.1: **Challenges of Object Category Detection.** A successful object category detector should be invariant to changes in illumination, occlusion, background clutter, scale, viewpoint, deformation and intra-class variance.

utilizing the previously learnt knowledge for solving novel tasks is known as *transfer learning* or *knowledge transfer*. Considering the fact that many visual categories resemble each other visually (such as animals, cars of different brands, birds, etc.), transfer learning can be successfully applied to object category recognition.

Object category *detection*, the task of determining if one or more instances of a category are present in an image and, if they are, localizing them (Everingham et al., 2010a), is one of the fundamental challenges in computer vision research. The main challenge is to be able to tolerate the variations in the appearance of the object of interest. A robust object detection system should be invariant to changes in illumination, occlusion, background clutter, scale, viewpoint, deformation and intra-class variance (see figure 1.1). Although all these challenges are hard to address, intra-class variance poses a more challenging problem as it concerns many other variations in style, material, texture, colour, etc. and it can gradually change together with the common sense perception of the category. For instance, the innovative designs of chairs redefine the boundaries of the common sense perception of a chair. Therefore,

moving from instances to categories, i.e. *generalization*, is a very challenging task to accomplish, yet as humans we are quite talented in identifying similarities and learning the categories even with very few samples. The goal of transfer learning is develop similar methodologies for computational learning. In this thesis, we investigate novel transfer learning models which better address the challenges in object category detection.

Object category detection has a wide spectrum of applications. Even though there exist many robust detection applications for certain category of objects (i.e. faces and pedestrians), the detection quality threshold decreases rapidly when we are exposed to a wide spectrum of categories (e.g. planes, sofas, chairs, dogs, lions, etc.) that exist in the world. Yet, even in this current state, we have many influential applications that found their places in our daily lives. For instance, face detection technology is actively used in digital cameras for photo enhancement (see figure 1.2). Pedestrian, car, bicycle, motorbike, bus and traffic sign detection systems are about to be released in the new generation of cars. For instance, Volvo already introduced the S60 model which actively uses object detection systems in order to warn the driver if a collision is imminent and even makes the decision for activating the vehicle's full braking power to avoid an accident (see figure 1.3). Following the same trend, Google's, and most recently Tesla Motors', ambitious aim for fully automatic cars have already gone through substantial tests and they will be a part of our lives perhaps sooner than we expect. The need for accurate detection of a large variety of object categories will be more indispensable with the release of automatic cars and many other automated robots (i.e. unmanned air vehicles (UAVs), household robots, robot guides, etc.) that are destined to be a part of our daily lives. Another fertile area for object category detection is understanding the visual content of images and

Figure 1.2: The Canon PowerShot G9 implements the face detection technology for better auto-enhancement for faces in photos. Photo courtesy Canon, USA.

Figure 1.3: The Volvo S60 uses pedestrian and car detection systems for warning the driver and even makes full power auto break decision in order to avoid an accident. Photo courtesy Volvo, Sweden.

videos on the web. Most of the searchable visual content on the web is retrieved using tags and text annotation which are poor and inadequate, furthermore the vast amount of unannotated visual content on the web is mostly unreachable. The auto-annotation of these images and videos through object detection will increase the quality and coverage of our reach to these contents. Similar technology also applies to better managing the internal visual archives of companies and broadcast media.

The traditional method for object category detection has two main phases: training and testing. For the training phase, thousands of annotated image examples of the

object category need to be annotated in order to compose the positive and negative training sets. Then visual features (e.g. colour, edge histograms, bag of visual words) are extracted from the training images and subsequently fed into a machine learning system (e.g. SVMs) which learns a model for the object category. In the testing phase, the classifier model is evaluated on images or videos over all positions, and scales. Even though there are successful methods for developing classifiers for specific object categories (Dalal and Triggs, 2005, Felzenszwalb et al., 2008, Viola and Jones, 2001), using these approaches for tens of thousands of categories is implausible due to the need for intensive computation and massive amounts of annotation work. For each object category, training a model from scratch requires annotating lots of images which is costly and time consuming, preventing the goal of recognizing tens of thousands of categories. Furthermore, using the learnt models in a brute force method for detection is unacceptable considering the massive amount of images and videos produced per second in the world. This thesis proposes novel transfer learning solutions for training object category detectors which reduces the annotation required hence enabling scaling up of the object category detection to tens of thousands of categories. Furthermore, immediate object category detection is introduced which potentially enables running category detectors on web scale image collections in less than a second.

1.2 Outline

This section will draw the outline of the thesis by briefly describing the chapters.

In **chapter 2**, transfer learning will be introduced from the perspectives of human

psychology, machine learning and computer vision. The related work on transfer research in machine learning and computer vision, particularly in object classification and detection, will be elaborated upon and necessary grounding for the other chapters will be established.

In **chapter 3**, transfer training of discriminatively trained object category detectors will be discussed. Given the source and the target task, this chapter particularly investigates "how" to perform transfer for object detection. To this end, we propose and compare three transfer learning formulations where a template learnt previously for another similar category is used to regularize the training of a new category. All the formulations result in convex optimization problems. Experiments demonstrate significant performance gains by transfer learning from one class to another (e.g. motorbike to bicycle), including one-shot learning, specialization from class to a subordinate class (e.g. from quadruped to horse) and transfer using multiple components. In the case of multiple training samples it is shown that a detection performance approaching that of the state of the art can be achieved with substantially fewer training samples.

In **chapter 4**, we stretch the limits of generalization and investigate how to generalize from a single sample. Particularly we focus on improving Exemplar SVMs (E-SVMs, (Malisiewicz et al., 2011)), a SVM trained with only a single positive sample. We introduce a method of part based transfer regularization that boosts the performance of E-SVMs, with a negligible additional cost. This Enhanced E-SVM (EE-SVM) improves the generalization ability of E-SVMs by softly forcing it to be constructed from existing classifier parts cropped from previously learnt classifiers. In content based image retrieval applications, where the aim is to retrieve instances

of the same object class in a similar pose, the EE-SVM is able to tolerate increased levels of intra-class variation and deformation over E-SVM, and thereby increases recall.

In **chapter 5**, in addition to introducing a convex objective for incorporation of pairwise co-occurrence statistics into the SVM framework, we relate transfer learning, optimization and feature maps by casting introduced transfer learning formulations to a "classical" SVM formulation, and subsequently providing easier and more robust optimization for transfer formulations through the use of expertise developed throughout the years for solving SVM optimization problems. Constructed feature maps also have certain implications which relates to two groups of active research namely transfer regularization and feature augmentation approaches.

In **chapter 6**, the objective is object category detection in large scale image datasets, where the object category is specified by a sliding window HOG classifier, and retrieval should be immediate at run time in the manner of Video Google (Sivic and Zisserman, 2003). To this end, we introduce new image and classifier representations through the use of a vocabulary of discriminative classifier patches which enables us to obtain immediate ($\sim$ 1 second) detection results on large image collections with the help of inverted file indexing. We also further improve the results with a fast method for spatial reranking of images using the detections which does not require accessing their HOG descriptors. We evaluate the fast detection method on the standard PASCAL VOC dataset, together with a 45K image subset of ImageNet, and demonstrate near state of the art detection performance in low ranks whilst maintaining immediate retrieval speeds. Applications are also demonstrated using an exemplar SVM for pose matched retrieval.

Finally, in **chapter 7**, we conclude with summarizing the proposed methods and providing potentially promising future directions including the suggestions for answering the "where to transfer from?" question, and transferring knowledge in the instance level as opposed to the category level.

1.3 Publications

The contributions discussed in section 1.2 are published in the papers listed below:

- Tabula Rasa: Model Transfer for Object Category Detection, International Conference on Computer Vision (ICCV), 2011
- Enhancing Exemplar SVMs using Part Level Transfer Regularization, British Machine Vision Conference (BMVC), 2012

The chapter 6 contains unpublished work which will be submitted to Computer Vision and Pattern Recognition (CVPR), 2014.

Chapter 2

Literature Review

Transfer of learning, learning by building upon previous experiences and knowledge, is studied in a variety of fields (Helfenstein, 2005) including Educational Psychology, Cognitive Psychology, Linguistics, Human Computer Interaction, and Machine Learning. Even though across fields there is no unique categorization and conceptualization of transfer research, there is a general consensus about the basic definition. It can be briefly defined as the ability to apply knowledge and skills learnt in previous tasks to novel tasks. This section will review the transfer learning literature from several aspects starting with human psychology and machine learning, and then discuss the concept particularly for object category detection.

2.1 Transfer of Learning in Psychology

Transfer of learning, how individuals would transfer knowledge learnt in one context to another context that shares similar characteristics, is one of the main research areas of educational and cognitive psychology. In the history of psychology the concept was originally introduced as *transfer of practice* by Thorndike and Woodworth (1901). In the International Encyclopedia of Education, Perkins and Salomon (1992) defines the notion as below:

> Transfer of learning occurs when learning in one context enhances (positive transfer) or undermines (negative transfer) a related performance in another context. Transfer includes near transfer (to closely related contexts and performances) and far transfer (to rather different contexts and performances).

The realization of transfer relies on the judgement of *similarity* which indeed has a very subjective definition depending on the qualities of the learner. Even though some part of our similarity judgements evolve throughout life, the basic mechanisms that drive our similarity judgements are defined by our nature and hard-wired into our brains. For instance, if a rat was taught to respond to the triangle T_1 shown in figure 2.1, and then later on shown a "different" triangle T_2 drawn with dotted lines it will not respond. However, a chimpanzee or a small child will respond to both of these triangles as being the same object (Haskell, 2001), and they will transfer their experiences. The notion of abstraction and generalization have very strong relations with the concept of similarity. Since transfer is mainly accomplished through connections created by the similarity judgements, identifying

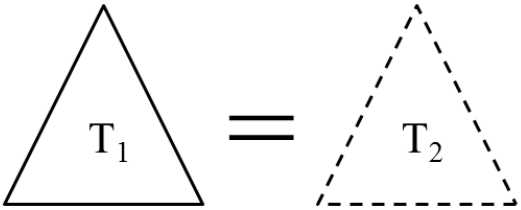


Figure 2.1: **The similarity judgements.** The two triangles above can be perceived as the same concept by a small child or a chimpanzee, but not by a rat. (Haskell, 2001)

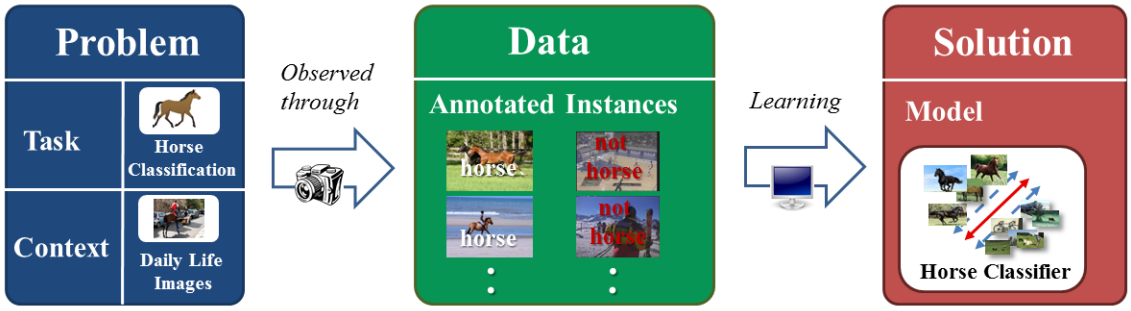


Figure 2.2: *Problem* consists of a *task* and a *context* and can only be observed through *data*. As a result of a learning process a *solution* is learnt from data.

these similarity connections is a vital part of the transfer process.

2.2 Transfer Learning in Machine Learning

In the machine learning perspective, *problems* are abstract concepts and they are observed through the *data* which consists of *instances* and associated *labels* to learn from, and the *solutions* are mainly the computational models (represented with parameters) that are learnt for solving the problem (see figure 2.2). An example problem from computer vision would be the classification of horses (i.e task) in the

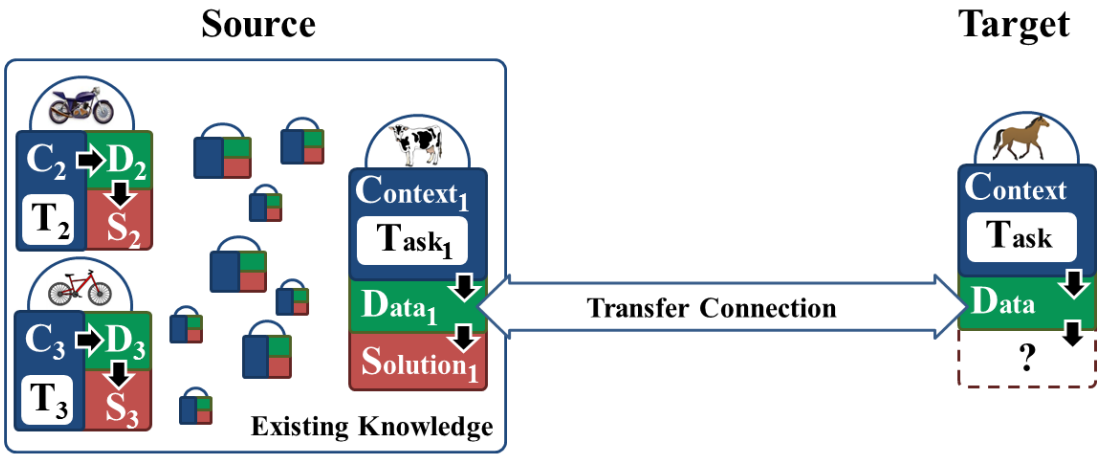


Figure 2.3: The key elements of the transfer process: (a) source problems with solutions, (b) a target problem, and (c) established transfer connections.

daily life scenes (i.e context). Here the data consists of image instances together with labels corresponding to the existence or absence of horses in the images. Instances can be named as positive or negative instances based on the labels, and they are mainly used in the form of extracted *features*. The *solution* is a computational model which can distinguish the positive instances from the negative ones, in other words, a model can assign a label to a test instance.

The transfer process can be briefly summarized with four main questions: (a) from where to transfer, (b) what to transfer, (d) how to transfer, and (c) when to transfer.

The transfer process starts with (i) a *target task* to be learnt in a *target context*, (ii) a set of *solutions* to the *source tasks* already learnt in the *source contexts*, and (iii) the *transfer (similarity) connections* which are decided based on the similarity or resemblance between the target and the source problems (see figure 2.3). Identification of these assignments answers the "where" question.

Even though there are many studies addressing diverse issues in transfer learning,

the "where" question is mostly left unanswered. This question mainly refers to the similarity notion which can be very subjective depending on the point of view as it is mentioned in the psychological perspective. Mostly the *similarity connections* have to be established between the target data and the source data or solutions (models). For instance, in order to decide if a motorbike detection problem is a suitable source for a bicycle detection problem, we can evaluate bicycle instances using the motorbike model together with many other models and decide to transfer from models which scored above certain threshold. If the target data is not observed but some meta-data for the target problem exists, then similarity connection can also be constructed using the meta-data of the source problems and the target problem. This setting is also known as "zero shot learning" (Aytar et al., 2008, Lampert, 2009, Rohrbach et al., 2010) and the general forms of the meta-data are class labels and associated definitions or attributes which can be obtained by both supervised (Lampert, 2009) and unsupervised methods (Aytar et al., 2008, Rohrbach et al., 2010).

Once the source problems, target problem and transfer connections are decided, the next step is to decide what to transfer, how, and when. The "what" term generally defines the type of information transferred from the source to the target which can be a solution (model parameters) or data (instances). "How" defines the nature of the transfer, if the transferred knowledge will be transformed or transferred as it is, and how it will be used while learning the new task. The question "when" asks in which situations transfer should be performed. It can be seen as a higher level decision on the amount of transfer from the selected sources as well. This question is highly related to the concept of *negative transfer*. If source tasks are not too similar and/or there is already adequate amount of data for learning the target

task successfully the transfer may hurt instead of helping the learning process.

2.2.1 Types of Transfer: Domain vs. Task

There are two main aspects that may change between the source problem and the target problem: the task or the domain (context).

Task Adaptation (inductive transfer of Pan and Yang (2010)). In this setting source and target tasks are different and there is no assumption about the domains. For instance learning a horse detector by transferring from a cow detector is an example of task adaptation.

Domain Adaptation (transductive transfer of Pan and Yang (2010)). In this setting the source and the target domains are different, but the tasks are the same or very similar. The labels in the target domain may or may not exist. For instance a generic face detectors which learns for general skin colour distributions might need to adapt itself for a photograph taken in a darker setting. In a similar setting Jain et al. (2011) adapted face detectors to a specific photograph for better handling the new domain conditions. A generic pedestrian detector can also be adapted to a specific traffic scene. Pedestrian detector can be evaluated on the new traffic scene, and a new detector can be trained by incorporating the domain specific pedestrian examples (Wang and Wang, 2011).

Although the domain and the task is fairly distinguishable while defining the problem in the conceptual level, it becomes a very challenging job to separate domain related and task related properties in the observed *data* and the *solution*. Therefore there is a considerable amount of similarity in the proposed solutions for both of the

problems. For example, the same model based transfer method, Adaptive Support Vector Machines (Li, 2007, Yang et al., 2007b), is used for both task adaptation (Tommasi and Caputo, 2009) and domain adaptation (Yang et al., 2007a).

Pan and Yang (2010) defines a third setting, *unsupervised transfer learning*, where no labels are available and the target problem is an unsupervised problem such as clustering, dimensionality reduction or density estimation. For instance Dai et al. (2008) studied self-taught clustering, an unsupervised transfer learning approach, which aims at clustering a small set of unlabeled data in the target domain with the help of a large amount of unlabeled data in the source domain by learning a common feature space across domains.

2.2.2 Materials and Methods of Transfer: What and How?

This section addresses the "what" and "how" questions of the transfer process mentioned in section 2.2. The "what to transfer" question is concerned with the part of the knowledge from the source that will be harnessed while solving the target problem. Transferred knowledge can be data (instances) or solutions (the parameters of the learnt models). Once the material that will be used for transfer is decided, the next step is to develop the learning methods that take the transferred knowledge into account while learning the target problem, which answers the "how" question. Here the methods will be categorized by the type of transferred knowledge and within each category the methods of transfer will be discussed.

Instance Transfer. This type of transfer focuses on the transfer of source instances that can be reused while solving the target problem. Although the source

data may not be used directly as it is, certain parts of it with appropriate re-weightings can boost the target learning process. The main difference between this transfer and multi-task learning (MTL) is that although instance transfer uses both source and target instances it is only learning the target solution whereas MTL is concerned with solving both problems together. A variety of methods for instance transfer are introduced in both machine learning and computer vision fields. Dai et al. (2007) proposed a transfer extension to the *Adaboost* algorithm. The underlying motivation is that due to the differences in the data distributions of source and target problems some of the source instances can be helpful but others may not help, or even hurt the performance. Thus it suggests an iterative method to re-weight the used source instances during learning. Jiang and Zhai (2007) proposed a heuristic method for minimizing the difference between the conditional probabilities of source and target domains by removing misleading training instances. Wu and Dietterich (2004) incorporated the source (auxiliary) instances in a Support Vector Machine (SVM) framework while learning the target task. Daumé III (2009) utilized instances from both source and target problems by introducing a feature replication method which is the used for learning a kernel function in a discriminative learning framework. Duan et al. (2009) proposed the Domain Transfer SVM (DTSVM) approach for video concept classification task. DTSVM minimizes the structural risk function of SVM and Maximum Mean Discrepancy, a criterion that identifies the difference between the distributions of source and target instances, by learning an optimized kernel function. Wang et al. (2010) introduced a maximum-margin visual object classification framework that takes advantage of the assumption that a good object model should respond more strongly to instances from similar source categories than to instances from dissimilar source categories. Lim et al. (2011)

illustrated a performance improvement by borrowing and re-weighting samples from different visual object categories while learning a target object detector.

Model (Parameter) Transfer. This type of transfer concentrates on transferring the model parameters which are the solutions of the source problems. In the generative learning perspective, Fei-Fei et al. (2006) introduced a transfer learning approach in a Bayesian framework that learns common priors over visual classifier parameters, and learns the target model by updating the prior model parameters in the light of one or more examples from the target category. Marx et al. (2005) performed Bayesian transfer methods while learning logistic regression classifiers for the target problem. Stark et al. (2009) proposed a probabilistic shape based model that allows knowledge transfer on three different levels: shape and appearance of the parts, local symmetry between parts, and part topology. Their model performs partial or full knowledge transfer explicitly by transferring the model parameters (see figure 2.4). In the discriminative learning perspective, Zweig and Weinshall (2007) investigated a transfer learning approach by combining object classifiers from different hierarchical levels into a single classifier using constellation based object category models. Their main intuition being that the classifiers from different levels capture different aspects of the object. Yang et al. (2007b) introduced Adaptive Support Vector Machines (A-SVM) for adapting classifiers to the new domains. A-SVM and its variants for transfer learning in maximum-margin frameworks, particularly SVMs, will be comprehensively articulated in section 2.2.3. Starting with an assumption that good detectors share certain statistical properties, Gao et al. (2012) learnt low-level statistical priors over the source model parameters, and proposed to enforce these statistics while learning a model for the target problem.

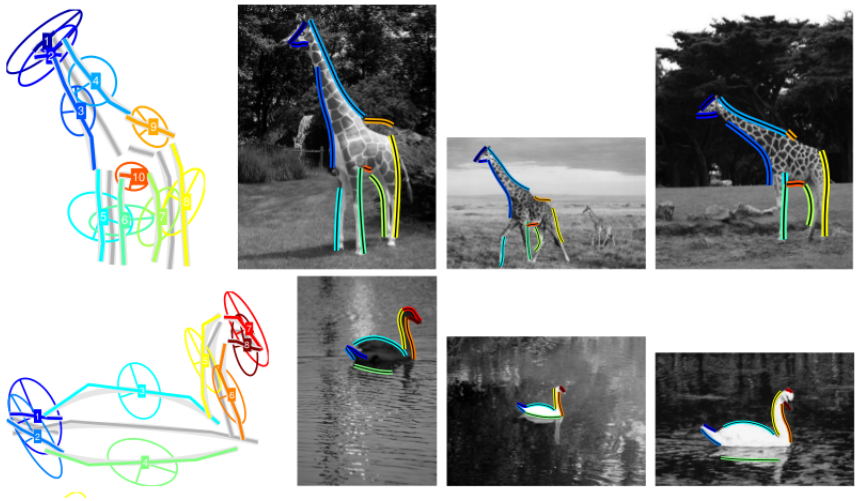


Figure 2.4: **Explicit part based transfer.** Each part visualized with means and variances. Stark et al. (2009) performs partial or full knowledge transfer explicitly by transferring the model parameters.

Feature-representation Transfer. This type of transfer can be located somewhere in between instance transfer and model transfer. The aim here is to learn a better feature representation that would ease the learning procedure of the target problem, in other words, establishing better building blocks with the help of the source problem to solve the target. Particularly, when the target problem has very few instances and the generalization is hard, this type of transfer better guides the learner by restricting its search space to a semantically meaningful one. It can also be considered as taking one step from low-level features to mid-level features, which are semantically more meaningful. For instance, Fink (2004) learnt a kernel based distance metric that emphasizes relevant feature dimensions of related source problems, then using a single sample of the target problem, he constructed a nearest neighbour classifier in this metric space. Quattoni et al. (2008) introduced a kernel based method for learning a sparse prototype representation from unlabelled

data and related tasks. Kulis et al. (2011), Saenko et al. (2010) examine the effect of a domain shift and introduce a method that adapts object models acquired in a particular visual domain to new imaging conditions by learning a transformation that minimizes the domain-induced changes in the feature distribution. Even though not named as transfer learning, many recent studies show that using mid-level (i.e. part or attribute) and high-level (i.e. responses of high-level classifiers) feature representations can improve the performance. Learning over mid-level and high-level feature representations (as opposed to using low-level features) can be seen as a generic *feature-representation transfer*, because they restrict the search space to a more semantically meaningful one with the help of previously learned classifiers (e.g. part, attribute or category classifiers). One popular branch is representing the image by the responses of a set of attribute classifiers which are learnt in a supervised fashion. This attribute-based representation is successfully employed for object classification tasks (Farhadi et al., 2009, Kumar et al., 2009, Lampert and Blaschko, 2009, Yao et al., 2011). In a similar but unsupervised fashion, Torresani et al. (2010) introduced the "classeme" descriptor (Rastegari et al., 2011, Torresani et al., 2010) which is composed of boolean outputs of a set of non-linear object classifiers that are learnt from images returned by text-based image search engines. Building upon the attribute-based representation, Douze et al. (2011) incorporated Fisher vectors to the representation and proposed an efficient coding technique for compressing the descriptor. Another branch is populating the features with the responses of high-level classifiers (Li et al., 2010a,b, Song et al., 2011, Yao et al., 2011). Li et al. (2010b) proposed to represent the image as a response map of a large number of pre-trained generic object detectors, named *Object Bank*. Similar to object bank representation, Yao et al. (2011) used stacked responses of objects and

human pose classifiers as a high-level feature for an action classification task. Song et al. (2011) proposed to improve object classification and detection in an iterative fashion by harnessing the outputs of one task as high-level features for learning the other task.

2.2.3 Model Based Transfer Regularization in SVMs

This section will discuss transfer regularization formulations which regularize the newly learnt models with their distance to the existing source category models. Initially we will start with the SVM formulation and gradually introduce A-SVM and its variants.

Support vector machines are very powerful and popular tools for solving binary classification problems. The main idea is to find the optimum hyperplane that separates the positive samples from the negative ones by maximizing the margin in the feature space. In the non-separable case, slack variables are introduced for accommodating outliers. Let $X = \{x_i\}_{i=1}^{N}$ be the training samples in the feature space $x_i \in \mathcal{X}$, $Y = \{y_i\}_{i=1}^{N}$ be the corresponding binary class labels such that $y_i \in \{+1, -1\}$, and N the number of training samples. Then the linear SVM objective is as follows:

$$min_{w,b} \quad ||w||^2 + C \sum_{i}^{N} \max\left(0, 1 - y_i(w^{\mathsf{T}} x_i + b)\right) \tag{2.1}$$

where w is the normal to the separating hyperplane and the scoring function which predicts the label of the test sample z_i is $f(z_i) = sign(w^{\mathsf{T}} z_i)$. The $||w||^2$ term encourages margin maximization and the other term, hinge loss $\left(\sum_i^N \max\left(0, 1 - y_i(w^{\mathsf{T}} x_i + b)\right)\right)$,

determines the location of the separating hyperplane by minimizing the misclassification error. The hyperparameter C controls the tradeoff between margin maximization and hinge loss.

Model based transfer regularization formulations start with A-SVM (Li, 2007, Yang et al., 2007b) which was introduced for adapting SVM classifiers for new domains. The key idea is to learn from the source model w^s by regularizing the distance between the learnt model w and w^s. The formulation is:

$$min_{w,b} \quad ||w - \Gamma w^s||^2 + C \sum_{i}^{N} \max\left(0, 1 - y_i(w^\mathsf{T} x_i + b)\right) \tag{2.2}$$

where Γ is a scalar that controls the amount of transfer regularization, and C controls the weight of the loss function. Subsequently, Yang et al. (2007a) used A-SVMs for cross-domain video concept detection which aims to adapt concept classifiers across various video domains. Tommasi and Caputo (2009) applied A-SVM to the Least Squares SVM framework:

$$min_{w,b} \quad ||w - \Gamma w^s||^2 + C \sum_{i}^{N} \left(y_i - (w^\mathsf{T} x_i + b)\right)^2 \tag{2.3}$$

The main difference in this formulation is the use of squared loss instead of hinge loss. Even though squared loss is more sensitive to outliers, it also provides an analytic least square solution for the objective. Furthermore, it enables efficient leave-one-out cross validation which is used for optimization of the hyperparameter Γ, the amount of transfer. Later on, Tommasi et al. (2010) expanded their formulation for

transferring from multiple source models:

$$min_{w,b} \quad ||w - \sum_{i}^{M} \alpha_i u_i||^2 + C \sum_{i}^{N} \left(y_i - (w^{\mathsf{T}} x_i + b)\right)^2 \qquad (2.4)$$
$$st : 0 \leq \alpha_i \leq 1 \qquad ||\alpha|| \leq 1$$

where u_i are the source models, M is the number of available source models, and α decides the weights of the source models that will be utilized for transfer regularization. Similarly they used leave-one-out cross validation for optimizing α.

Duan et al. (2010) proposed the Adaptive Multiple Kernel Learning (A-MKL), a method that combines A-SVMs and DTSVMs, for a visual event recognition task. In visual classification setting, Luo et al. (2011) proposed another multiple source transfer method using the responses of unconstrained prior models on the target data. They also transferred from one multi-class classification problem to another. Building upon the multiple source A-SVM (2.4) model, Kuzborskij et al. (2013) suggested updating the source classifiers as well while learning the target classifier using a multiple source A-SVM approach.

2.2.4 Benefits of Transfer: Performance Metrics

The aim of transfer learning is to improve the performance of a target solution with the help of already existing knowledge. It is especially effective when the target instances are limited. Therefore the performance of a transfer method can be better analysed when illustrated as a function of number of target instances used in the learning process. (see Figure 2.5) (Olivas et al., 2009, Tommasi, 2013). Transfer

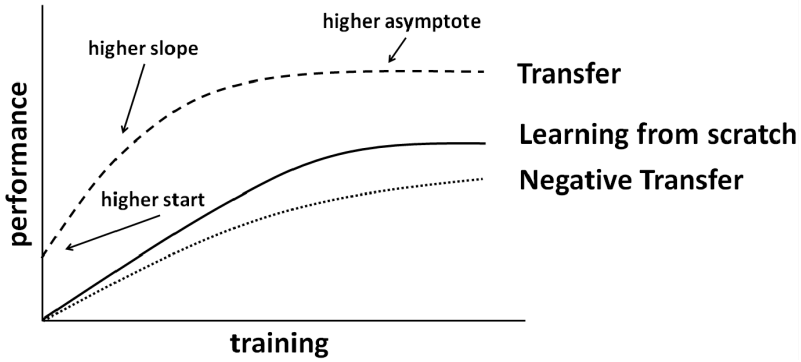


Figure 2.5: **The benefits of transfer learning.** The three types of performance improvement aspirations from transfer learning. The x-axis is the number of training instances for the target problem. (figure from Olivas et al. (2009), Tommasi (2013)).

learning can provide three types of performance improvements over learning from the target data alone.

Higher start. Even with very few target instances, transfer approach performs much better compared to learning from scratch.

Higher slope. The performance of transfer approach grows more quickly when additional target instances are introduced to the learning process.

Higher asymptote: The final performance of the transfer method is superior to the learning target problem alone.

These metrics will be taken into consideration while evaluating the performance of the proposed transfer methods in the forthcoming chapters.

2.2.5 Multi-task Learning

Unlike transfer learning settings, where a transfer is performed from the existing source knowledge to a target problem, multi-task learning (MTL) suggests to learn

all the tasks simultaneously (Caruana, 1997). Particularly when each task has very small number of training instances, MTL performs significantly better than learning each task alone. The underlying intuition behind MTL is to enforce task relatedness while learning each task. Depending on how the task relatedness is encouraged, MTL approaches can be divided into two main groups.

In the first group, classifier vectors are encouraged to be close to each other. This relatedness can either be satisfied by minimizing the Frobenius norms of the classifier vector differences (Evgeniou and Pontil, 2004, Liu et al., 2009, Parameswaran and Weinberger, 2010), or by sharing a common prior (Daumé III, 2009, Lee et al., 2007, Yu et al., 2005). One of the primary studies, regularized multi-task learning of Evgeniou and Pontil (2004), is given below:

$$\min_{w_0,\mathbf{w}} \quad \lambda_1||w_0||^2 + \lambda_2 \sum_{t}^{T} ||w_t - w_0||^2 + \sum_{t}^{T}\sum_{i}^{N} max(0, 1 - y_i^t w_t^{\mathsf{T}} x_i^t) \quad (2.5)$$

where x_i^t are the features, y_i^t are the labels, and w_t are the classifiers for the t^{th} task; w_0 is the shared common vector, T is the number of tasks and N is the number of training samples. The hyper parameters λ_1 and λ_2 control the trade-off between regularization and data error as well as the penalization over the shared vector w_0. The proximity between tasks w_t is achieved by applying relatively less penalization on the shared vector w_0.

In the second group, the model parameters are generated from a common latent feature representation (Amit et al., 2007, Ando and Zhang, 2005, Argyriou et al., 2006, 2008, Caruana, 1997, Harchaoui et al., 2012, Obozinski et al., 2010). This common latent representation is mostly induced by trace norm (nuclear norm) reg-

ularization of the model parameter matrix where each column corresponds to the parameters of a separate task. The general framework of these approaches is as below:

$$\min_W \quad \lambda||W||_* + \sum_t^T \sum_i^N max(0, 1 - y_i^t w_t^\mathsf{T} x_i^t) \tag{2.6}$$

where $W = [w_1, w_2, ..., w_T]$ is the classifier matrix and $||W||_*$ is the trace norm of W (sum of the singular values of W, or l_1-norm of the spectrum). Similar to l_1 norm inducing sparsity in the vectors, trace norm induces sparsity in the spectrum of W hence encouraging low rank solutions of W. This induces similarity between the tasks w_t by enforcing them to lie on a smaller subspace spanned by the singular vectors of W.

In traditional MTL settings, all tasks are equally enforced to be close to each other. Another recent direction of research in MTL is to decide on relationships between tasks (Kang et al., 2011, Kumar and Daumé III, 2012, Zhong and Kwok, 2012) so that different subgroups of tasks can be encouraged to be close with different weights and unrelated tasks would not influence each other.

2.2.6 Transfer Learning for Object Category Detection

Although there exist many image classification papers performing transfer learning (Orabona et al., 2009, Tommasi and Caputo, 2009, Tommasi et al., 2010, Yang et al., 2007b), object detection papers are rather limited due to additional challenges that object detection poses such as proper alignment of training images for detector training, prior to performing the transfer establishing correspondences be-

tween source and target models by considering different sizes, aspect ratios and matching viewpoints of detectors. One branch of papers adapts general person detectors to the specific domains. For instance, Zhang et al. (2008) introduces a Taylor expansion based adaptation method for person detection. Wang and Wang (2011) introduces a framework for adapting pedestrian detectors to traffic scenes, and subsequently Wang et al. (2012) proposes a non-parametric detector adaptation method for adapting general pedestrian detectors to video domain. Sharma and Nevatia (2013) proposes an efficient adaptation procedure using random ferns. Jain et al. (2011) adapts pre-trained cascade of face detectors to a specific photograph. Transfer learning approaches across categories are rather scarce (Gao et al., 2012, Lim et al., 2011). Lim et al. (2011) borrow instances from the source classes and reweight them while learning a target object detector. Gao et al. (2012) transfer local correlation structures across object categories. Salakhutdinov et al. (2011) learns to share visual appearance for multi-class object detection in an MTL framework.

2.3 Datasets

2.3.1 PASCAL Visual Object Classification Dataset

PASCAL VOC dataset (Everingham et al., 2010b) consists of realistic daily life images taken from Flickr (FLICKR). Unlike *caltech* (Griffin et al., 2007) image classification dataset, in which the images mostly contain a single well-centred object occupying the entire image, PASCAL VOC objects are scattered around the image, may be occluded or truncated, and more than one categories might exist in the im-

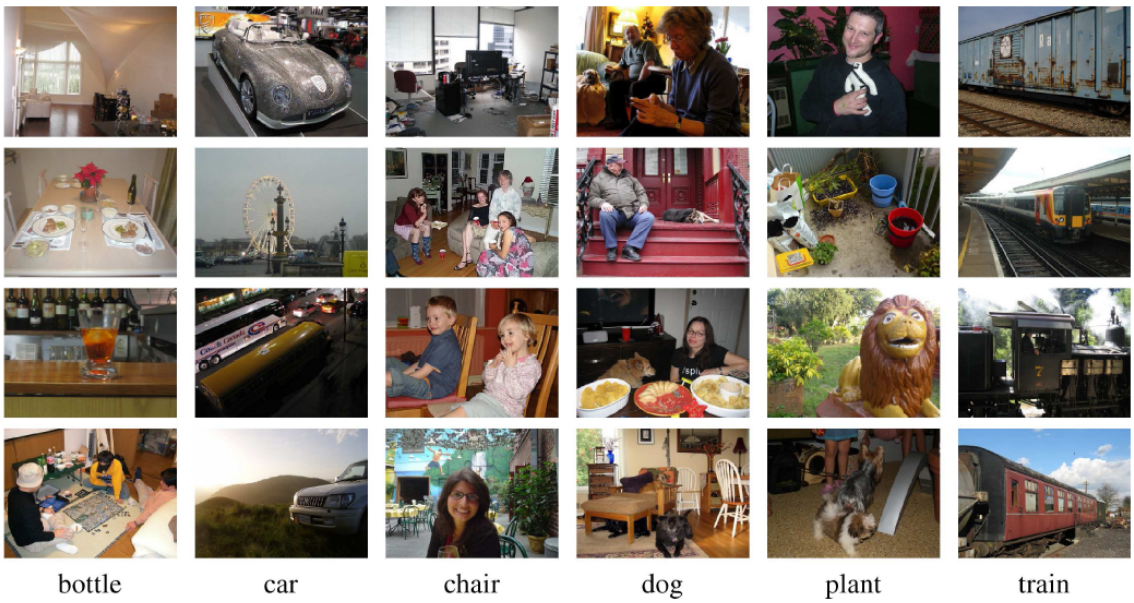


Figure 2.6: **Example images from the PASCAL VOC dataset.** Note large intra-class variation, and changing location and scales of the objects.

age. Intra-class variation is high and objects are viewed from multiple viewpoints. This dataset is composed of 20 object categories organized in four major groups: (a) Person: person, (b) Animal: bird, cat, cow, dog, horse, sheep, (c) Vehicle: aeroplane, bicycle, boat, bus, car, motorbike, train, and (d) Indoor: bottle, chair, dining table, potted plant, sofa, TV/monitor. Some example images from the dataset are shown in figure 2.6. Location and scale annotations are in the form of tight bounding boxes around the object. Objects are also annotated by their pose, occlusion and truncation properties. Four main poses exist namely *left, right, frontal, rear* and each pose accepts up to ± 20 degrees separation from its canonical view. An additional pose annotation, "unspecified", is used for the objects which can not be reliably assigned to one of the available poses or when the pose does not exist at all (i.e. potted plant). This dataset is a challenging dataset and mainly used for evaluation of object category classification/detection systems. Using this dataset,

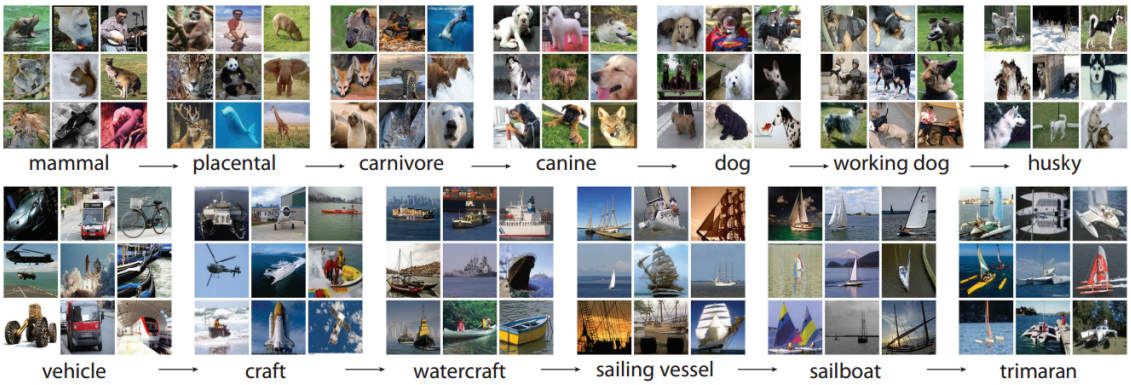


Figure 2.7: **Example images from the ImageNet dataset.** Illustration of two specific synsets (i.e. *husky* and *trimaran*) and their classification in the taxonomy. The top row is from the mammal sub-tree and the bottom row is from the vehicle sub-tree of ImageNet (Deng et al., 2009).

object category classification/detection competitions are held annually for evaluation of state-of-the-art systems (Everingham et al., 2010b). The dataset is organized to training, validation and test data. Except for VOC2007 the test annotations are not available and the methods are evaluated through an evaluation server. Most of the experiments in this thesis are performed on PASCAL VOC2007 data which contains 9,963 images and 24,640 annotated objects.

2.3.2 ImageNet Dataset

ImageNet (Deng et al., 2009) is a large-scale image dataset which built upon the backbone of WordNet structure. WordNet is a large scale lexical database. It contains nouns, verbs, adjectives and adverbs which are grouped into sets of distinct concepts named as synsets. ImageNet focuses on the nouns of WordNet. It consists of $\sim 15M$ images collected for $\sim 22K$ categories with an average of 500-1000 clean and full resolution images per category. The objects in the images are mostly well-

centred and dominated by the category which it is assigned to. Bounding box annotations are also available for a subset of concepts (300-500 annotations for $\sim 1K$ concepts). Similar to WordNet the dataset is organized as a semantic hierarchy of concepts. Some subgroups of the taxonomy is shown in the figure 2.7. This dataset is used in large scale classification/detection competitions for identifying the state-of-the-art large scale systems.

Chapter 3

Exploiting Models for Single Source Transfer

Given a pair of source and target category detection tasks, this chapter focuses on how to transfer between detector models. There has been considerable progress recently in object category *detection*. Indeed, for certain types of object categories and images (e.g. compact objects, typical viewpoints), discriminatively trained template part-based detectors perform very well, and source code is freely available for training and development (Felzenszwalb et al., 2010a). The negative though is that current methods require training the detector from scratch for each new category – a costly procedure which requires an adequate supply of positive and negative annotated data. For challenges like ImageNet (Deng et al., 2009) and beyond, where the goal is thousands of categories, this procedure is not scalable for object category detection.

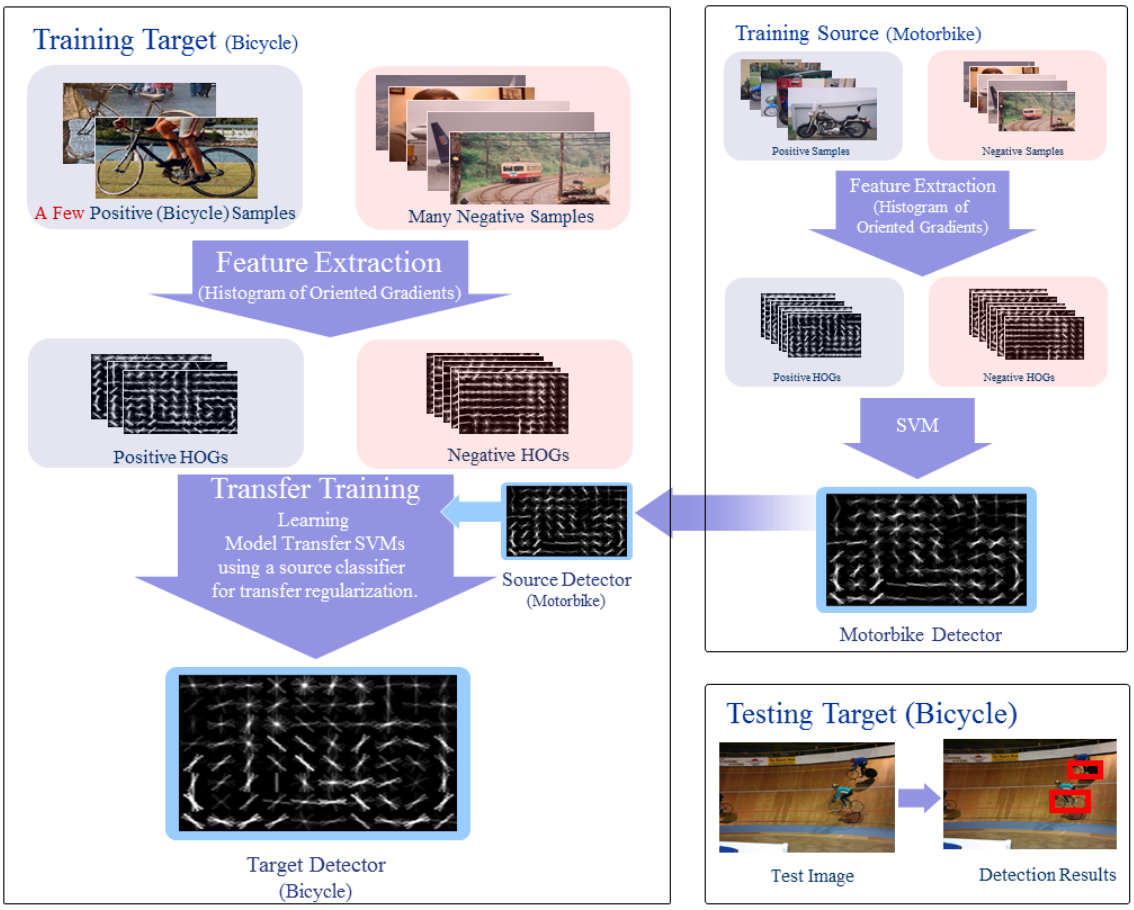
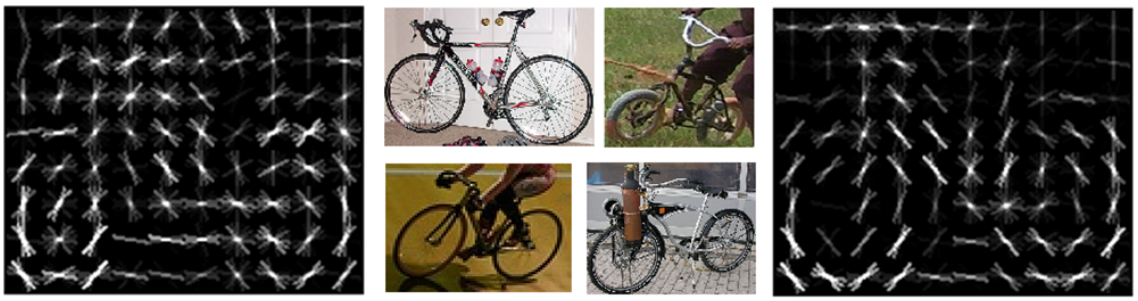


Figure 3.1: **Transfer learning framework.** A target category (i.e bicycle) with a few samples is learned by building upon the source detector (i.e motorbike). Later on tested for the target category.

As an alternative solution to tabula rasa (i.e. learning from scratch), that we investigate in this chapter, is to benefit from category detectors that have previously been learnt for *similar* categories by *transferring* information to a new target category.

In particular, our objective is to apply transfer learning to the SVM discriminative training framework for HOG template models of Dalal and Triggs (2005) and Felzenszwalb et al. (2010a). Figure 3.1 illustrates the overall framework of our transfer learning approach. The key intuition is that the learnt template records the



(a) (b) (c)

Figure 3.2: **The benefit of transfer learning**. The learnt HOG detector template for a motorbike (a) is used as the source for learning a bicycle template together with the samples shown in (b). The resulting learnt bicycle HOG detector template (c) clearly has the shape of a bicycle. Note, here and in the rest of the paper we only visualize the positive components of the HOG vector.

spatial layout of positive and negative orientations. Classes that are geometrically similar (e.g. a horse and a donkey) – those that can be 'morphed' into one another by local deformations – will have correspondingly similar HOG templates. To achieve this transfer in the target detector training, the previously learnt template is introduced as a regularizer in the cost function. As illustrated in figures 3.1 and 3.2, this enables a detector to be learnt for the target category using substantially fewer samples than tabula rasa.

To this end, we introduce and compare three models, including a novel cost function for rigid template transfer and a *geometric* transfer model where the template is deformed using local flow. All three models are defined by convex optimization problems. We also show that the selection of training samples is critical but can be determined by the source category for best results in the case of one-shot learning.

In the next section we define the problem, and then introduce three model transfer methods for object detection building on an existing model transfer method of Li

(2007), Yang et al. (2007b). In the experiments (section 3.3) we compare transfer learning models for both one-shot learning and multiple samples in the case of transferring from one category to another; and also investigate the case of specialization from a class to a subordinate class.

3.1 Model Transfer Support Vector Machines

Suppose that we wish to detect an object category of interest, which will be referred as the *target category*, and assume that we have a well trained detector for a visually similar *source* category. Then, the goal is to learn an object detector for the target category by transferring knowledge from the source category detector with the guidance of a few available samples of the target category.

We are concerned here with detection using a sliding window classifier in the manner of Dalal and Triggs (2005). The classifier is linear and is specified by a template vector w, with a scoring function $w^{\mathsf{T}}x$, where x is the feature vector. Then the task is to learn w for the target category using a few positive training instances x_i, and the source category detector w^s.

In the remainder of the section we introduce and motivate model based transfer learning for support vector machines (SVM). Three variants will be defined: Adaptive Support Vector Machines (A-SVM), a direct application of the classifier transfer of Li (2007), Yang et al. (2007b); Projective Model Transfer SVM (PMT-SVM) which relaxes the transfer of the A-SVM model; and Deformable Adaptive SVM (DA-SVM), where the source template w^s is geometrically transformed during the learning.

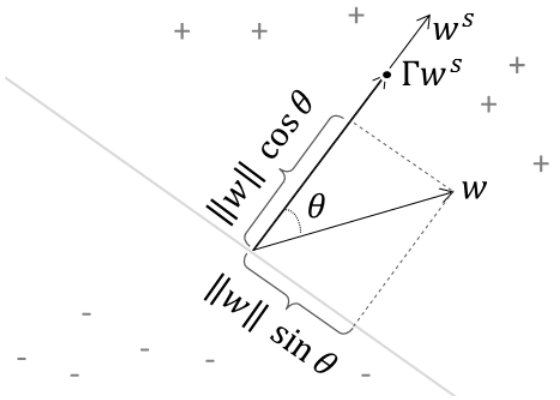


Figure 3.3: Visualization of the projection of the vector w onto w^s, and onto the separating hyperplane orthogonal to w^s.

3.1.1 Adaptive SVM (A-SVM)

As it is already mentioned in the literature review (Li, 2007, Yang et al., 2007b), the idea is to learn from the source model w^s by regularizing the distance between the learned model w and w^s. As usual, x_i are the training samples, $y_i \in \{-1, 1\}$ the corresponding labels, and $l(\mathbf{x}_i, y_i; w, b) = \max\left(0, 1 - y_i(w^\mathsf{T} x_i + b)\right)$ the hinge loss. The objective function is:

$$L_A = \min_{w,b} \|w - \Gamma w^s\|^2 + C \sum_i^N l(\mathbf{x}_i, y_i; w, b) \tag{3.1}$$

where Γ controls the amount of transfer regularization, C controls the weight of the loss function, and N the number of samples.

Discussion. We present a brief analysis of A-SVMs to motivate our second model. Intuitively transfer regularization for an A-SVM is like a spring between Γw^s and w,

and is equivalent to providing training samples from the source class. The transfer can also be understood by expanding the regularization term. Assume that w^s is l_2 normalized to 1 then

$$\|w - \Gamma w^s\|^2 = \|w\|^2 - 2\Gamma\|w\|cos\theta + \Gamma^2 \tag{3.2}$$

where $\|w\|^2$ provides the 'normal' SVM margin maximization and $-2\Gamma\|w\|cos\theta$ induces the transfer by maximizing $cos\theta$, i.e. by minimizing the angle θ between the w^s and w as shown in figure 3.3. Note, that the transfer term $\|w\|cos\theta$ is maximized (and thus the cost minimized) when $\theta = 0$.

However, the term $-2\Gamma\|w\|cos\theta$ also encourages $\|w\|$ to be larger (as this reduces the cost) and this prevents margin maximization. Thus Γ, which should define the amount of transfer regularization, becomes a tradeoff parameter between margin maximization and knowledge transfer.

3.1.2 Projective Model Transfer SVM (PMT-SVM)

Rather than transfer by maximizing the transfer term $\|w\|cos\theta$, we can instead minimize the projection of w onto the separating hyperplane orthogonal to w^s (and thereby reduce θ, see again figure 3.3). In this approach, we can increase the amount of transfer (Γ) without penalizing margin maximization. The objective function for Projective Model Transfer SVM (PMT-SVM) is:

$$\begin{aligned} L_{PMT} &= \min_{w,b} \|w\|^2 + \Gamma\|Pw\|^2 + C\sum_{i}^{N} l(\mathbf{x}_i, y_i; w, b) \\ st &: w^{\mathsf{T}} w^s \geq 0 \end{aligned} \tag{3.3}$$

where P is the projection matrix $P = I - \frac{w^s w^{s\mathsf{T}}}{w^{s\mathsf{T}} w^s}$, Γ controls the amount of transfer regularization, and C controls the weight of the hinge loss. $\|Pw\|^2 = \|w\|^2 sin^2\theta$ is the squared norm of the projection of the w onto the source hyperplane. $w^{\mathsf{T}} w^s \geq 0$ constrains w to the positive halfspace defined by w^s. As $\Gamma \to 0$, (3.3) becomes a 'classical' SVM objective. Note that the formulation is convex and can be minimized using quadratic optimization.

3.1.3 Transferring from a deformable source

Transfer regularization can also be performed by using a deformable source template, where small local deformations are allowed for a better fit of the source template to the target. For instance, the wheel part of a motorbike template can be increased in radius and reduced in thickness for a better fit to a bicycle wheel (see figure 3.2). These small deformations provide more flexible regularization. Local deformations are implemented by the flow of weight vectors from one cell to another, as described in more detail in section 3.2. The deformation is governed by the following flow definition:

$$\tau(w^s)_i = \sum_j^M f_{ij} w_j^s,$$

where τ denotes the flow transformation, w_j^s is the j^{th} cell in the source template, the flow parameters f_{ij} define the amount of transfer from the j^{th} cell in the source template to the i^{th} cell in the transformed template. Note that one source template cell can contribute to multiple transformed template cells.

To generalize from the rigid A-SVM to deformable transfer formulation, w^s in (3.1) is replaced with $\tau(w^s)$ which gives the following Deformable Adaptive SVM (DA-SVM) objective:

$$\begin{aligned} L_{DA} &= \min_{f,w,b} \|w - \Gamma\tau(w^s)\|^2 + C\sum_i^N l(\mathbf{x}_i, y_i; w, b) \\ &\quad +\lambda\left(\sum_{i\neq j}^{M,M} f_{ij}^2 d_{ij} + \sum_i^M (1 - f_{ii})^2 d\right) \end{aligned} \tag{3.4}$$

where d_{ij} is the spatial distance between the i^{th} cell and j^{th} cell, d is the penalization for the additional flow from the i^{th} source cell to the i^{th} target cell, and λ the weight of the deformation.

Again, the hyper-parameter Γ controls the amount of transfer. The additional parameter λ controls the deformability: high values of λ make the model more rigid

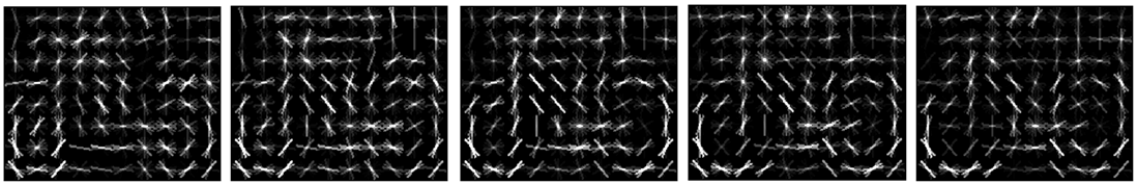


Figure 3.4: Bicycle classifiers learned using a motorbike classifier as the source and with an increasing number of samples (1,3,6,9,20) from left to right. Note the transition from a template that looks like a motorbike (left) to one that looks like a bicycle (right). A-SVM method is used for learning.

so that the solution of (3.4) approaches that of (3.1), conversely small values allow a very flexible source template with less regularization ability.

Discussion. The DA-SVM objective (3.4) defines a convex optimization problem. Even though the term $\|w - \Gamma\tau(w^s)\|^2$ may appear to be non-convex (due to the product of the terms in f_{ij} and w_k), a short calculation shows that the Hessian matrix is positive definite.

3.1.4 Introducing latent variables

In a similar manner to Felzenszwalb et al. (2010a), latent variables can be introduced that specify the position and scale of the detection ROIs relative to the annotation ROIs. As demonstrated in Felzenszwalb et al. (2010a) the introduction of latent variables boosts the performance of the detector, because the training is no longer adversely affected by variations in the annotation and can use each training sample more fully. The disadvantage is that the complete optimization problem is no longer convex.

In more detail, the optimization problem becomes $L_{latent} = \min_{z \in Z(X)} L$, where L

can be any of the model transfer objectives, $Z(X)$ defines all possible bounding box positions and scales of positive and negative samples, and z selects from these sets. In brief, for positive samples z defines a better alignment and for negative samples it defines the boxes that are more confused with positives and harder to classify as negative. Even though L_{latent} is not convex, it becomes a convex objective function when z is fixed, as all the transfer model objectives are convex.

3.2 Implementation details

HOG features. The features used are HOG (Dalal and Triggs, 2005) with the extensions proposed by Felzenszwalb et al. (2010a) and using their source code. Each HOG cell is represented by a 32 dimensional vector. The feature vector x thus has dimension 80×32. As in Felzenszwalb et al. (2010a) we use a 8×10 arrangement of HOG cells. The transfer experiments are based only on the root filter of Felzenszwalb et al. (2010a) without parts. Most of the experiments use a single component, but multiple components are used in section 3.3.1.3.

Flow. The weight vector w corresponds to the grid of $M = 80$ HOG cells tiled in a rectangle, with each cell represented by a $D = 32$ dimensional vector (so w has dimension $M \times D$). Write $\mathbf{h}_j$ for the D dimensional part of w corresponding to the HOG cell j (so that w is a concatenation of M vectors $\mathbf{h}_j$, $j = 1 \dots D$). The flow is restricted to act on the entire vectors $\mathbf{h}_j$, as $\sum_j f_{ij}\mathbf{h}_j$ i.e. it does not mix components of the cell vectors. This is a simplification that could be removed if necessary.

Training. We are provided with a training set of annotated images with tight bounding boxes around each positive instance (see section 3.3). During training,

we switch between optimizing latent variables (bounding box positions and scales of positive and negative samples) and SVM objectives. The SVMs are trained either via quadratic programming using the MOSEK (Mosek: Optimization Toolkit, 2010) optimization toolkit, or via stochastic gradient descent using the VLFeat library (Vedaldi and Fulkerson, 2008). In other respects (alternating for latent variables etc) the training follows that of Felzenszwalb et al. (2010a), and in section 3.3 we show that we obtain a similar performance to Felzenszwalb et al. (2010a).

3.3 Experiments

As it is mentioned in section 2.2.4, transfer learning can provide three types of performance improvements over learning from the target class alone (see Figure 2.5): *(1) higher start*, *(2) higher slope*,*(3) higher asymptote*. The experiments are designed to see which of these are achieved by the model transfer methods.

There are two types of experiments: (i) *inter-class transfer* where the transfer is from one category to another; and (ii) *specialization* where the transfer is from superior class to subordinate (e.g. from a generic quadruped category to a specific category).

The evaluations are performed on the PASCAL VOC 2007 dataset Everingham et al. (2007). The training and validation sets are used to learn the detector, and the performance is reported as average precision (AP) on the test set using the standard PASCAL procedure and evaluation software. For efficiency purposes, we also select a smaller test subset referred as *PASCAL-500* which consists of all the positive samples of the target class and a random selection of other images up to

500. The complete test set is referred as *PASCAL-COMPLETE*.

For most of the experiments we restrict the training to *side views* of the categories horse, sheep, cow, motorbike, and bicycle. These are obtained directly from the VOC data using the pose attributes provided in the annotations. In the training only side facing objects are used and the obtained filter is always facing left (i.e right facing samples are mirrored before training). Table 3.1 gives the number of side facing positives in the training and test sets for each class. In section 3.3.1.3 we lift the restriction of side view samples and use all the views for training multiple component models.

Evaluations are performed using two different procedures: (i) *pascal-default*, and (ii) *pascal-side-only*. The *pascal-default* case is the PASCAL VOC (Everingham et al., 2007) evaluation procedure including all views. In *pascal-default* evaluations, while obtaining detections we also use the mirrored version of the side facing detector. In the *pascal-side-only* case, only the left side view ground truth of test samples is used for evaluation and true detections of other poses belonging to the target class are not counted towards AP computation.

The hyperparameters C, Γ and λ are learned on the validation set. C is fixed to 0.002 for all the experiments.

Baseline detectors.

The baseline detectors are SVM classifiers trained directly on positive samples without any transfer learning. These classifiers provide the source models in the transfer experiments, and also establish the performance that can be achieved for the target class if all the positive samples of that class are used for training. Other than the learning procedure, the baseline is essentially the same as the discriminatively

Categories	#pos. samples		Felzenszwalb et al. (2010a)	Base. SVM
	train	test		
horse	45	53	40.1%	**44.6%**
sheep	45	43	30.7%	**37.1%**
cow	38	41	**24.9%**	21.5%
bicycle	62	76	50.1%	**59.0%**
motorbike	36	57	**38.1%**	33.8%

Table 3.1: The comparison of the average precision (AP) results obtained from baseline SVM and Felzenszwalb *et. al.* Felzenszwalb et al. (2010a) without parts for the *pascal-side-only* object detection task on *PASCAL-COMPLETE* test set.

trained detector of Felzenszwalb et al. (2010a) with only the root filter (no parts), and we compare to this method using the code provided by the authors. As shown in table 3.1 the baseline has very similar performance to Felzenszwalb et al. (2010a) (if not better). This establishes the state of the art performance for this dataset.

3.3.1 Between category transfer

In these experiments we investigate two cases: learning a bicycle classifier by transferring from the motorbike classifier, and learning a horse classifier by transferring from the cow classifier. We discuss first one shot learning, i.e. learning from a single positive sample of the target class, and then multiple shot learning.

3.3.1.1 One Shot Learning

The one shot learning scenario investigates the *higher start* aspect of transfer learning benefits (see figure 2.5). Given a single positive target class sample, we compare four learning methods: learning from the given positive sample only (baseline SVM), rigid transfer using A-SVM and PMT-SVM, deformable transfer using DA-SVM.

Evaluations are conducted on the *PASCAL-500* test set using the *pascal-side-only* evaluation procedure. The experiments are performed for each training sample of the target class separately.

We explore the scenario where many samples of the target class are available, and we wish to pick the best one in order to gain the maximum performance from single shot learning. In order to see how these four methods respond to varying the quality of the samples, the training samples are ranked using the source classifier. A few examples from high ranked, medium ranked and low ranked bicycle images are displayed in figure 3.5.

The results for one shot learning are presented in tables 3.2 and 3.3. The APs are averaged over the samples from each quality group, namely: high ranked, medium ranked and low ranked. For all the samples, model transfer methods significantly improve over the baseline SVM which shows that the transfer models enjoy the *high start* benefit. For the high ranked samples, model transfer methods improve approximately 15% over the baseline SVM on both the bicycle and horse detection tasks. The improvement over baseline SVM is around 20% for medium ranked samples in both tasks. On the low ranked samples the baseline SVM is increased from 7% to 21% in horse detection, and 14% to 42% in bicycle detection.

For high ranked samples both PMT-SVM and A-SVM have similar performances. For medium ranked samples PMT-SVM outperforms A-SVM. In general we expect PMT-SVM to outperform A-SVM, as argued in section 3.1. Experimentally, classical SVMs tend to have a very small $\|w\|$ for one positive sample models. But we observe that A-SVM models trained with one positive sample have relatively larger (almost ten times larger) w vectors. This is due to the transfer term which

doesn't let w reduce further after a certain value. This problem is rectified in the PMT-SVM model. Note, that PMT-SVM performs much better using some of the medium ranked samples than using high ranked samples. The explanation is that probably these medium ranked samples are closer to the target hyperplane, and this contributes more to the learning procedure than the high ranked ones. For the low ranked samples, which are either highly occluded, blurred or distorted, A-SVM clearly outperforms PMT-SVM. We conclude that PMT-SVM is highly sensitive to bad (low ranked) samples. DA-SVM performs very similarly to A-SVM for all the samples.

To our knowledge there is no previous work on how to select or weight samples of the target class while performing transfer learning. Under the assumption that the source class is visually similar to the target class, ranking samples with the source detector provides an idea about the quality of available samples. Note, the ranking of samples using the source and full target (i.e. trained with all available samples) classifier has 50% overlap in the top 15 samples, and 66% overlap for the last 15 samples. This source ranking can help during the sample selection or weighting of the samples for transfer learning. Since bad samples clearly deteriorate the performance (see table 3.2 and 3.3), low scored samples either should be removed or at least should be assigned small weights.

3.3.1.2 Multiple shot learning

For the experiments we use a fixed (but random) ordering for four learning methods: target samples only, transfer from the rigid source template using A-SVM and PMT-SVM, and transfer from the deformable source template using DA-SVM. Each

Ranks	Base. SVM	A-SVM	DA-SVM	PMT-SVM
01-15	40.5 ± 07.2	**53.9 ± 04.2**	53.7 ± 04.3	53.5 ± 05.7
16-30	33.0 ± 13.5	52.5 ± 08.3	51.9 ± 08.8	**54.7 ± 05.7**
31-45	26.4 ± 13.3	47.1 ± 07.3	47.1 ± 07.6	**48.5 ± 08.7**
46-60	14.0 ± 09.3	42.4 ± 03.7	**42.5 ± 04.2**	27.8 ± 11.3

Source: motorbike(**44.7%**), **Target:** bicycle(**70.1%**), **Test-set:** PASCAL-500,
Test-procedure: pascal-side-only

In all the tables, test configuration information is given similar to the line above, the values for the source and target are the AP scores of the source(i.e. motorbike) and full target(i.e. bicycle trained with all available samples) detectors on the target task.

Table 3.2: **Average Precision (AP) comparison of baseline and transfer SVMs on the one shot learning task.** Models are learned using one sample of *bicycle* class and the *motorbike* classifier as the source. The top row displays the average AP results using one of the top 15 (high ranked) samples ranked by the source classifier. Next rows display the next 15 in the ranking. Note the tremendous boost obtained by the transfer method compared to the base SVM (without transfer).

Ranks	Base. SVM	A-SVM	DA-SVM	PMT-SVM
01-15	15.0 ± 08.0	30.2 ± 04.0	**30.3 ± 03.5**	30.1 ± 05.4
16-30	07.2 ± 04.1	27.0 ± 04.2	27.0 ± 04.3	**27.7 ± 07.1**
31-45	07.2 ± 08.2	**24.1 ± 06.0**	23.8 ± 05.9	11.9 ± 10.3

Source: cow(**26.1%**), **Target:** horse(**60.2%**), **Test-set:** PASCAL-500,
Test-procedure: pascal-side-only

Table 3.3: **AP comparison of baseline and transfer SVMs on the one shot learning task.** Models are learned using one sample of *horse* class and the *cow* classifier as the source.

#	Base. SVM	A-SVM	DA-SVM	PMT-SVM
1	05.2 ± 05.6	23.6 ± 02.9	**24.1 ± 03.2**	22.7 ± 12.1
2	09.0 ± 07.7	32.3 ± 03.9	32.4 ± 04.9	**34.3 ± 07.3**
3	18.9 ± 10.9	34.7 ± 05.5	35.0 ± 04.7	**36.0 ± 10.5**
4	24.7 ± 12.2	37.1 ± 04.5	**37.7 ± 04.0**	35.0 ± 05.6
5	28.7 ± 09.5	37.7 ± 06.6	**37.9 ± 06.4**	34.8 ± 06.9

Source: cow(**26.1%**), **Target:** horse(**60.2%**), **Test-set:** PASCAL-500, **Test-procedure:** pascal-side-only

(a)

#	Base. SVM	A-SVM	DA-SVM	PMT-SVM
1	26.9 ± 11.2	51.3 ± 04.5	49.9 ± 05.1	**54.9 ± 04.0**
2	48.4 ± 05.0	**55.5 ± 05.8**	55.2 ± 05.1	55.4 ± 06.0
3	46.9 ± 11.0	54.2 ± 07.1	54.1 ± 06.7	**56.4 ± 06.9**
4	48.2 ± 09.5	**56.0 ± 08.5**	55.4 ± 07.3	54.2 ± 06.0
5	52.5 ± 09.1	58.1 ± 06.5	**58.7 ± 05.6**	56.8 ± 06.4

Source: motorbike(**44.7%**), **Target:** bicycle(**70.1%**), **Test-set:** PASCAL-500, **Test-procedure:** pascal-side-only

(b)

#	Base. SVM	A-SVM	DA-SVM	PMT-SVM
1	26.9 ± 11.2	06.0 ± 09.4	06.2 ± 09.3	**27.8 ± 08.1**
2	48.4 ± 05.0	26.4 ± 05.0	27.8 ± 05.4	**50.0 ± 06.0**
3	46.9 ± 11.0	33.3 ± 12.7	33.5 ± 12.5	**51.4 ± 12.9**
4	48.2 ± 09.5	36.3 ± 14.7	36.0 ± 14.3	**51.1 ± 12.7**
5	52.5 ± 09.1	45.4 ± 13.0	45.5 ± 13.4	**56.0 ± 09.3**

Source: horse(**00.9%**), **Target:** bicycle(**70.1%**), **Test-set:** PASCAL-500, **Test-procedure:** pascal-side-only

(c)

Table 3.4: **AP results of baseline SVM and model transfer methods.** Transfers are performed **(a)** from cow to horse, **(b)** from motorbike to bicycle, and **(c)** from horse to bicycle (negative transfer). Leftmost column displays the number of positive samples used for learning. In this, and all the following tables and figures, the experiments are performed five times with different randomized orderings of the positive samples.

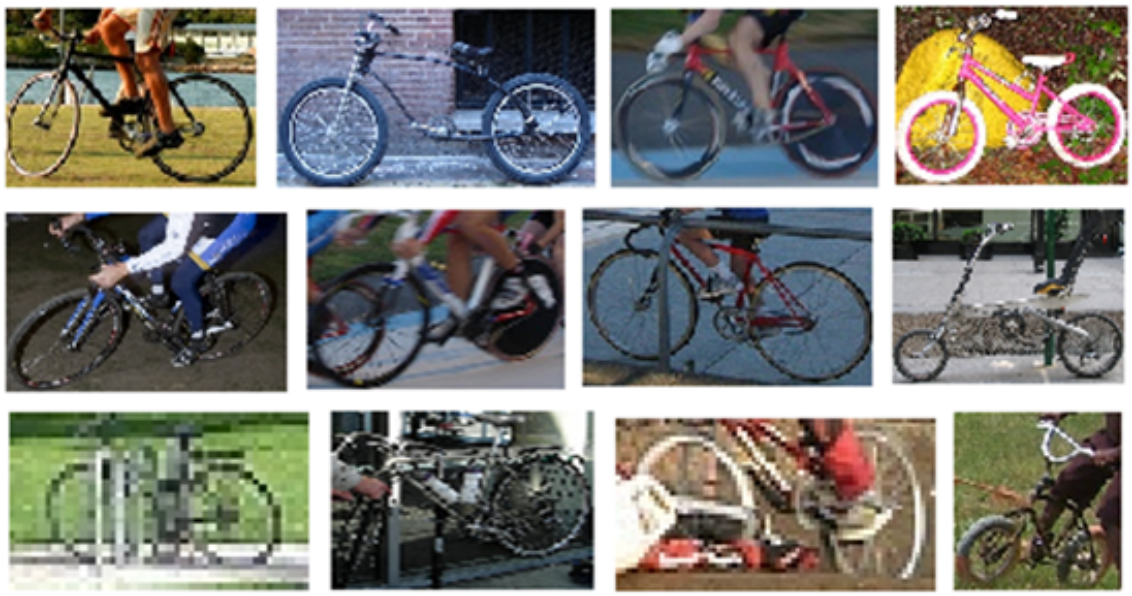


Figure 3.5: **Examples of ranked *bicycle* samples using the *motorbike* detector.** *Top row:* examples from the high ranked (top 15) samples which are generally not occluded, clean and clearly side facing; *middle row:* the middle ranked (other than top or last 15) samples which are mainly clear, might have small occlusion and viewpoint distortions; and *bottom row:* the low ranked (last 15) samples which are either highly occluded, blurred or not good representatives of side facing class.

experiment is repeated 5 times with a different random order. The APs are averaged for each number of samples. The average removes any idiosyncrasies due to particularly good or bad samples turning up early in the training. These fluctuations in AP can be seen in the standard deviations of the results given in table 3.4. The transition of the learned template is illustrated in figure 3.4.

As is clear from table 3.4, transfer from the source class using A-SVM, DA-SVM and PMT-SVM performs *significantly* better than the baseline SVM, especially for a small number of positive samples. Note that the standard deviations of all three methods are smaller than the baseline SVM, showing that the idiosyncrasies due to good and bad training samples are better tolerated. As the number of positive sam-

ples increases, the improvements from transfer learning methods over the baseline decreases.

Number of Samples	**Test-procedure:** pascal-side-only			**Test-procedure:** pascal-default		
	Base. SVM	**A-SVM**	**DA-SVM**	**Base. SVM**	**A-SVM**	**DA-SVM**
1	09.3 ± 08.8	28.4 ± 08.1	**28.7 ± 08.2**	07.0 ± 04.4	14.9 ± 02.5	**15.3 ± 02.5**
3	34.2 ± 11.5	40.9 ± 06.1	**42.1 ± 05.7**	18.6 ± 05.2	20.1 ± 02.7	**20.6 ± 02.4**
5	41.9 ± 05.9	47.3 ± 04.4	**48.3 ± 03.6**	22.7 ± 02.1	24.0 ± 01.7	**24.5 ± 01.6**
7	44.0 ± 09.9	48.8 ± 08.4	**49.1 ± 07.6**	24.7 ± 04.5	25.2 ± 03.1	**25.5 ± 03.4**
10	49.9 ± 05.4	**52.0 ± 05.9**	**52.0 ± 05.2**	27.1 ± 02.3	27.0 ± 02.0	**27.3 ± 01.7**
15	55.9 ± 06.8	56.0 ± 03.8	**57.0 ± 04.7**	29.6 ± 01.9	29.9 ± 01.2	**30.2 ± 01.0**
20	55.2 ± 03.5	57.0 ± 03.3	**58.0 ± 01.9**	30.1 ± 01.2	31.0 ± 00.9	**31.1 ± 00.8**
30	57.9 ± 02.0	59.0 ± 01.6	**60.3 ± 02.0**	30.7 ± 01.4	31.5 ± 01.3	**31.5 ± 01.3**
50	58.9 ± 01.3	**60.2 ± 01.5**	59.5 ± 00.9	31.6 ± 00.9	**32.3 ± 00.5**	32.2 ± 00.7

Source: motorbike(pascal-side-only:**16.9%**, pascal-default:**13.2%**), **Target:** bicycle(pascal-side-only:**59.0%**, pascal-default:**32.5%**), **Test-set:** PASCAL-COMPLETE

Table 3.5: AP results of baseline SVM and model transfer methods for the *bicycle* detection task.

# of Samples	**3**	**5**	**8**	**10**	**13**	**15**
Base. SVM	10.0 ± 0.8	13.7 ± 4.9	16.6 ± 6.0	16.9 ± 5.7	20.5 ± 4.4	23.4 ± 6.5
A-SVM	**11.9 ± 1.8**	**14.4 ± 4.2**	**17.7 ± 4.0**	**21.9 ± 3.0**	**22.2 ± 3.2**	**25.5 ± 5.0**

# of Samples	**18**	**20**	**23**	**25**	**50**
Base. SVM	**27.8 ± 4.7**	**28.0 ± 4.3**	28.4 ± 3.1	**29.8 ± 5.0**	**36.4 ± 3.4**
A-SVM	26.0 ± 3.3	**28.0 ± 3.7**	**29.7 ± 2.0**	29.4 ± 2.2	33.9 ± 1.9

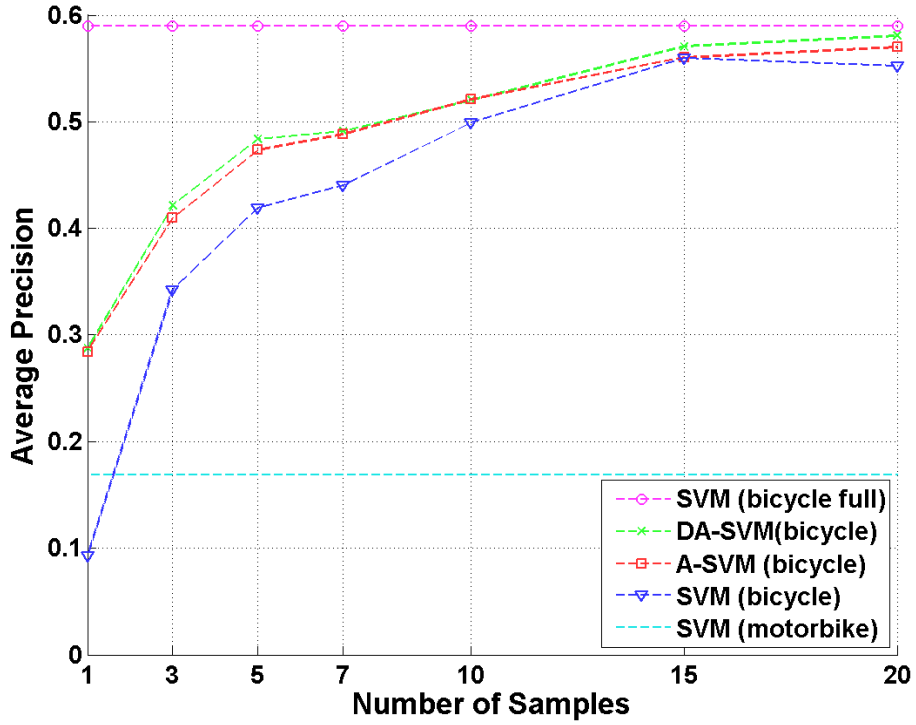
Source: motorbike(**11.9%**), **Target:** bicycle(**46.6%**), **Test-set:** PASCAL-COMPLETE, **Test-procedure:** pascal-side-only

Table 3.6: AP results of multiple component baseline SVM and A-SVM for *bicycle* detection task. For A-SVM, the transfer is performed from a multiple component *motorbike* classifier.

As can be seen from table 3.4, PMT-SVM works better for a small number of samples (1-2-3). In table 3.4(a), one sample PMT-SVM appears worse than A-SVM. In fact, this is caused by one of the orderings where a bad sample is introduced. When we remove that ordering and average the other 4 orderings, for the one sample case, PMT-SVM clearly outperforms A-SVM, with performance 28.12% to 24.64%, respectively. This incident also shows that PMT-SVM is good for one shot learning but it is highly sensitive to sample quality. PMT-SVM also doesn't perform well with large number of samples, probably caused by the introduction of bad samples.

Gao et al. (2012) stated that the "good detectors" share certain low-level statistical properties and these properties can also be transferred while learning the new detector even though the source is not from a visually similar class (e.g. transferring from bicycle to horse class). Transfer is performed by enforcing certain covariance structure in the newly learnt detector. Another way of looking at the same problem is learning detectors in a decorrelated space, in other words, instead of using the covariance matrix for regularizing the learnt detector we can also use it for transforming the feature space, i.e. decorrelation, and learn the detector in the new feature space. In order to see if the transfer is caused by low-level statistics or high level properties that are shared by the source and target classes we also performed experiments for transferring from unrelated classes. This setting is named as negative transfer where we learn a bicycle classifier using a horse classifier as the source. As can be seen from table 3.4(c) A-SVM and DA-SVM perform worse than the baseline. The deterioration of A-SVM and DA-SVM showed that these methods are not very successful for transferring low-level statistical properties. However PMT-SVM still manages to perform some boost over the baseline SVM, which shows that this weaker but flexible model can transfer some low-level statistics at the coarse (objectness) level. Even though transferring from an unrelated class resulted in some improvement, transferring from a visually similar class performed substantially better as opposed to an unrelated class (see table 3.4).

As another type of baseline for the transfer experiments, we include the performance of the source classifier (without transfer) for detecting the target category. This measures the confusion between the two categories. As shown in table 3.4, if more than 1 positive sample is used then the transfer methods outperform the source classifier for the target category detection.



Source: motorbike(**16.9%**), **Target:** bicycle(**59.0%**),
Test-set: PASCAL-COMPLETE, **Test-procedure:** pascal-side-only

Figure 3.6: AP comparison between baseline SVM and model transfer methods on *bicycle* detection task.

We also evaluated A-SVM and DA-SVM using a larger scale of positive samples on the PASCAL-COMPLETE test set (see table 3.5 and figure 3.6). In all the cases A-SVM and DA-SVM perform better that the baseline SVM. DA-SVM is superior to A-SVM except for the 50 samples case.

In summary, the results show a significant improvement through transfer learning in terms of higher start and higher slope (refer to figure 2.5).

3.3.1.3 Multiple shot learning with multiple components

These experiments are conducted on classifiers trained with multiple components (aspects) (similar to Felzenszwalb et al. (2010a)). We compare two methods: baseline SVM, a classifier with multiple components for the root filter learned from the positive samples only; and A-SVM, a multiple component classifier learned by transferring from a multiple component source classifier. The experimental settings are as above using the PASCAL-COMPLETE test set, except now, in training, positive training samples are selected from all poses (not just those of the side views). A model with two distinct components (corresponding to four components once mirrored) is used for the source and the target classifiers. Transfer is performed from the motorbike class to bicycle class.

In the transfer training, each positive training sample is assigned to one of the components of the source classifier, depending on the score of the sample obtained from the source components. Then training is performed in a similar fashion to Felzenszwalb et al. (2010a) except we use transfer SVM training instead of classical SVM training. Similarly to the other transfer experiments, transfer methods achieve a good performance, improving over the classical SVM training, particularly for a small number of samples (see table 3.6). The boost gradually decreases when we increase the number of samples.

3.3.2 Specialization: superior to subordinate transfer

These transfer experiments are conducted on the horse, cow and sheep categories of PASCAL VOC 2007. A 'quadruped' category detector is trained from 100 randomly

Number of Samples	**Test-procedure:** pascal-side-only			**Test-procedure:** pascal-default		
	Base. SVM	**A-SVM**	**DA-SVM**	**Base. SVM**	**A-SVM**	**DA-SVM**
1	03.6 ± 03.8	**21.2 ± 05.5**	20.9 ± 05.6	03.6 ± 03.6	**11.5 ± 04.0**	11.3 ± 04.5
3	14.3 ± 07.6	**29.7 ± 06.0**	29.2 ± 06.0	10.3 ± 02.6	**14.5 ± 03.2**	14.2 ± 03.4
5	20.0 ± 09.0	30.9 ± 04.3	**31.5 ± 03.9**	10.6 ± 01.8	**13.8 ± 03.3**	13.6 ± 03.0
7	25.0 ± 07.3	**32.6 ± 04.7**	32.1 ± 04.4	12.7 ± 02.0	15.2 ± 03.4	**15.3 ± 03.0**
10	29.9 ± 04.3	35.3 ± 03.0	**36.6 ± 02.8**	13.8 ± 03.3	16.0 ± 01.8	**16.2 ± 01.7**
15	35.9 ± 05.7	**37.8 ± 05.6**	37.2 ± 04.7	14.6 ± 02.4	16.0 ± 02.8	**16.1 ± 02.7**
20	40.1 ± 02.8	**40.4 ± 03.3**	40.3 ± 02.9	16.6 ± 01.1	**17.6 ± 01.0**	**17.6 ± 01.0**
30	**45.8 ± 02.6**	43.6 ± 03.5	42.9 ± 03.1	**19.9 ± 00.9**	**19.9 ± 01.4**	19.8 ± 01.9
50	**47.1 ± 02.3**	45.4 ± 01.3	44.0 ± 01.0	**21.1 ± 01.5**	20.6 ± 00.8	20.8 ± 00.4

Source: quadruped(pascal-side-only:**15.4%**, pascal-default:**07.9%**), **Target:** horse(pascal-side-only:**44.6%**, pascal-default:**22.3%**), **Test-set:** PASCAL-COMPLETE

Number of Samples	**Test-procedure:** pascal-side-only			**Test-procedure:** pascal-default		
	Base. SVM	**A-SVM**	**DA-SVM**	**Base. SVM**	**A-SVM**	**DA-SVM**
1	03.9 ± 05.0	12.1 ± 03.1	**12.6 ± 03.3**	04.9 ± 04.1	**10.6 ± 02.3**	**10.6 ± 02.5**
3	03.5 ± 03.9	**12.3 ± 05.9**	12.1 ± 05.8	07.2 ± 03.1	**10.4 ± 03.5**	10.3 ± 03.7
5	07.1 ± 05.8	**13.2 ± 04.2**	13.1 ± 04.0	07.0 ± 03.2	**09.1 ± 02.5**	08.7 ± 02.7
7	08.4 ± 06.1	12.7 ± 04.9	**12.8 ± 05.2**	06.8 ± 03.4	09.5 ± 03.1	**09.6 ± 03.3**
10	10.8 ± 07.0	**14.7 ± 04.6**	14.5 ± 04.5	08.4 ± 02.7	**10.0 ± 02.2**	09.7 ± 02.5
15	14.0 ± 03.0	**15.0 ± 04.5**	14.7 ± 03.6	**10.9 ± 01.2**	10.8 ± 02.1	10.6 ± 01.8
20	13.6 ± 04.7	16.1 ± 03.2	**16.2 ± 03.5**	**11.6 ± 02.5**	11.3 ± 00.9	11.1 ± 01.6
30	17.0 ± 03.4	**17.8 ± 02.9**	**17.8 ± 02.1**	**12.4 ± 01.7**	11.7 ± 00.6	11.5 ± 00.6
50	**21.7 ± 00.0**	18.7 ± 00.0	18.5 ± 00.0	12.6 ± 00.0	11.8 ± 00.0	**13.3 ± 00.0**

Source: quadruped(pascal-side-only:**08.1%**, pascal-default:**07.1%**), **Target:** cow(pascal-side-only:**21.5%**, pascal-default:**14.9%**), **Test-set:** PASCAL-COMPLETE

Table 3.7: AP results of baseline SVM and specialization using model transfer methods for the *horse* and *cow* detection tasks.

selected examples from the horse, cow and sheep categories. It is then specialized to one of those categories by transfer learning using the model transfer methods. We omitted PMT-SVM since it doesn't perform well for a large number of samples. The experimental settings and procedures are the same as multiple shot learning experiments on PASCAL-COMPLETE test set.

Table 3.7 shows that specializing from the quadruped class to a subordinate class using model transfer again gives a significant performance improvement over the baseline SVM, especially for a small number of positive samples. Occasionally, using a large number of samples the baseline SVM can perform better than transfer methods. In our experiments the transfer parameter Γ is fixed, decreasing Γ when we have large number of samples would solve the problem, since our models converges

to the classical SVM formulation when $\Gamma \to 0$.

Discussion. The benefits of training a superior class (quadruped in this case) are that the training samples can come from multiple subordinate classes. This is similar to the case of attribute training. Indeed the superior class need not involve any training from the target class. However, we are restricted here in using PASCAL VOC – since there are so few categories it is not possible to train a superior class without also including all subordinate classes. Nevertheless, the benefit of specializing by transfer learning is well demonstrated.

3.4 Conclusions and Future Work

Almost all object category detection methods to date learn the classifier from scratch – tabula rasa. We have proposed a straightforward modification of the learning objective function which retains the benefits of (i) convexity, (ii) optimization methods honed to the special structure of an SVM, and also brings the benefit of learning with fewer training samples. The model transfer methods can act as a 'power-boost' plug-in to any SVM training scheme.

There are a number clear extensions to the model: (i) So far the transfer learning has been applied to the root filter and multiple component scenario of Felzenszwalb et al. (2010a), the next step is to extend it to multiple parts. (ii) So far a single feature type has been used, but the model can also be extended to multiple features, such as are used in Vedaldi et al. (2009).

3.5 Impact

The work described in this chapter was published in ICCV 2011 with the title "Tabula rasa: Model transfer for object category detection", and it is one of the first published studies on transfer learning applied to object detection. In the past two years, as of September 2013, it has been cited in 23 articles according to Google Scholar, a search engine for academic publications. Being inspired from our work, Gao et al. (2012) suggested an alternative model transfer method that transfers local structured priors obtained from HOG templates. The models introduced in this chapter are also used in image classification. Kuzborskij et al. (2013) applied PMT-SVM to one-versus-all multiclass classification settings where the transfer is applied from one mutli-class problem to another. Donahue et al. (2013) expanded our PMT-SVM method by incorporating a manifold regularization method and applied it to multi-category classification in multi-view data and object detection in videos.

Chapter 4

Multi-Source Part Level Transfer for Exemplar SVMs

Content based image retrieval (CBIR), the problem of searching digital images in large databases according to their visual content, is a well established research area in computer vision. In this work we are particularly interested in retrieving subwindows of images which are similar to the given query image, i.e. the goal is detection rather than image level classification. The notion of *similarity* is defined as being the same object class but also having similar viewpoint (e.g. frontal, left-facing, rear etc.). A query image can be a part of an object (e.g. head of a side facing horse), a complete object (e.g. frontal car image), or a composition of objects (visual phrases as in Sadeghi and Farhadi (2011), e.g. person riding a horse). For instance, given a query of a horse facing left, the aim is to retrieve any left facing horse (intra-*class* variation) which might be walking or running with different feet formations

(exemplar deformation).

Recently exemplar SVMs (E-SVM) (Malisiewicz et al., 2011), where an SVM is trained with only a single positive sample, have found applications in the areas of CBIR (Shrivastava et al., 2011) and object detection (Malisiewicz et al., 2011). Since the E-SVM is trained from a single positive sample (together with many negatives), it is specialized to that given sample. This means that it can be strict (on viewpoint for example), and the negatives give some background suppression. However, the single positive is also a limitation: only so much can be learnt about the foreground of the query (and this can lead to false detections), and more significantly it can lead to lack of generalization. In our context, *generalization* refers to intra-class variation and deformation whilst maintaining the viewpoint. Learning such generalization from a single positive is challenging given the lack of examples of allowable deformations and intra-class variation.

In this work we propose a transfer learning approach for boosting the performance of E-SVMs using part-like patches of previously learned classifiers. The formulation softly constrains the learned template to be constructed from classifiers that have been fully trained (i.e. using many positives). For instance, the neck of a horse can be transferred from the tail of an aeroplane (see figure 4.1), or a jumping bike can borrow part of wheel patches from regular side facing bike or motorbike classifiers (see figure 4.2). The intuitive reason behind borrowing patches from other well trained classifiers is that these classifier patches bring with them a better sense of discriminative features and background suppression. The classifier patches also bring some generalization properties which an E-SVM may lack because it is only trained on a single positive sample. The result of the transfer learning is an enhancement

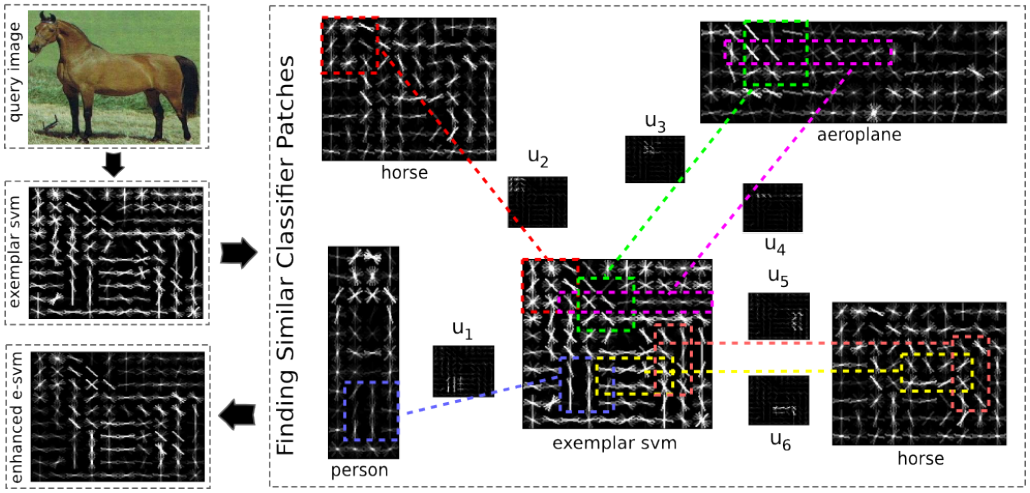


Figure 4.1: **Overview of the EE-SVM learning procedure.** The box on the right shows mining classifier patches from existing classifiers by matching subparts of E-SVM trained from the given query image. Comparing E-SVM and EE-SVM, better suppression of the background can be seen from the visualized classifiers. Note, here and in the rest of the chapter we only visualize the positive components of the HOG classifier.

of background suppression and tolerance to intra-class variation. However, these enhancements incurs no (significant) additional cost in learning and testing. We term the boosted E-SVM, Enhanced Exemplar SVM (EE-SVM).

We describe the relation with the prior work in section 4.1, then introduce the enhanced E-SVM in sections 4.2 and 4.3, and give a quantitative and qualitative evaluation in section 4.5. Although it might be feared that judging the quality of retrieval results will be very subjective, we show that available annotation and measures from PASCAL VOC (Everingham et al., 2010b) can be used for this task.

4.1 Relation to Prior Work

Certainly, a possible solution for improving the E-SVM generalization would be transfer learn on the complete classifier (i.e. use the entire template, see chapter 3). However this requires a visually similar classifier trained with the same object class, pose, scale and aspect ratio to transfer from. With the help of part level transfer, these constraints become less problematic due to the facts that (i) parts can be relocated, (ii) the possibility of finding a good match for transfer increases when we look at smaller classifier patches. Deformable transfer (described in chapter 3) of the complete classifier level would be another alternative solution, however we observed little significant boost over simpler rigid transfer. EE-SVM approach can also be viewed as a deformable transfer considering that parts are being relocated.

Another line of work, that utilizes shared parts across different classes, builds upon the observation that the classes share some common visually coherent substructures, such as wheels, feet, heads, etc. Torralba et al. (2004) introduced a method for sharing small patch oriented templates in a boosting framework, and Opelt et al. (2006) extended this approach to shared boundary fragments. Fidler et al. (2009) explored the shareability of features among object classes in a generative hierarchical framework. Stark et al. (2009) proposed a method for transferring part-like shape features through explicit migration of model parameters for each part, however this transfer is manual at the moment. Ott and Everingham (2011) introduced part sharing across classes for object detection in the framework of discriminatively trained part-based models (Felzenszwalb et al., 2010a). In a slightly different way, our work uses the notion of parts as patches of classifier templates. These patches are mined from a set of previously learned classifiers depending on the quality of

match with the subparts of a E-SVM template.

The proposed approach is mainly described from the transfer learning perspective, however it has very strong relations to the line of work that focuses on enriching the image descriptors with the responses of mid-level and high-level classifiers (Farhadi et al., 2009, Kumar et al., 2009, Lampert and Blaschko, 2009, Li et al., 2010a,b, Song et al., 2011, Yao et al., 2011). All these approaches either replace or augment the original low-level descriptor with the outputs of higher level classifiers. The proposed method also employs a similar augmentation scheme, however we augment the feature vector with the responses of previously learned classifier patches which are selected and relocated based on the quality of match with an E-SVM template learned from the query image.

Combining these two views of the proposed method constructs an equivalence between transfer regularization and feature augmentation. We explicitly prove this equivalence and discuss its implications in chapter 5.

4.2 Enhanced Exemplar SVM

This section discusses the E-SVM formulation and introduces the enhanced E-SVM objective. The formulation of the E-SVM (Malisiewicz et al., 2011) is:

$$min_{w,b} \quad \lambda||w||^2 + \sum_{i}^{N} \max\left(0, 1 - y_i(w^\mathsf{T} x_i + b)\right) \tag{4.1}$$

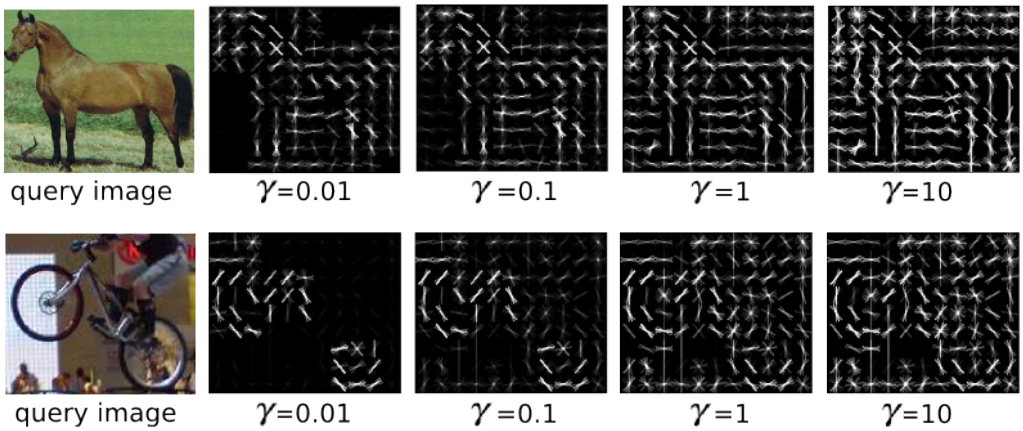


Figure 4.2: **Two limits of EE-SVM from reconstruction($\gamma = 0.01$) to E-SVM($\gamma = 10$).** Learned EE-SVM templates with varying γ values are displayed. λ is fixed to 1.

where λ controls the weight of regularization term, w is the classifier vector, b is the bias term, x_i and y_i are the training samples and their labels, respectively. Note that there is only one positive sample in the training set and its error is weighted more (50 times in Malisiewicz et al. (2011)) than the negative samples. In order to simplify the formulation, different weightings of positive and negative samples are not explicitly shown.

In enhanced E-SVM, part based transfer regularization is incorporated to the E-SVM formulation. The objective is:

$$min_{w,b,\alpha} \quad \lambda||w - \sum_{i}^{M} \alpha_i u_i||^2 + \gamma \sum_{i}^{M} \alpha_i^2 + \sum_{i}^{N} \max\left(0, 1 - y_i(w^\mathsf{T} x_i + b)\right) \qquad (4.2)$$
$$st : \alpha_i \geq 0, \qquad \forall i$$

where λ and γ controls the balance between the two regularization terms as well

as the tradeoff between error term and regularization terms. u_i's are the classifier patches cropped from source classifiers and relocated on a w sized template padded with zeros other than the classifier patch (see Figure 4.1), and α_i's are transfer weights. As α_i's are the reconstruction weights on u_i's and negative use of a part is not possible, non-negativity constraints are imposed on α_i's. Note that given a fixed set of u_i's the formulation is convex.

The two limits of this formulation are E-SVM and reconstruction from the classifier patches. As $\gamma \to \infty$, since α_i's will be forced to be zero due to infinite penalization, $\sum_i^M \alpha_i u_i$ will be a zero vector and (5.2) converges to the E-SVM formulation (4.1). As $\lambda \to \infty$, w will be forced to be equal to $\sum_i^M \alpha_i u_i$ and thus it will be forced to be constructed as a weighted combination of u_i's. Therefore by tweaking λ and γ we can obtain a midway solution between E-SVM and reconstruction from the other classifiers. Figure 4.2 shows the smooth transition from reconstruction to E-SVM by changing γ with a fixed λ.

Discussion. Transfer regularization is introduced with Adaptive SVM (A-SVM) Li (2007), Yang et al. (2007b) which transfers information from a single auxiliary classifier. Subsequently A-SVMs are extended to transfer from multiple classes Yang et al. (2007a). The proposed formulation is also a transfer regularization objective which transfers from the parts of previously learned classifiers. The main difference is that we control the weight of transfer with an additional regularization term $(\gamma \sum_i^M \alpha_i^2)$ where $\gamma \to \infty$ indicates no transfer and $\gamma \to 0$ indicates maximum transfer. The equivalence of this formulation to a "classical" SVM formulation and advantages will be elaborated in the next chapter. Note that this formulation is not specific to E-SVM and this transfer regularization can also be applied to "classical"

SVM formulations.

4.3 Incorporating Part Correlations to EE-SVM

Parts generally occur with certain correlations. The existence of a part would tell a hint about the existence or absence of some other parts. For instance occurrence of a wheel-like part might increase the occurrence probability of another wheel-like part and decrease the existence probability of a cat-face-like part. Consequently, in addition to transferring parts from the well trained source classifiers, we can also transfer statistical correlations between parts. This section will elaborate on incorporation of such pairwise statistical correlations to EE-SVM objective. The new version of EE-SVM will be referred as EE-SVM-COR and the formulation is:

$$\begin{aligned} min_{w,b,\alpha} \quad & \lambda||w - \sum_{i}^{M} \alpha_i u_i||^2 + \gamma \sum_{i}^{M} \alpha_i^2 + \sum_{i}^{N} \max\left(0, 1 - y_i(w^\mathsf{T} x_i + b)\right) \\ & + \theta \left(\sum_{i,j}^{M,M} C_{ij}^{+} ||\alpha_i - \alpha_j||^2 - \sum_{i,j}^{M,M} C_{ij}^{-} ||\alpha_i + \alpha_j||^2 \right) \\ & st : \alpha_i \geq 0, \qquad \forall i \end{aligned} \tag{4.3}$$

where θ is the hyperparameter that controls the weight of enforcing statistical correlations. C_{ij} is the correlation matrix between parts.

C_{ij}^{+} are the positive entires of the correlation matrix C, and C_{ij}^{-} are the negative entires of C. This captures certain statistical correlations which reflects the semantic relations between the parts. Intuitively the first term $\sum_{i,j}^{M,M} C_{ij}^{+} ||\alpha_i - \alpha_j||^2$ causes α_i

and α_j to have similar values if C^+_{ij} is a high positive value and doesn't effect much otherwise. As a consequence, this term encourages them to either be used with similar values or to be both zero. Conversely the second term $\sum_{i,j}^{M,M} C^-_{ij}||\alpha_i + \alpha_j||^2$ enforces α_i and α_j to be far away from each other if they have a high negative correlation. Due to the regularization term $\sum_i^M \alpha_i^2$, whichever of α_i and α_j have stronger support from the data term suppresses the other one. Finer details about enforcing statistical correlations and optimization of this objective will be elaborated in the next chapter.

The entries of the correlation matrix C_{ij} are estimated using the sample correlation coefficient $\rho_{i,j}$ which is computed based on the joint occurrence of parts i and j on the source filters (i.e. source filters are the samples). The computation of the correlation coefficient is given as below:

$$\rho_{i,j} = \frac{cov(i,j)}{\sigma_i \sigma_j} = \frac{E[(i-\mu_i)(j-\mu_j)]}{\sigma_i \sigma_j} \tag{4.4}$$

where *cov* is the sample covariance, σ_i is the standard deviation of occurrence of part i computed over samples (1 states occurrence and 0 states absence of part i in the given sample), μ_i is the mean occurrence of part i across all the samples, and E is the expectation.

Discussion. Another way of enforcing part correlations is to use covariance matrix regularization, i.e. $\alpha^\mathsf{T}\Sigma^{-1}\alpha$, assuming that Σ is a valid covariance matrix computed based on the occurrences of parts on source filters. However, the estimated Σ may not be positive definite. Gao et al. (2012) thought of Σ as an affinity matrix (i.e. stronger correlation means higher affinity). They directly used it for constructing the

precision matrix of a Gaussian and used it in the regularization as $\alpha^{\mathsf{T}}(I-\lambda\Sigma)\alpha$ where λ is a hyperparameter that ensures the positive definiteness. Similarly we treated the correlation matrix C_{ij} as the affinity matrix and enforced part correlations in a similar manner. The detailed derivations of the formulations are given in chapter 5.

4.4 Implementation

In this section the details of the implementation will be discussed. Initially training source classifiers and E-SVM will be described. Afterwards, the EE-SVM training procedure will be explained in two phases: (a) mining regularization parts from source classifiers and (b) optimizing the EE-SVM objective.

The classifiers are linear SVM classifiers (templates) over HOG (Dalal and Triggs, 2005, Felzenszwalb et al., 2010a) features. The source classifiers are trained using Imagenet (Deng et al., 2009) training set using three components for each class without parts, similar to the procedure in Felzenszwalb et al. (2010a) . In total 329 models out of 1000 are selected which perform over 30% AP on a small validation set of 1000 images (50 positive images per class). Together with the mirrored versions of these templates it adds up to $329\times3\times2 = 1974$ source classifiers. To build the vocabulary of classifier patches, all patches with sizes of 5×5 are extracted from the components of the trained DPMs, then a k-means clustering is performed with $k = 10K$ centers.

Each E-SVM is composed of 100 or slightly less HOG cells where the aspect ratio is chosen according to the query image. The E-SVM is trained with the given query image as the positive sample and randomly selected 2000 negative images from the

PASCAL'07 training set. The training is performed iteratively in a similar fashion to Malisiewicz et al. (2011) where mined hard negatives are incorporated to the learning after each iteration.

The training procedure of EE-SVM, which is briefly visualized in figure 4.1, starts with training an E-SVM classifier from the given query image. After obtaining the E-SVM, for each 5×5 cell classifier patch a *good match* is searched for within the vocabulary. This search can be efficiently done using fast nearest neighbour methods. However we performed it as a linear search since we have a limited number of vocabulary items. Even though we use 5×5 cell classifier patches for experimental validation, any other varying size and aspect ratio can also be applied. A *good match* is defined by thresholding the cosine similarity (normalized dot product) between E-SVM patch (a $5 \times 5 \times 32$ dimensional vector) and vocabulary items. This threshold value is fixed to 0.2. After determining where to transfer from, each patch is relocated on a w sized HOG template padded with zeros other than the transferred classifier patch. Finally learning of EE-SVM is performed using the same set of training samples used for training E-SVM and no new hard negatives are collected. Correlations of classifier patch occurrences are computed from the 1974 source classifiers. The optimization of the EE-SVM and EE-SVM-COR objectives are performed using the LIBSVM (Chang and Lin, 2011) package through the equivalent feature augmentation formulations which will be introduced and further analysed in chapter 5. The only additional cost of EE-SVM and EE-SVM-COR over E-SVM is the transformation of training samples, and training another SVM, which constitutes less than 1% of the training time (i.e. mining hard negatives is costly). The test time complexity of EE-SVM and EE-SVM-COR is exactly the same as that of E-SVM.

4.5 Experiments

In this section the experimental results will be described. Initially we'll give the experimental settings, evaluation metrics and the defaults for the hyperparameters. In the next two sections, we'll discuss two sets of experiments performed on PASCAL VOC 2007 (Everingham et al., 2010b) dataset and ImageNet (Deng et al., 2009). Average precision (AP), and precision at top K (PR@5, PR@10, PR@50, PR@100) retrievals are used for evaluating the quality of retrieval results. A correct retrieval is defined as the same object class with the same pose as the query image and the retrieved subwindow should have at least 50% overlap with the true bounding box around the object class. The definition of the pose is inherited from the PASCAL VOC metrics (Everingham et al., 2010b) where four main poses exist namely *left, right, frontal, rear* (the pose "unspecified" is omitted). In all the experiments the proposed approach is compared with E-SVM method. Both λ and γ parameters are fixed to 1 and θ is fixed to 10 in all the experiments unless otherwise stated. The matching similarity threshold, which determines the *good* classifier patch matches based on the normalized dot product of two vectors, is fixed to 0.2.

4.5.1 PASCAL VOC Experiments

The retrievals of PASCAL'07 classes with four main poses are evaluated. The query images are selected as all non-truncated images of the 17 classes (bottle, dining table and potted plant are omitted since they don't have poses) with 4 main poses from the training set. For each query image, an E-SVM and an EE-SVM are trained and run on the test set. Ground truth is identified as the same object class with same

pose label. The detections of the same object class other than the target pose is omitted and not counted towards AP computation. For instance if we are searching for a bicycle facing left, we ignore (i.e neither counts as positive nor negative) the detections of front, rear, left or unspecified poses of bicycle. In total 1659 queries from 17 classes are evaluated, and the pose distribution is: 452 left, 439 right, 561 frontal, and 207 rear.

For some query images, due to being unusual examples of the pose (e.g. left facing bicycle with front wheel up as in figure 4.2), the AP results can be very low. Conversely for some others, which are canonical examples of the pose, the AP results are much higher. In order to have a better insight on the results and see the boost in different quality of samples, we grouped the queries as being above some AP threshold. The query belongs to the quality group $AP > threshold$, if either AP of E-SVM, EE-SVM or EE-SVM-COR is above the defined *threshold* (for instance group $AP \geq 0$ means all the queries). In all the tables improvement in AP is shown as the relative improvement.

Table 4.1 shows the overall results and the AP improvement of EE-SVM and EE-SVM-COR over E-SVM. In all the quality groups EE-SVM significantly improves over E-SVM, and EE-SVM-COR improves over EE-SVM. Even though the actual AP improvements are changing across the quality groups of samples, the relative boost of EE-SVM and EE-SVM-COR are consistently similar in different quality groups. In table 4.2 the AP results and improvements are shown for individual classes for the quality level ($AP \geq 0.05$). For statistical significance only the classes which have more then 5 queries are shown. For all the classes EE-SVM and EE-SVM-COR significantly outperforms E-SVM.

AP $\geq$	**0.00**	**0.01**	**0.05**	**0.10**	**0.15**	**0.20**	**0.30**	**0.50**
# of queries	1659	650	346	241	169	121	70	14
E-SVM	3.3	8.2	14.0	17.9	21.6	25.3	30.0	44.2
EE-SVM	4.2	10.6	17.9	22.8	27.6	32.4	39.1	55.2
EE-SVM-COR	4.3	10.8	18.3	23.4	28.5	33.4	40.4	57.3
Rel. Imp. of EE-SVM	28.3	28.4	27.9	27.6	28.1	27.9	30.1	24.9
Rel. Imp. of EE-SVM-COR	30.9	31.1	30.9	31.0	32.3	32.0	34.7	29.7

Table 4.1: **Relative MAP (Mean Average Precision) improvements of EE-SVM and EE-SVM-COR over E-SVM with changing quality groups.** For instance $AP \geq 0.01$ means the queries which achieved $AP = 0.01$ or above. Queries are the images of 17 classes with four major poses (i.e. left, right, frontal, rear) from PASCAL'07 training set. EE-SVM-COR constantly outperforms EE-SVM which significantly improves over E-SVM.

class	**plane**	**bicycle**	**bus**	**car**	**cow**	**dog**	**horse**	**m.bike**	**sheep**	**tvmonitor**
# of queries	5	59	15	129	13	6	33	28	16	32
E-SVM	5.3	23.3	8.5	16.0	4.9	5.6	9.3	11.3	8.6	10.2
EE-SVM	7.9	30.7	9.0	20.7	7.7	8.9	11.4	13.0	11.9	11.3
EE-SVM-COR	7.1	32.1	8.8	21.0	8.5	10.2	11.6	12.9	12.3	11.3
Rel.Imp.	34.3	38.0	3.4	31.0	74.2	82.8	25.4	13.9	44.3	10.2

Table 4.2: **MAP results and relative improvements of EE-SVM-COR over E-SVM for individual classes for the quality group ($AP \geq 0.05$).** Only classes which have more than 5 queries are shown.

A few qualitative results can be seen in figure 4.3 where the top three positives and negatives are shown with their ranks in the ordered list of retrieved subwindows. In EE-SVM retrievals the ranks of the top three negatives are much later, this shows that EE-SVM better suppresses the negatives and thus increases the recall.

The effect of parameter selections. There are few parameters of the system that can be adjusted for different purposes. For instance, in order to handle the occlusions we need a more aggressive transfer (i.e. small γ), and large patches to transfer from (i.e. 5×5 or 6×6). If the query sample is not a common pose of a common class we can prefer small patches (i.e. 3×3) to increase the chance of a

	lion		deer		tandem		bulldozer		ambulance		MEAN	
	e-svm	ee-svm	e-svm	ee-svm	e-svm	ee-svm	e-svm	ee-svm	e-svm	ee-svm	e-svm	ee-svm
PR@5	0.47	**0.60**	0.60	**0.73**	1.00	1.00	0.93	0.93	1.00	1.00	0.80	**0.85**
PR@10	0.40	0.40	0.50	0.50	1.00	1.00	0.90	0.90	1.00	1.00	0.76	0.76
PR@50	0.19	**0.21**	0.30	**0.32**	0.98	**0.99**	0.62	**0.67**	0.85	**0.87**	0.59	**0.61**
PR@100	0.13	**0.14**	0.22	**0.28**	0.93	**0.97**	0.42	**0.49**	0.76	**0.80**	0.49	**0.54**

Table 4.3: **Precision at top K comparison of ImageNet Queries.** Three queries with varying poses are evaluated for each class from ImageNet and the mean precisions are presented. Retrieval is performed on the collection composed of PASCAL'07 test images and corresponding ImageNet category.

possible match and perhaps decrease the patch similarity threshold.

Handling occlusion and truncation via EE-SVM. It is quite common to came across truncated and occluded query images. With the help of partial classifier patch matchings we can complete the truncated parts or remove the effect of partial occlusion. A partial match is defined as, given a partial match ratio β as a threshold, two classifier patches are matching if any β% subselection of the two patches match with the confidence above the similarity threshold. Figure 4.5 shows two examples of handling occlusion and completing the truncated parts. In these examples 5×5 patches are used and the partial match ratio β is 70%. When we increase the size of the patches even though the chance of finding a good match decreases, if there is one it improves the result drastically (see precision recall curves in figure 4.5). γ, which defines the strength of transfer, is set to 1 for these experiments.

4.5.2 ImageNet Experiments

These experiments are conducted on ImageNet and the transfer is from detectors learnt on PASCAL VOC. For quantitative experiments five ImageNet classes (synsets) are selected: *lion, deer, tandem, bulldozer,* and *ambulance.* For each of

these classes three queries (from one of the main canonical poses) are selected from web images and evaluated on the corresponding ImageNet class images. PASCAL'07 test set is also added to the evaluated samples in order introduce noise in the image database. The evaluations are compared using precision at the the top K retrievals. From the results, displayed in table 4.3, we can conclude that the recall of EE-SVM is much better than the recall of E-SVM, particularly for top 50 and top 100 retrievals.

In addition to canonical poses, the method is also qualitatively demonstrated for unusual poses. With the help of part based transfer, since parts can be relocated and migrated across classes, even for quite unusual poses we can obtain significant improvements. The left facing bicycle with the front wheel up (see figure 4.2 and figure 4.4) is a nice example where the wheel patches are transferred from motorbike and bicycle classifiers with regular poses. Another example, displayed in figure 4.4, is a sitting lion where the ranks of positives clearly show EE-SVM's ability for better recall.

For visual phrases our method successfully reconstructs the classifier template from the patches of existing source classifiers. A visual phrase example (i.e. person riding horse) is also displayed in figure 4.4.

4.6 Conclusion

As has been shown, part level transfer regularization can be used not only for enhancing classifiers, but also for going beyond the spatial extent of the query by completing occlusions and truncations via partial part matchings.

Figure 4.3: **Retrieval results of PASCAL'07 queries.** Top 3 positives and negatives are being displayed. Orders in the ranked list is shown left bottom corner of each image.

Figure 4.4: Retrieval of unusual poses on ImageNet. A visual phrase retrieval is also shown on the rightmost column.

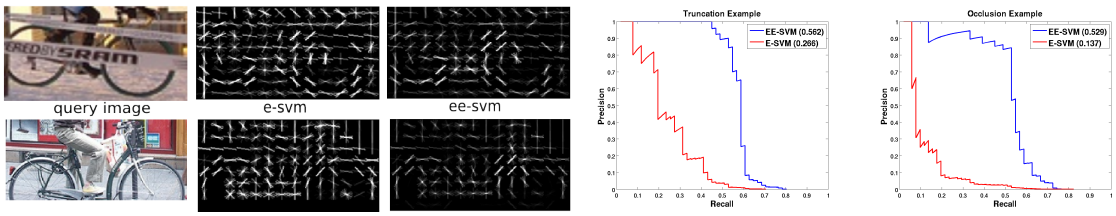


Figure 4.5: Occlusion and truncation handling via EE-SVM. It is better visualized with zooming into the document.

Chapter 5

Relating Feature Maps, Transfer Learning and Optimization

Objects and parts don't occur in isolation to each other. They appear with certain correlations in nature. For example, we don't expect to see a zebra in a city scene with road and cars, or a bicycle next to a sailing boat in the middle of the sea. Stemming from these observations, co-occurrence statistics are utilized in the computer vision problems such as object detection (Desai et al., 2011), and semantic segmentation and labelling of objects (Ladicky et al., 2010, Rabinovich et al., 2007) in the scenes. In chapter 4, we also applied co-occurrence statistics for enforcing the correlation between classifier parts used for softly reconstructing a HOG template. Together with some theoretical elaborations, in this chapter we'll discuss how that convex pairwise potential function, which can be conveniently incorporated to regularized risk minimization frameworks, is constructed.

The outstanding performance of SVMs resulted in widespread popularity of SVM frameworks which became the major tool for many machine learning solutions in a variety of domains, such as computer vision, natural language processing, speech recognition, etc. Due to this popularity many robust tools for linear (e.g. LIBLINEAR (Fan et al., 2008)) and non-linear (e.g. LIB-SVM (Chang and Lin, 2001)) SVM optimization are already developed and tested throughout many applications over the past years. One common approach for minimizing more complex energy functions (e.g. non-linear SVMs, max-margin multi-class classification objectives) is to map them to a linear SVM objective (through a feature map), and solve them more robustly and efficiently with the help of existing tools.

Utilization of feature maps is in the core of the large margin classification frameworks. The very idea behind the non-linear SVMs is implicitly mapping the features into higher (potentially infinite) dimensional feature spaces where a linear separation in this space corresponds to a non-linear decision boundary in the original feature space. This method is also known as the "kernel trick" (Joachims, 1999). Feature mapping can also be performed explicitly so that the non-linear SVM problem can be solved more efficiently as a linear classification problem in the high dimensional feature space directly (Vedaldi and Zisserman, 2010, 2011). Another well known feature mapping approach is commonly used in multi-class classification problems. Tsochantaridis et al. (2005) replicated the input features for all the class labels and transformed the multi-class classification problem into a linear binary classification problem. In the transfer learning field, Daumé III (2009) introduced a feature mapping approach where the features of source and target instances are augmented via appropriate replication.

In this chapter, initially we will investigate transforming transfer learning formulations to a "classical" SVM formulation, which comes with the benefit of using easy and robust optimization for transfer learning approaches and makes it potentially possible to use for practical purposes without the need of expert knowledge in transfer learning. Several equivalence relations between transfer regularization and feature mapping will be investigated, and their corresponding implications will be discussed. Afterwards, the incorporation of co-occurrence statistics to transfer regularization will be articulated, and appropriate feature maps specific to robust optimization of this new objective will be constructed.

5.1 Transfer Regularization by Feature Mapping

Transfer regularization (Yang et al., 2007b) is applied for many transfer learning approaches in object classification and detection. The general form of the formulation used by Tommasi et al. (2010) for transferring from multiple sources was already given in (2.4) in the literature review. Due to the robustness of the hinge loss over the squared loss, we substitute the squared loss term with the hinge loss and obtain:

$$min_{\alpha,w,b} \quad \lambda||w - \sum_{i}^{M} \alpha_i u_i||^2 + \sum_{i}^{N} \max\Big(0, 1 - y_i(w^\mathsf{T} x_i + b)\Big) \qquad (5.1)$$
$$st : ||\alpha|| \leq 1$$

where u_i's are the source classifiers and α_i's are the corresponding weights of transfer for each source classifier. After a slight modification to the transfer regularization

formulation (i.e. from (5.1) to (5.2) by bringing the constraint into the objective), we'll transform the problem to a "classical" SVM formulation through a feature augmentation approach. The derivation below steps through the rearrangements for mapping transfer regularization objective to an equivalent "classical" SVM formulation where the feature vector is augmented with the responses of source classifier models. Let $w = \hat{w} + \sum_i^M \alpha_i u_i$, then the derivation is:

$$\lambda||w - \sum_i^M \alpha_i u_i||^2 + \gamma \sum_i^M \alpha_i^2 + \sum_i^N \max\left(0, 1 - y_i(w^\mathsf{T} x_i + b)\right) \tag{5.2}$$

$$= \lambda||\hat{w}||^2 + \gamma \sum_i^M \alpha_i^2 + \sum_i^N \max\left(0, 1 - y_i\left(\hat{w}^\mathsf{T} x_i + \left(\sum_i^M \alpha_i u_i\right)^\mathsf{T} x_i + b\right)\right) \tag{5.3}$$

$$= ||\bar{w}||^2 + \sum_i^N \max\left(0, 1 - y_i(\bar{w}^\mathsf{T} \bar{x}_i + b)\right) \tag{5.4}$$

where

$$\bar{w} = [\sqrt{\lambda}\hat{w}; \sqrt{\gamma}\alpha_1; \sqrt{\gamma}\alpha_2; ...; \sqrt{\gamma}\alpha_M],$$

$$\bar{x}_i = [\tfrac{1}{\sqrt{\lambda}} x_i; \tfrac{1}{\sqrt{\gamma}} u_1^\mathsf{T} x_i; \tfrac{1}{\sqrt{\gamma}} u_2^\mathsf{T} x_i; ...; \tfrac{1}{\sqrt{\gamma}} u_M^\mathsf{T} x_i],$$

$\bar{w}$ is the transformed classifier and $\bar{x}_i$ is the augmented feature vector with the responses of u's on x_i. The classifier w, the solution to the original problem (5.2), can easily be computed from $\bar{w}$ since $w = \hat{w} + \sum_i^M \alpha_i u_i$. As it is clear from (5.4), the transformed problem is a "classical" SVM formulation with feature augmentation, and it can be optimized efficiently using existing powerful SVM solvers. Note that

this derivation is applicable to both EE-SVM objective and previously used transfer regularization formulations.

Discussion. The major implication of this derivation is that transfer regularization can also be stated as a classical SVM minimization problem where the feature vector is augmented with the responses of source classifiers. This equivalence constructs a bridge between papers implementing feature augmentation or populating the features with the responses of high-level classifiers Douze et al. (2011), Farhadi et al. (2009), Kumar et al. (2009), Lampert and Blaschko (2009), Luo et al. (2011), Torresani et al. (2010), Yao et al. (2011) and papers performing transfer regularization Tommasi and Caputo (2009), Tommasi et al. (2010), Yang et al. (2007a,b). Another direct implication is that transfer regularization approaches Tommasi and Caputo (2009), Tommasi et al. (2010), Yang et al. (2007a,b), which requires specialized optimization, can be reformulated to be efficiently optimized with the state-of-the-art SVM solvers.

Discussion. Another way of converting (5.2) to a normal SVM formulation would be solving α analytically and substituting it in the original equation. Let the regularization part of (5.2) be $R = \lambda||w - U\alpha||^2 + \gamma||\alpha||^2$ then the derivation is as follows:

$$\frac{\partial R}{\partial \alpha} = 0 \qquad \rightarrow \qquad \alpha = \lambda(\gamma I + \lambda U^{\mathsf{T}} U)^{-1} U^{\mathsf{T}} w \tag{5.5}$$

$$\tag{5.6}$$

By substituting α in R we obtain:

$$M = \sqrt{\lambda}(I - \lambda U(\gamma I + \lambda U^{\mathsf{T}}U)^{-1}U^{\mathsf{T}}) + \lambda\sqrt{\gamma}(\gamma I + \lambda U^{\mathsf{T}}U)^{-1}U^{\mathsf{T}} \tag{5.7}$$

$$R = ||Mw||^2 = w^{\mathsf{T}}M^{\mathsf{T}}Mw \tag{5.8}$$

Similar to the derivation in 5.4, this regularization can also be implemented as a feature transformation (i.e. $\bar{w} = Mw, \quad \bar{x}_i = (M^{\mathsf{T}})^{-1}x_i$) and solved using a normal SVM formulation.

5.2 Enforcing Co-occurrence Statistics by Feature Mapping

In this section, initially incorporating co-occurrence statistics in the regularized risk minimization frameworks, particularly SVMS, will be investigated. Afterwards, robust and efficient optimization of the proposed energy through the use of feature maps will be elaborated.

5.2.1 Co-occurrence Statistics for SVMs

One common approach of enforcing pairwise statistics is performed through pairwise potential functions using Markov Random Fields (MRF) or Conditional Random Fields (CRF) frameworks (Bishop, 2006). These potential functions are mostly non-convex and

defined over discrete random variables, however, they can be efficiently optimized using graph-cuts or linear programming relaxations. Unfortunately they are not directly applicable for the SVM framework. Here we introduce a pairwise potential function that is convex and can be conveniently applied to convex regularized risk minimization frameworks, particularly SVMs. We are particularly interested in enforcing pairwise statistics in the transfer regularization formulations (i.e. Eq. 4.3) where the correlation is enforced to the activation of classifier parts u_i's in order to construct the classifier w. Assuming that a positive value of α_i represents the activation of the part u_i and $\alpha_i = 0$ indicates that u_i is not activated, then the pairwise statistics can be captured through correlations between α_i, α_j pairs. The energy function to enforce this statistics can be defined as:

$$min_{\alpha} \quad \phi(\alpha) = \sum_{i,j} C_{ij} ||\alpha_i - \alpha_j||^2 \tag{5.9}$$

where C_{ij} is the correlation between the variables α_i and α_j. Then, the complete transfer objective is:

$$\begin{aligned} min_{w,b,\alpha} \quad \lambda||w - \sum_{i}^{M} \alpha_i u_i||^2 + \gamma \sum_{i}^{M} \alpha_i^2 + \sum_{i}^{N} \max\Big(0, 1 - y_i(w^{\mathsf{T}} x_i + b)\Big) \\ +\theta \sum_{i,j}^{M,M} C_{ij} ||\alpha_i - \alpha_j||^2 \qquad st : \alpha_i \geq 0, \qquad \forall i \end{aligned} \tag{5.10}$$

Intuitively a positive C_{ij} will enforce the values of α_i and α_j to be as close as possible, therefore if one activated the other will be activated as well, especially when C_{ij} is a very high positive value. Conversely, a negative C_{ij} will force them to be as different as possible.

However this objective is not necessarily convex, particularly when C_{ij} contains negative values. In order to obtain a convex energy function, we introduce a slight addition to the pairwise potential and define $\bar{\phi}$ as:

$$\bar{\phi}(\alpha) = \sum_{i,j} C_{ij}||\alpha_i - \alpha_j||^2 - 4\sum_i D^-_{ii}\alpha_i^2 \tag{5.11}$$

$$= \sum_{i,j} C^+_{ij}||\alpha_i - \alpha_j||^2 - \sum_{i,j} C^-_{ij}||\alpha_i + \alpha_j||^2 \tag{5.12}$$

where C is decomposed into its positive and negative components as $C = C^+ + C^-$, D^+ is a diagonal matrix with entries $D^+_{ii} = \sum_j C^+_{ij}$, and D^- is a diagonal matrix with entries $D^-_{ii} = \sum_j C^-_{ij}$. Intuitively by introducing some more penalization (i.e. $-4\sum_i D^-_{ii}\alpha_i^2$) over highly negatively correlated α_i's, which is sensible since these α_i's have smaller probabilities to be activated, we can obtain a convex pairwise potential (5.12). Finally by plugging $\bar{\phi}(\alpha)$ into (5.10) as the pairwise potential, we obtain a convex minimization objective that combines transfer regularization and co-occurrence statistics of the parts:

$$min_{w,b,\alpha} \quad \lambda||w - \sum_i^M \alpha_i u_i||^2 + \gamma\sum_i^M \alpha_i^2 + \sum_i^N \max\Big(0, 1 - y_i(w^\mathsf{T} x_i + b)\Big) \tag{5.13}$$

$$+\theta\left(\sum_{i,j}^{M,M} C^+_{ij}||\alpha_i - \alpha_j||^2 - \sum_{i,j}^{M,M} C^-_{ij}||\alpha_i + \alpha_j||^2\right) \qquad st: \alpha_i \geq 0, \qquad \forall i$$

5.2.2 Optimization by Feature Mapping

This section will discuss how to map the objective (5.13) to a "classical" SVM formulation by defining appropriate feature maps. Initially the formulation will be converted to a spec-

tral form, then by incorporation of some new variables, a few pseudo-training samples and appropriate rearrangements the objective will be transformed to a linear SVM objective.

Assuming that C_{ij} are the weights on a graph, $\phi(\alpha)$ in (5.9) can be re-written in the spectral form as:

$$\begin{aligned} \phi(\alpha) &= \sum_{i,j} C_{ij}||\alpha_i - \alpha_j||^2 = 2\alpha^\mathsf{T} L\alpha \\ & \quad where \quad L = D - C, \quad D_{ii} = \sum_j C_{ij}, \end{aligned} \tag{5.14}$$

where L is the graph laplacian. Similarly we can rewrite $\bar{\phi}(\alpha)$ in (5.11) as:

$$\begin{aligned} \bar{\phi}(\alpha) = \sum_{i,j} C_{ij}||\alpha_i - \alpha_j||^2 - 4\alpha^\mathsf{T} D^- \alpha = 2\alpha^\mathsf{T} \bar{L}\alpha \\ \bar{L} = \bar{D} - C, \quad \bar{D}_{ii} = \sum_j |C_{ij}| = D^+ - D^- \end{aligned} \tag{5.15}$$

Let $U = [u_1, u_2, .., u_M]$ and $w = \hat{w} + \sum_i^M \alpha_i u_i = \hat{w} + U\alpha$, rewriting (5.13) with appropriate substitutions:

$$\begin{aligned} min_{w,b,\alpha} \quad \lambda\hat{w}^\mathsf{T}\hat{w} + \alpha^\mathsf{T}(\gamma I + 2\theta\bar{L})\alpha + \sum_i^N \max\Big(0, 1 - y_i(\hat{w}^\mathsf{T} x_i + \alpha^\mathsf{T} U^\mathsf{T} x_i + b)\Big) \\ st : \alpha_i \geq 0, \quad \forall i \end{aligned} \tag{5.16}$$

Considering both $\bar{L}$ and I are positive-semi definite, we can decompose the term $\gamma I + 2\theta\bar{L}$

using Cholesky decomposition as RR^T then introduce $\beta = R^\mathsf{T}\alpha$ and rewrite:

$$\alpha^\mathsf{T}(\gamma I + \theta\bar{L})\alpha = \alpha^\mathsf{T} R R^\mathsf{T}\alpha = \beta^\mathsf{T}\beta \tag{5.17}$$

$$\alpha^\mathsf{T} = \beta^\mathsf{T} R^{-1} \tag{5.18}$$

Plugging in (5.17) and (5.18) to (5.16) we obtain:

$$\begin{aligned} & min_{\hat{w},b,\beta} \quad \lambda\hat{w}^\mathsf{T}\hat{w} + \beta^\mathsf{T}\beta + \sum_i^N \max\Big(0, 1 - y_i(\hat{w}^\mathsf{T}x_i + \beta^\mathsf{T}R^{-1}U^\mathsf{T}x_i + b)\Big) \\ & \qquad st: \Big[(R^{-1})^\mathsf{T}\beta\Big]_i \geq 0, \quad \forall i \\ = \; & min_{\bar{w},b} \quad ||\bar{w}||^2 + \sum_i^N \max\Big(0, 1 - y_i(\bar{w}^\mathsf{T}\bar{x}_i + b)\Big) \qquad st: \Big[(R^{-1})^\mathsf{T}\beta\Big]_i \geq 0, \quad \forall i \\ & where \qquad \bar{w} = [\sqrt{\lambda}\hat{w}; \beta], \quad \bar{x}_i = [\tfrac{1}{\sqrt{\lambda}}x_i; R^{-1}U^\mathsf{T}x_i], \end{aligned} \tag{5.19}$$

Other than the linear constraints $\big[(R^{-1})^\mathsf{T}\beta\big]_i \geq 0$, the objective becomes a linear SVM objective.

5.2.3 Enforcing linear constraints in SVMs

For any classical SVM formulation:

$$min_{w,b} \quad ||w||^2 + \sum_i^N \max\Big(0, 1 - y_i(w^\mathsf{T}x_i + b)\Big) \tag{5.20}$$

we can implement a linear constraint $a^{\mathsf{T}}w \geq 0$ by introducing an additional sample $x_{N+1} = [\infty \times a]$ with the label $y_{N+1} = +1$. Here $\infty \times a$ is the multiplication of the vector a with the infinity which is approximated with a very high numeric value (i.e. 10^6). Similarly for implementing the linear constraints in (5.19) we add M additional samples to our training set:

$$\bar{x}_{N+k} = [\mathbf{0}^{|\hat{w}|}; \infty \times (R^{-1})^{\mathsf{T}}_{k,:}], \quad y_{N+k} = +1, \quad 1 \leq k \leq M \tag{5.21}$$

After handling the constraints as well, the transformation of the original objective (5.13) to a classical SVM formulation is completed. Therefore we can optimize (5.13) using available robust SVM optimization tools.

5.3 Conclusion

In this chapter we introduced a convex potential function for incorporating the pairwise co-occurrence relations into convex max-margin learning frameworks. We also showed that by defining appropriate feature maps, many transfer learning formulations are transformed to a classical SVM formulation, and subsequently solved by much easier and more robust optimization tools developed throughout the past years.

Chapter 6

Fast and Scalabe Object Detection for Large Image Collections

Over the last decade there has been considerable progress on large scale object instance retrieval where the goal is to retrieve all images containing a specific object in a large scale image dataset, given a query image of that object (Jégou et al., 2008, 2010, Nister and Stewenius, 2006, Philbin et al., 2007, Sivic and Zisserman, 2003). These methods are now appearing in web and mobile applications such as Google Goggles, Kooaba, and Amazon's SnapTell. Similarly, large scale image classification, where the goal is to retrieve images containing an object category, has also seen improvements in both accuracy (Deng et al., 2012) and descriptors (Chatfield et al., 2011, Perronnin et al., 2010). With the advent of more efficient descriptor compression methods such as Product Quantization (Jégou et al., 2011) or binary codes (Raginsky and Lazebnik, 2009, Torralba et al., 2008, Weiss

et al., 2008) large scale image search can proceed quickly in memory without requiring (slow) disk access, for example by using efficient approximate nearest neighbor matching or binary operations respectively.

However, object *category detection*, where the goal is to determine if one or more instances of a category are present in an image with their corresponding locations, has not seen a similar development of immediate large scale retrieval. The complexity of detecting an object category is still linear in the number of images in the dataset. Despite methods of greatly speeding up sliding window search (over all possible scales and positions) on a per image basis (Dean et al., 2013, Dubout and Fleuret, 2012, Felzenszwalb et al., 2010b, Girshick et al., 2013, Kokkinos, 2011, Pedersoli et al., 2011, Song et al., 2012, Vedaldi and Zisserman, 2012), the cost of applying these methods at large scale is still extremely high. The objective of this chapter is to fill this gap by building upon the part-level transfer (i.e. classifier reconstruction) idea introduced in chapter 5.

One might consider image representations developed for object instance retrieval methods using the sparse BoW encoding. This encoding is often quite high dimensional ($100K$–$1M$) but sparse, and this allows the use of an inverted index (Nister and Stewenius, 2006, Sivic and Zisserman, 2003) for immediate retrieval, and subsequent localization of the object in each image. This has the advantages of being invariant to scale, illumination and viewpoint changes particularly for non-deformable objects with strong textures, but they can't tolerate deformation and intra-class generalization well. Therefore these representations are adequate for retrieving specific objects, however, they do not generalize well over deformation and intra-class variation. For example, consider searching for a cow with such

methods; instances of the same cow as the query image may be returned but instances of other cows in a similar viewpoint are not in general, because the visual words (and underlying SIFT descriptors) differ between instances. On the other hand, object category detection methods (Felzenszwalb et al., 2010a, Vedaldi et al., 2009) using a classifier template (trained on multiple instances) can perform generalization and localization.

To this end, given a HOG template classifier, we describe an approach that can *immediately* retrieve detections throughout a large scale dataset. Note, the goal is detection rather than image level classification and consequently we operate in a very large space of subwindow candidates. We are agnostic about the source of this template; it could be the component of a linear SVM based object category detection (Felzenszwalb et al., 2010b), or from an exemplar SVM (E-SVM) (Malisiewicz et al., 2011, Shrivastava et al., 2011), with the latter more suited to detecting object categories in a similar pose. We can also benefit from recent fast methods of training such classifier templates, such as LDA (Linear Discriminant Analysis) models (Hariharan et al., 2012).

Two main questions are investigated : (A) can an image representation be designed to enable fast and accurate large scale object category detection? and (B) can spatial reranking methods, similar to those used in object instance retrieval (Jégou et al., 2008, Philbin et al., 2007), be used to improve precision of the high ranked images?

6.1 Architecture Overview

We first briefly overview the architecture of the approach, and then describe the stages in more detail in the following sections. The method follows the framework established for the bag-of-visual-words (BoW) instance retrieval systems (Nister and Stewenius, 2006, Philbin et al., 2007, Sivic and Zisserman, 2003) with an offline processing stage (that can be computationally expensive), followed by an online search stage using an inverted index and posting-lists that is fast at run time and scalable. The online search involves two steps: an initial ranking of the images using BoW vectors alone; followed by reranking of a short list using spatial information.

The key ideas are (i) to build a vocabulary of discriminative classifier patches (CP). The HOG classifier template is approximated by the CPs. This is inspired by the recent development of mid-level features for discriminative object (Girshick et al., 2013, Song et al., 2012) and scene category recognition (Pandey and Lazebnik, 2011, Parizi et al., 2012, Singh et al., 2012); and (ii) to represent each image in the dataset by the *maximum response* to each of the CPs, i.e. if there are V CPs then the vector representing each image would be of dimension V if only the maximum response is stored (in fact, the location and scale of the maximum CP response are also stored). The CP vocabulary building, maximum response computation and sparsification, and indexing, are performed offline.

At run time, detection for a given HOG classifier template then proceeds in three stages (see figures 6.1 and 6.2):

1. Representing the HOG classifier template: The classifier is approximated by a

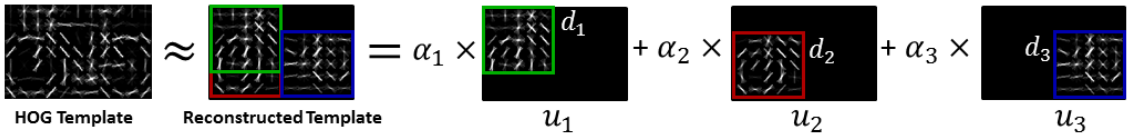


Figure 6.1: **Query representation.** The query HOG template w is approximated with the reconstructed template $w^{rt} = \sum_j \alpha_j u_j$ which is used as the intermediate query represention. Note, the representation u_j records both the classifier patch (CP) d_j used and also its position relative to the original HOG template.

set of the CPs.

2. Image retrieval: A 'posting list' is obtained for each CP used to represent the template, and those images occurring in the posting lists are ranked based on their aggregated maximum scores.

3. Spatial reranking of a short list: The top k images of the ranked list are reranked based on the spatial consistency of the maximum response CPs in that image. Spatial consistency is measured with respect to the rectangle of the original template. This stage is inspired by the voting based methods of (Chum and Zisserman, 2007, Marszalek and Schmid, 2006).

6.2 Approach

This section describes the scalable detection system. The two main aspects are the query representation and the image representation for the dataset to be searched. We describe how to obtain each of these representations and then their joint use for fast detection.

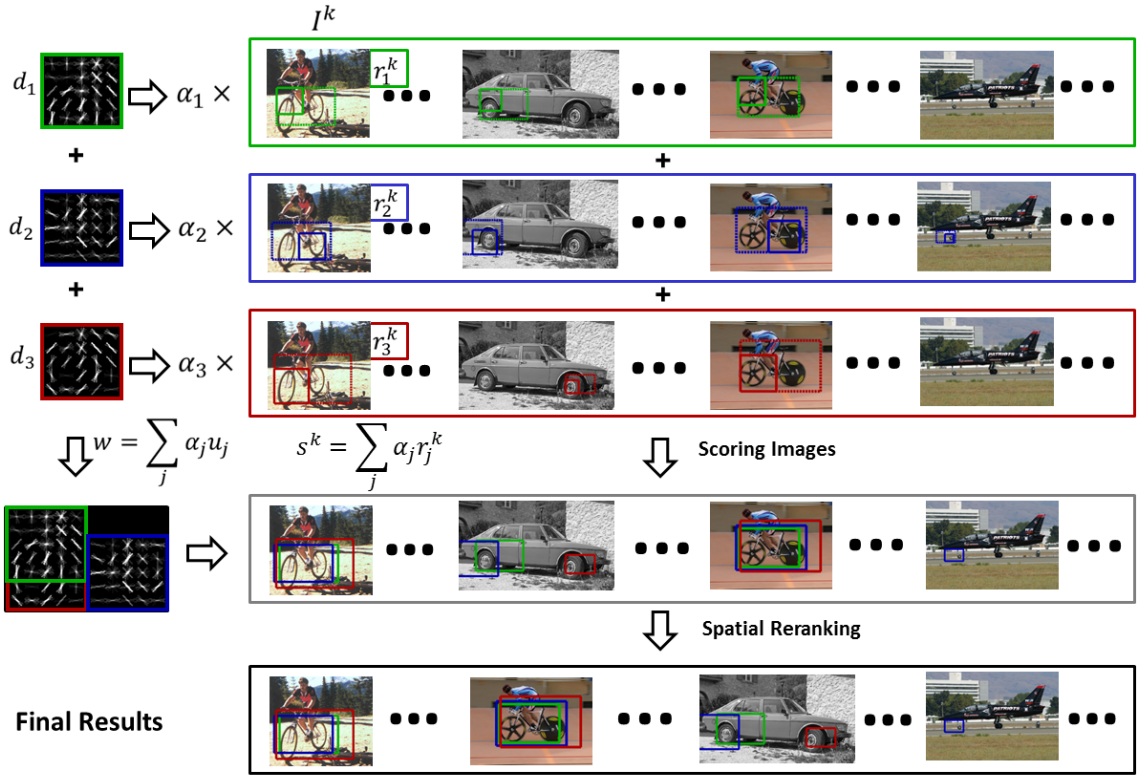


Figure 6.2: **Detection using a template composed of CPs.** Each CP retrieves a posting list of images, and a score for each image is computed from the weighted maximum response representation. Images are initially ranked on this score, and the top k then reranked using the spatial information associated with each maximum response by aggregating candidate bounding boxes.

6.2.1 Query Representation

The query is specified as a HOG classifier template which can be obtained using various methods (e.g. DPMs, E-SVMs, LDAs, etc.). The first step of the method is obtaining the intermediate query representation which is referred as the *reconstructed template (RT)*. This representation is obtained by approximating the query HOG template as another HOG template constructed using a sparse weighted combination of CPs. This procedure

is illustrated in figure 6.1. The CPs are essentially parts of classifiers that have earlier been trained in a discriminative manner using a DPM on another dataset. The intuition here is that there are naturally repeating classifier patches that can be shared across objects, such as parts of wheels, legs, corner-like patterns of doors and windows, etc. The method of generating the CP vocabulary is described in the implementation details of section 6.3.

Given the query HOG template w the learning of the reconstructed template w^{rt} can be performed in two different ways – a dense or sparse tiling. The two methods are illustrated in figure 6.3.

Dense tiling. The first method splits the query template into non-overlapping tiles and represents each tile with one of the CPs together with a reconstruction weight. The objective for this formulation is:

$$\min_{\alpha} \sum_{i} ||w_i - \sum_{j} \alpha_{ij} d_j|| \quad st : ||\alpha_i||_0 = 1 \quad \forall i \tag{6.1}$$

$$w_i^{rt} = \sum_{j} \alpha_{ij} d_j \tag{6.2}$$

where the w_i are the parts of the original template w, the w_i^{rt} are the parts of the reconstructed template w^{rt}, d_j are the CPs, and α_i performs the selection and weighting of the best CP that minimizes the objective. Note that each w_i^{rt} is reconstructed using a single CP from the vocabulary. Since each tile is independent of the other, the optimization reduces to individually choosing the best CP for each tile.

Sparse tiling. The second method of reconstruction does not densely tile the original template w, rather it uses a set of possibly overlapping CPs, so that the template may be only sparsely approximated. For the representation, each d_j is located on a w sized zero filled templates named u_j's, and the w^{rt} are then reconstructed as a sparse weighted combination of u_j's (see figure 6.1). In other words, u_j encodes the d_j with a specific spatial location x_j on the w sized template. Then the reconstructed template is learned as a weighted combinations of u_j's with the objective:

$$\min_{\alpha, x_j} ||w - \sum_j \alpha_j u_j|| + \gamma ||\alpha||_1 \quad st : \alpha_j \geq 0 \tag{6.3}$$

$$w^{rt} = \sum_j \alpha_j u_j \tag{6.4}$$

where scalar α_j is the combination weight of the relocated classifier patch d_j, and γ controls the sparsity of the reconstruction. Large values of γ encourage very sparse reconstructions, i.e. a small number of CPs are used. The aim is to choose the spatial location of the d_j's such that it best matches the corresponding location on w. In the implementation, instead of using all the u_j's in learning, optimization is performed over a smaller subset that is determined by a similarity thresholding operation.

Note that in both of the reconstruction methods we can write the RT in the form of $w^{rt} = \sum_j \alpha_j u_j$. In the rest of the paper this form of RT will be used.

Discussion. There are many other reconstruction methods that could be explored – for

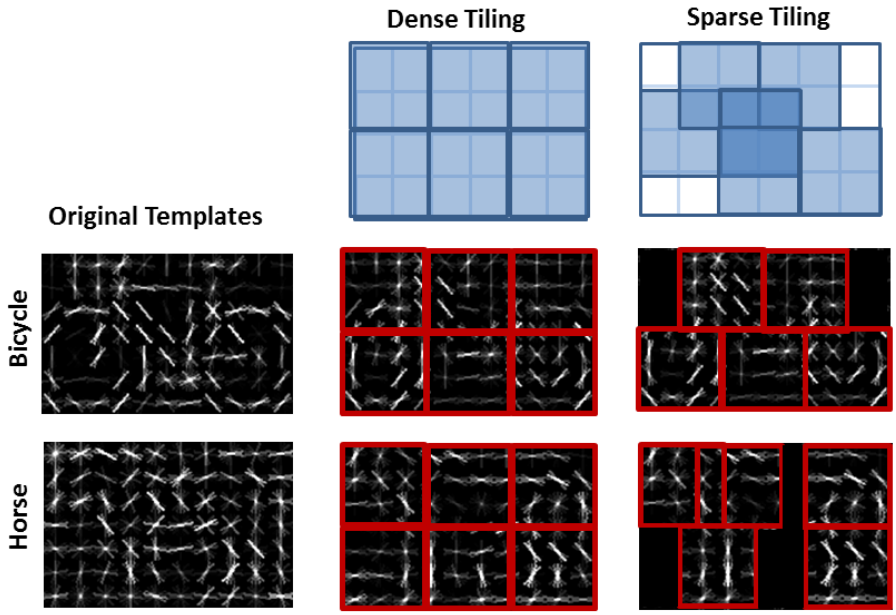


Figure 6.3: **Tiling methods for query reconstruction.**

example, sparse but non-overlapping, discriminative learning, and we discuss this further in section 6.5 after the use of this query representation has been explained.

Others have considered reconstruction of HOG templates (Girshick et al., 2013, Song et al., 2012). However, their main focus has been on fast evaluation of a set of templates on a given image, rather than on evaluating the given template over a large set of images as here. Girshick et al. (2013), Song et al. (2012) separated each HOG template into small non-overlapping parts and each part is constructed as a weighted combination of vocabulary items. Song et al. (2012) reconstructs already trained templates and learns the vocabulary for precise *reconstruction* rather than *discrimination*, whereas Girshick et al. (2013) learns both the vocabulary and the classifiers in a discriminative framework.

6.2.2 Image Representation

This section describes the image representation that is used for fast detection. The aim is to define such a representation that is suitable for obtaining a good upper bound on the score of w^{rt} on a given image. Each image in the dataset is represented as a V-dimensional vector, r, which contains the maximum scores of each CP evaluated on the image over all possible locations and scales. The location and the scale of the CP detection is also stored so that we can obtain a candidate bounding box as suggested by the specific CP. These candidates will be used later for the spatial reranking.

In detail, if the maximum response of any classifier c on the image I over all possible locations and scales is represented as $\Psi(c, I)$, then the image representation vector r is obtained as $r_j = \Psi(d_j, I) \quad \forall j$.

Note, that r_j is an *upper bound* on the score of w^{rt} on the image. Since the reconstructed template is $w^{rt} = \sum_j \alpha_j u_j$ where u_j's are d_j's with spatial relocation, the upper bound can be written as follows:

$$\sum_j \alpha_j \Psi(d_j, I) = \alpha^\mathsf{T} r \geq \Psi(w^{rt}, I) \tag{6.5}$$

Discussion. Since the maximum response of each CP can occur at any position and scale, there is no reason why the CPs should 'fire' in the relative positions assigned to them in the w^{rt} template. This is why r_j is an upper bound rather than equal to the maximum

score of w^{rt} on I, i.e. it ignores the spatial locations of the CPs detected on the image.

There are several generalizations possible for the image representation and we explore the first of these in the experiments. Instead of scoring only the maximum response of each CP, the top n responses can be stored. As will be seen, this will enable multiple instances of the category to be localized and also improves the localization of each. For example, suppose the image contains two cars, and the CP corresponds to a wheel, then if only the maximum response is stored then one instance can be recovered, but the location of the wheel might match the wrong wheel of that instance. In this ideal case at least two responses are required, one for each car, and four responses guarantees that there is a candidate BB for each of the true locations.

The second generalization is to introduce a spatial tiling and record the maximum responses for each tile. Note, the representation has some similarity to the max-pooling used for a visual BoW vocabulary (Mutch and Lowe, 2006), e.g. over a spatial pyramid. The principal difference is that here the vocabulary is composed of discriminative CPs, rather than SIFTs clustered using k-means.

Implicit shape model (ISM) by Leibe et al. (2004) also uses a similar representation, where each image is represented with image patches which are matched to the codebook entries, for obtaining a category specific segmentation mask for the given image. The main difference of our representation is that instead of using interest point detectors we use the responses of CPs.

6.2.3 Fast Detection

This section will describes how to perform immediate detection using the image and the query representations described in the previous two sections. Given a HOG template as the query the fast detection is performed in three stages: (a) query representation, (b) shortlist retrieval, (c) spatial reranking.

Query representation. As it is already discussed in section 6.2.1 the query HOG template is represented with the reconstructed template $w^{rt} = \sum_j \alpha_j u_j$ using one of the described methods.

Shortlist retrieval. At this stage a shortlist of images, potentially matching well with the query, are retrieved. The score of the image I^k for the query w^{rt} is computed as $s^k = \alpha^\mathsf{T} r^k$ where the reconstruction weights α are obtained from the query representation. As noted in section 6.2.2, this score is indeed an approximation of the maximum response of w^{rt} evaluated on the image I^k. Then the shortlist is compiled as the selection of top M images sorted by the score s^k. This stage is carried out very efficiently for large scale collections using the inverted file index (see below).

Spatial reranking. After obtaining the shortlist of candidate images, a reranking process is performed which aggregates the scores of candidate bounding boxes (BB) coming from each CP. This aggregation procedure is similar to the standard method of greedy non-maximal suppression (NMS) used in object detection. More precisely, if the overlap between two candidate BBs are more than 50% threshold, then the two bounding boxes are aggregated by taking the maximally scoring BB position and summing up the scores

coming from the two BBs. Starting with the maximum scoring BB, this procedure is repeated until there is no couple of overlapping BBs above 50% threshold. Note that if all the CPs fire with the same relative positioning as they are in the RT, in other words all the candidate BBs overlap 100%, then this procedure will give us the score of the RT exactly together with the right bounding box. Also note that specifying the overlap threshold as 50% rather than a larger value allows certain level of spatial deformation for the CPs.

Discussion. Similar to our spatial reranking method, Leibe et al. (2004) also performed a voting approach using generalized hough transforms for assigning class confidences to the pixels in the image, and it is particularly used for category specific segmentation.

6.3 Implementation Details

6.3.1 Construction of Classifier Patch Vocabulary

This vocabulary is composed of CPs obtained from previously trained classifiers. Similar to the vocabular used in chapter 5, DPM models are trained using 3 components (6 in total with mirrors) without parts over 1000 classes of the ImageNet 2012 challenge Deng et al. (2012) using the provided images. The models performing over 30% AP (329 models) are selected as the source for the CPs – we restrict to these as we wish to use a vocabulary of well trained discriminative patches.

To build the vocabulary, all patches with sizes of 3×3, 4×4, 5×5, 6×6, and 7×7 are extracted from the components of the trained DPMs. For each size, a k-means clustering

is performed with $k = 10$K centres. Finally from each cluster the most central patch is selected and is used for the vocabulary (i.e. we do not use the mean of the cluster as this tends to average out details). The outcome is five 10K vocabularies, one for each patch size.

Since the DPMs are trained discriminatively using a large number of training samples, the vocabulary items have a strong sense of discrimination, and since they are extracted from semantically meaningful objects, they possess certain semantic properties as well.

6.3.2 The inverted index and offline processing

For each image in the test dataset, the maximum response of each CP together with the corresponding locations and scale is computed and stored. If there are V CPs and N images, then this is equivalent to a $V \times N$ matrix M where each entry m_{ji} is the maximum response of the jth CP on the ith image. Since the approximated template is composed from a small number of CPs, the score for each image involves computing a weighted sum of a small number of rows of M. This is fine for the small scale regime where M easily fits into memory.

However, for (i) the large scale regime where M does not fit in memory, or (ii) cases where an image may be represented by the top set of responses (rather than only the maximum), or (iii) cases where the maximum response is thresholded so that M becomes sparse (as some images have no response for the thresholded CP); then posting lists for each CP held in an inverted file are the ideal solution. Each entry in the inverted index is

a posting list for the CP containing: the image identifier, responses to the particular CP, and location and scale of the associated templates.

In order to see the effect of sparsity in the experiments we limit the number of items in each posting list to either 1000 and 5000 for each CP. In addition to keeping the maximum response for each CP, we also experiment with the top 3 responses. This increases the possibility of overlap of BBs in the stage of spatial reranking, and enables the system to detect multiple instances of an object in a single image.

6.4 Experiments

The experiments are conducted on two image datasets: (1) the test set of the PASCAL Visual Object Classes 2007 challenge, which will be referred as VOC07; and (2) the validation set of the ImageNet Large Scale Visual Recognition Challenge 2012, will be referred to as ImageNet12. The number of test images in VOC07 is ∼5K, and in ImageNet12 it is ∼45K. The method is assessed on VOC07 alone, and then on VOC07+ImageNet12 for large scale detection, where the $45K$ images of ImageNet12 act essentially as distractors.

We evaluated the methods based on their object detection (find any instances of the object category in the image and localize them) performance. Here we follow the VOC practice and software (for example using an overlap threshold of 0.5 between prediction and ground truth for a true positive detection).

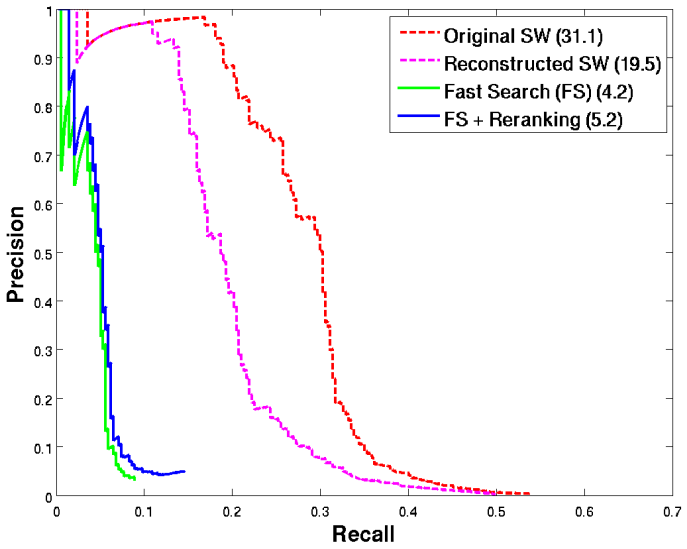
The evaluated methods are (i) sliding window evaluation of the original HOG template

(Original SW), (ii) sliding window evaluation of the RT (Reconstructed SW), (ii) fast search (FS) through the use of inverted index files (5000 entry per CP), (iii) sparse fast search (FS Sparse) where we use the inverted index files with 1000 entries per CP. For the fast search results, we also include the effect of spatial reranking separately. Note that, since the first two methods are using sliding window search, they are very slow and not applicable for fast detection. In all the experiments 5×5 CP size is used unless stated otherwise. In the shortlisting stage, the top 1000 images are retrieved for the fast search.

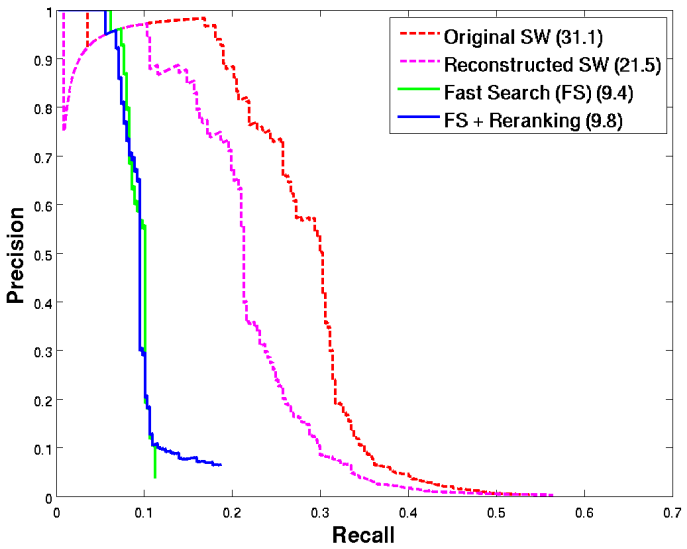
6.4.1 Evaluation on a single component

Initially we evaluate the framework with a single side-facing bicycle component which is obtained from a three component DPM model trained using VOC07 data. The RT is constructed using 5×5 vocabulary with both dense tiling and sparse tiling methods.

The detection results, displayed in table 6.1 and in figure 6.4, show that (a) the maximum response representation does indeed enable detection, and (b) spatial reranking improves precision at low rank. And this performance can be obtained immediately. The FS methods are approaching the performance of SW methods especially for the precision at low ranks (PR@10 and PR@50), though the same is not true for the AP. When we decrease the length of inverted index files from 5K (FS) to 1K (FS sparse), there is no noticeable decrease in the performance (the inverted index is about 80% sparse in this case). Keeping the top 3 maximum scores rather than one for each CP indeed improves the performance for both dense tiling and spatial reranking, since it enables us to detect multiple objects



(a) Dense Tiling



(b) Sparse Tiling

Figure 6.4: **Detection precision-recall curves.** Evaluation of the side-facing bicycle component on VOC07 test set. Note the high precision in the small recall values and the performance improvement after spatial reranking.

in a single image.

The significant timing difference between exhaustive sliding window search ($\sim$ 1.3 hours) and fast search ($\leq$ 1 second) is also displayed in table 6.1. Although sparse tiling performs better, as it uses more CPs during the construction, it increases the search time as well. Even though approaches such as cascade detection by Felzenszwalb et al. (2010b) ($\sim 20\times$ speedup) and application of product quantization by Vedaldi and Zisserman (2012) ($\sim 6\times$ speedup) also improves the detection speed, they are not comparable to the proposed method ($\sim 5000\times$ speedup) and their methods scale linearly with the number of images as opposed to near constant time performance of the fast search, though in terms of AP their results are better compared to the proposed method.

The framework in this paper is mainly designed for immediate search ($\leq 1s$). However, if we can spare more time, a possible improvement is reranking the short-list of images (or BBs) using the original classifier template. This requires the HOG descriptors of the images (pre-loaded to the memory) and so is slower than the spatial reranking proposed here. For instance, in the sparse tiling setting, the PR@50 increases from 64% to 84% and AP increases from 9.8% to 15% when we rerank top 1000 BBs using the original

6.4.2 Fast Detection using DPM Models

In the previous section the experiments are conducted on a single template. Here we discuss the evaluation of the fast search using the DPM models with multiple components (templates). The DPM models are trained over the training and validation set of VOC07

Methods	PR@10	PR@50	PR@100	AP	TIME
Original SW	100.0	98.0	77.0	31.1	$\sim$ 1.3 hrs
Reconstructed SW	90.0	92.0	58.0	19.5	$\sim$ 1.3 hrs
Fast Search (FS)	70.0	34.0	19.0	4.2	0.1 s
FS Sparse	70.0	34.0	19.0	4.1	0.1 s
FS + Top 3 Max	70.0	40.0	24.0	5.2	1.8 s
FS + Reranking	70.0	38.0	21.0	5.2	0.3 s
FS Sparse + Reranking	70.0	36.0	21.0	4.9	0.3 s
FS + Top 3 Max + Reranking	70.0	40.0	24.0	5.6	2.1 s
FS + Reranking + Orig. Rescoring	100.0	66.0	41.0	11.3	1.4 s

(a) Dense Tiling

Methods	PR@10	PR@50	PR@100	AP	TIME
Original SW	100.0	98.0	77.0	31.1	$\sim$ 1.3 hrs
Reconstructed SW	90.0	88.0	67.0	21.5	$\sim$ 1.3 hrs
Fast Search (FS)	100.0	60.0	34.0	9.4	0.6 s
FS Sparse	100.0	58.0	34.0	9.5	0.6 s
FS + Top 3 Max	100.0	66.0	41.0	11.5	10.4 s
FS + Reranking	100.0	64.0	32.0	9.8	1.1 s
FS Sparse + Reranking	100.0	62.0	32.0	9.6	1.0 s
FS + Top 3 Max + Reranking	100.0	70.0	41.0	12.7	11.7 s
FS + Reranking + Orig. Rescoring	90.0	84.0	52.0	15.0	2.0

(b) Sparse Tiling

Table 6.1: **Detection performance comparison of sliding window (SW) and fast search (FS) methods on VOC07 test set using the side-facing bicycle component.**

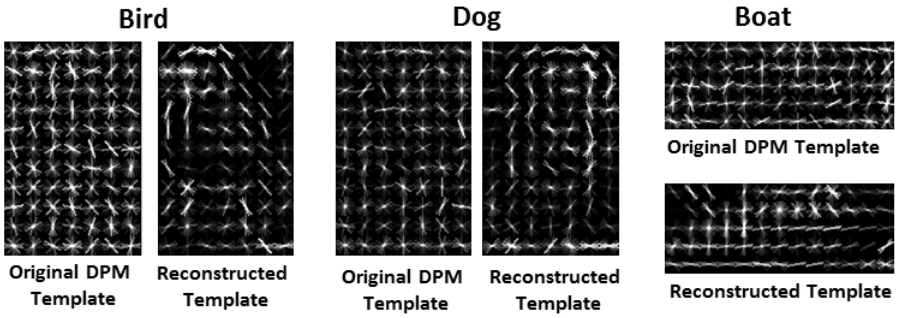


Figure 6.5: Reconstruction picks one of the views from the noisy superpositioned views in the DPM component.

using 3 components (6 with mirrors) excluding the parts. In total we have $20 \times 6 = 120$ HOG templates that are evaluated for fast detection and the mean performances for each class is reported in table 6.2.

Looking at the mean performances, especially PR@10 for each class, we observe that FS with spatial reranking approaches the quality of the exhaustively evaluated DPM models with the elapsed time roughly 1 second for each class (rather than $\sim$1.3 hours). On average spatial reranking significantly improves over fast search in all three performance measures.

When the intra-class variation is too high, this is especially true of animals, we observe a superpositioning of different views in the DPM components, since each view can not find a component to exist separately. We observed that the reconstruction picks one of the dominant views whichever is represented best with the CPs (see figure 6.5).

PR@10 Results	**aero.**	**bike**	**bird**	**boat**	**bottle**	**bus**	**car**	**cat**	**chair**	**cow**	**mean**
Original SW	55.0	93.3	28.3	28.3	53.3	81.7	100.0	21.7	71.7	60.0	**59.2**
Reconstructed SW	36.7	73.3	23.3	18.3	28.3	50.0	88.3	13.3	13.3	50.6	**35.9**
FS	25.0	58.3	3.3	10.0	13.3	31.7	78.3	3.3	3.3	35.4	**24.1**
FS+RR	31.7	61.7	8.3	16.7	25.0	48.3	90.0	1.7	10.4	50.8	**32.7**

PR@10 Results	**table**	**dog**	**horse**	**m.bike**	**person**	**plant**	**sheep**	**sofa**	**train**	**tv.**
Original SW	25.0	10.0	96.7	81.7	88.3	43.3	48.3	50.4	75.0	71.7
Reconstructed SW	8.3	5.0	48.3	46.7	70.0	15.0	40.0	5.0	36.7	46.7
FS	0.0	3.3	30.0	40.0	80.0	3.3	13.9	3.3	23.3	23.3
FS+RR	5.0	3.3	51.7	51.7	80.0	8.3	20.2	0.0	43.3	45.0

PR@50 Results	**aero.**	**bike**	**bird**	**boat**	**bottle**	**bus**	**car**	**cat**	**chair**	**cow**	**mean**
Original SW	32.8	70.3	11.0	16.3	34.0	48.7	96.3	11.0	38.8	35.0	**39.3**
Reconstructed SW	16.7	54.7	9.0	10.7	19.7	23.7	79.3	7.3	8.7	26.3	**23.7**
FS	8.0	39.0	2.7	4.0	8.7	11.3	50.0	1.0	3.4	14.6	**13.1**
FS+RR	11.3	47.3	4.3	6.7	12.7	22.7	65.3	1.0	6.7	20.0	**17.7**

PR@50 Results	**table**	**dog**	**horse**	**m.bike**	**person**	**plant**	**sheep**	**sofa**	**train**	**tv.**
Original SW	16.2	6.3	65.3	53.0	82.7	21.7	23.1	23.4	52.0	48.3
Reconstructed SW	3.7	6.7	26.3	32.3	61.9	5.0	20.3	3.3	23.0	35.3
FS	2.0	0.7	12.4	13.3	66.3	2.0	4.0	0.7	8.0	10.5
FS+RR	4.0	1.7	20.3	20.3	64.0	3.0	7.4	0.0	15.7	18.8

PR@100 Results	**aero.**	**bike**	**bird**	**boat**	**bottle**	**bus**	**car**	**cat**	**chair**	**cow**	**mean**
Original SW	22.4	51.0	8.0	11.1	24.8	31.0	88.7	8.2	26.2	22.1	**29.4**
Reconstructed SW	11.7	39.7	6.2	8.3	15.5	16.3	67.7	5.3	6.5	18.0	**18.3**
FS	4.7	23.3	1.7	2.5	5.7	6.5	33.7	1.0	2.0	9.2	**9.1**
FS+RR	6.7	31.5	2.2	4.2	8.0	12.8	46.0	1.0	4.2	13.3	**12.1**

PR@100 Results	**table**	**dog**	**horse**	**m.bike**	**person**	**plant**	**sheep**	**sofa**	**train**	**tv.**
Original SW	12.3	6.3	45.1	35.2	78.7	14.5	15.0	17.0	31.3	39.5
Reconstructed SW	3.0	6.3	18.5	21.3	58.1	3.8	13.0	2.8	15.3	29.4
FS	1.5	0.8	7.3	8.8	57.8	1.2	2.7	0.3	4.3	6.4
FS+RR	3.2	1.3	11.2	11.8	57.8	1.8	4.7	0.0	9.0	11.9

Table 6.2: Performances of the complete DPM models (3 components) on the detection evaluation over VOC07 test set with the sparse tiling setting. Note that fast search nearly reaches the performance of sliding window search using reconstructed template for PR@10 results with significantly small search time ($\sim$ 1.3 hours vs. $\sim$ 1 second)

PR@10 Results	**3x3**	**4x4**	**5x5**	**6x6**	**7x7**
Reconstructed SW	44.8	41.7	35.9	34.6	37.2
Fast Search	12.8	21.7	24.1	29.8	18.6
FS + Reranking	19.2	28.1	32.7	36.7	18.2

PR@50 Results	**3x3**	**4x4**	**5x5**	**6x6**	**7x7**
Reconstructed SW	27.5	26.9	23.7	23.0	23.6
Fast Search	5.7	10.8	13.1	14.4	10.9
FS + Reranking	9.1	14.2	17.7	18.5	11.3

PR@100 Results	**3x3**	**4x4**	**5x5**	**6x6**	**7x7**
Reconstructed SW	20.9	20.5	18.3	17.2	18.2
Fast Search	3.7	7.2	9.1	9.5	7.6
FS + Reranking	6.1	9.8	12.1	12.6	8.1

Table 6.3: **Performance as a function of CP size.** As CP size increases, the performance of fast search gradually increases as well reaching the limit at 6×6. Sparse tiling is used for reconstruction.

	3x3	**4x4**	**5x5**	**6x6**	**7x7**
Reconstructed SW	$\sim$ 1.3 hours	$\sim$ 1.3 hours	$\sim$ 1.3 hours	$\sim$1.3 hours	$\sim$1.3 hours
Fast Search	2.5 s	1.2 s	0.8 s	0.6 s	0.6 s
FS + Reranking	5.5 s	2.5 s	1.5 s	1.2 s	1.1 s
Avg. number of used CPs	132.2	63.8	41.4	32.6	29.0

Table 6.4: **Average elapsed time for each CP size.** As CP size increases, the number of CPs used for reconstruction decreases, hence the retrieval time decreases as well. Sparse tiling is used for reconstruction.

6.4.3 Effect of CP Size

The effect of CP sizes on the performance is also evaluated and reported in figure 6.6 and table 6.3. The reported results are mean detection performances for each component of all 20 classes of VOC07. While increasing the CP size from 3×3 to 7×7, expectedly, the quality of reconstruction decreases and this is reflected on the reconstructed template (SW) results. However, as CP size increases, the CPs become more semantically meaningful and definitive, and this improves the fast search results. Hence, while increasing the CP size, performances of the reconstructed template (SW) and fast search approaches each other and even meets for the PR@10 results at the size of 6×6. In the limit, this increase in CP size would reach to the size of the HOG templates, where we may not find a good match due to becoming too specific. We observe this behaviour at the level of 7×7 CPs where the performance of fast search drastically decreases.

The retrieval timings for each CP size is given in the table 6.4. As CP size increases, the speed of fast retrieval increases as well. The reason behind it is that while reconstructing the template using small CPs (e.g. 3×3) we need more items in construction as opposed to a larger CP size (e.g. 6×6). Using more CPs in reconstruction requires reaching a larger set of posting lists (i.e. one for each used CP) that are used for retrieval and spatial reranking. Therefore, as CP size increases the number of used CPs decreases which entails shorter retrieval times.

		PR@10	PR@50	PR@100	TIME
VOC07 (5K)	**FS**	70.0	34.0	19.0	123 ms
	FS+RR	70.0	38.0	21.0	238 ms
Imagenet12 + VOC07 (50K)	**FS**	70.0	28.0	15.0	126 ms
	FS+RR	60.0	28.0	16.0	273 ms
	FS+RR(2.5K)	60.0	34.0	22.0	678 ms
	FS+RR(5K)	60.0	40.0	23.0	1355 ms

Table 6.5: **Detection performance of the side-facing bicycle component evaluated in large-scale (∼50K).** Neither the timings nor the performances are affected by the additional ∼45K distractors. Dense tiling is used for reconstruction.

6.4.4 Large scale detection

In this experiment we extended the VOC07 test set by adding distractors from Imagenet12 validation set. In the combined test set we have ∼50K images. Since we don't have reliable ground truth for the Imagenet12 set, we performed manual evaluation.

We evaluated the side-facing bicycle component on VOC07 alone and the combined set. The results are shown in the table 6.5 and in the figure 6.7. We observed that the quality of the detections at low ranks is not affected by the distractors coming from Imagenet12, and the timings are similar even though we move from a 5K set to a 50K set, i.e. scaling up to large databases does not affect the speed or accuracy of the system – our original objective.

Note that the tandem images (see figure 6.7), 2 of them ranked in the top 10, and 6 of them ranked in the top 100, are not counted as bicycles and marked as false retrievals. If we count them as bicycles as well, the performance would be much higher.

6.4.5 Fast Detection using E-SVMs

This section discusses the evaluation of the fast search using the E-SVM models. For the quantitative results, E-SVMs are trained for randomly selected 50 BBs for each of the 20 classes using the training set of VOC07. The truncated and the difficult BBs are not used for training since they are not good representatives of the class. Each E-SVM model is evaluated individually for the category that it belongs to.

Many of the original E-SVMs (*using sliding window search*) did not return any positive images in the top 10 retrieval. Therefore, for a more sound evaluation, we used 192 of the queries where either the original ESVM or the reconstructed one retrieves at least 3 instances of the target class at top 10 using the sliding window search. Table 6.6 illustrates the mean performances of the evaluated E-SVM queries where we can observe quite good performances in the low rank results. Furthermore, a final reranking using the original E-SVM detector is also investigated. After spatial reranking, the top 1000 detections are rescored using original E-SVM detectors. Note that for this rescoring method we need to access the HOG features of the images which are pre-loaded to the memory in advance for fast access.

As a related baseline, we also compared our method with the Video Google (Sivic and Zisserman, 2003) approach with two different vocabulary sizes, $10K$ and $200K$. Video Google using $10K$ performed better than $200K$ since the latter is mostly suitable for object instance search hence can't generalize as well as $10K$ vocabulary. In summary, fast search methods perform significantly better than Video Google results with similar

	PR@10	PR@50	PR@100	TIME
Original SW	46.8	24.9	17.0	$\sim$ 1.3 hrs
Reconstructed SW	31.7	21.1	15.8	$\sim$ 1.3 hrs
Fast Search (FS)	15.4	8.7	6.4	0.2 s
FS + Top 3 Max	15.4	8.9	6.7	1.7 s
FS + Reranking	15.5	8.7	6.5	0.3 s
FS + Top 3 Max + Reranking	15.5	9.0	6.8	2.0 s
FS + Reranking + E-SVM Rescoring	24.5	13.0	9.0	1.1 s
Video Google 10K	8.8	4.5	3.0	0.1 s
Video Google 200K	4.0	0.8	0.4	0.1 s

Table 6.6: **Mean performance comparison of sliding window (SW) and fast seach (FS) methods on VOC07 test set using 192 E-SVMs trained from randomly selected samples from the 20 classes of VOC07 training set.** Dense tiling is used for reconstruction. Note the tremendous timing difference between sliding window methods (which employs exhaustive search) and fast search methods.

timings.

We also present some qualitative search results in figure 6.8 which are also retrieved immediately.

6.5 Summary and discussion

Traditionally object category detection is performed in a sliding window fashion, which is a very costly procedure for large scale datasets. We have presented a fast detection framework which obtains near state of the art low rank results. It is now possible, by learning the HOG classifier using an E-SVM or LDA (see figure 6.8), to go from specifying a visual query in an image, to retrieving object category detections over a large scale dataset in a just a matter of seconds.

Note, that our method is also applicable in conjunction with methods that first filter the images using image classification to obtain a short list. These methods can also employ efficient indexing strategies, e.g. Rastegari et al. (2011). Our method can then be used to immediately return object detections on the short list (as has been done in this paper for a short list generated by the maximum response representation).

This method opens up new research questions in the area of immediate object category retrieval. For example, is there a benefit in learning the approximating template discriminatively as is done in Girshick et al. (2013)? Is the sparse template representation improved by non-overlapping constraints in the learning? Would a maximum response representation extended to include a spatial pyramid layout of each image improve precision of the short list?

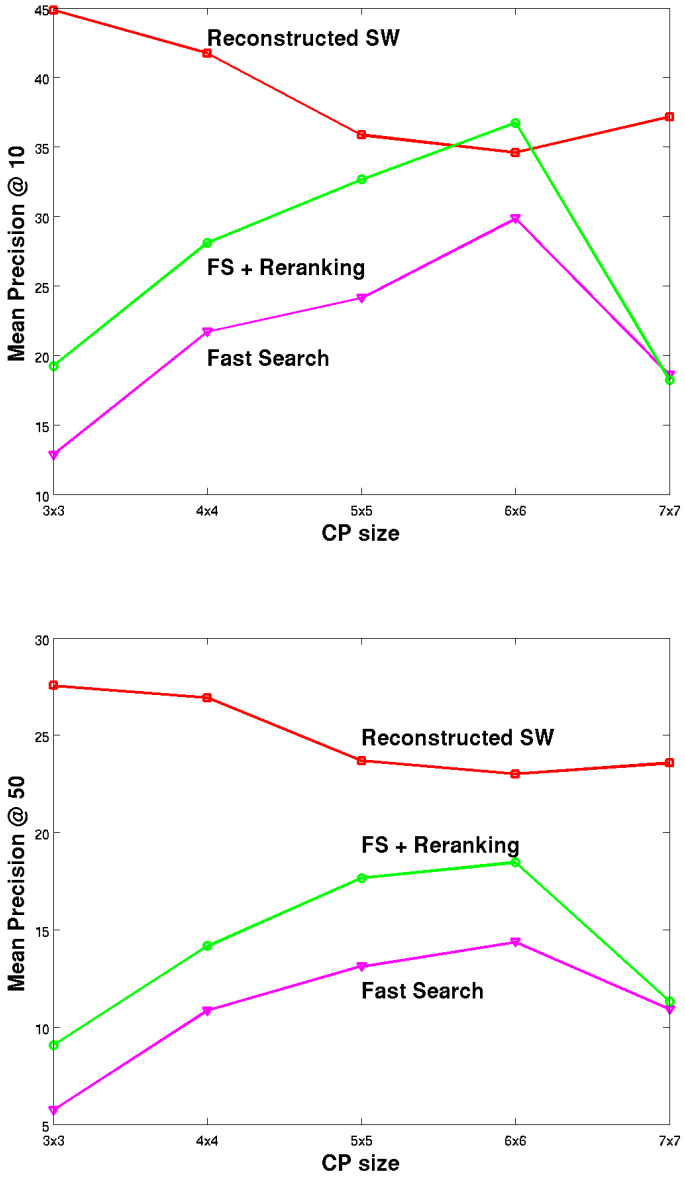


Figure 6.6: **Mean Precision results as a function of CP size for detection.** Scores are averaged over all 20 classes of VOC07. Sparse tiling is used for reconstruction.

Figure 6.7: Comparison of top 6 detections of the side-facing bicycle component over 5K and 50K test sets. Dense tiling is used for reconstruction.

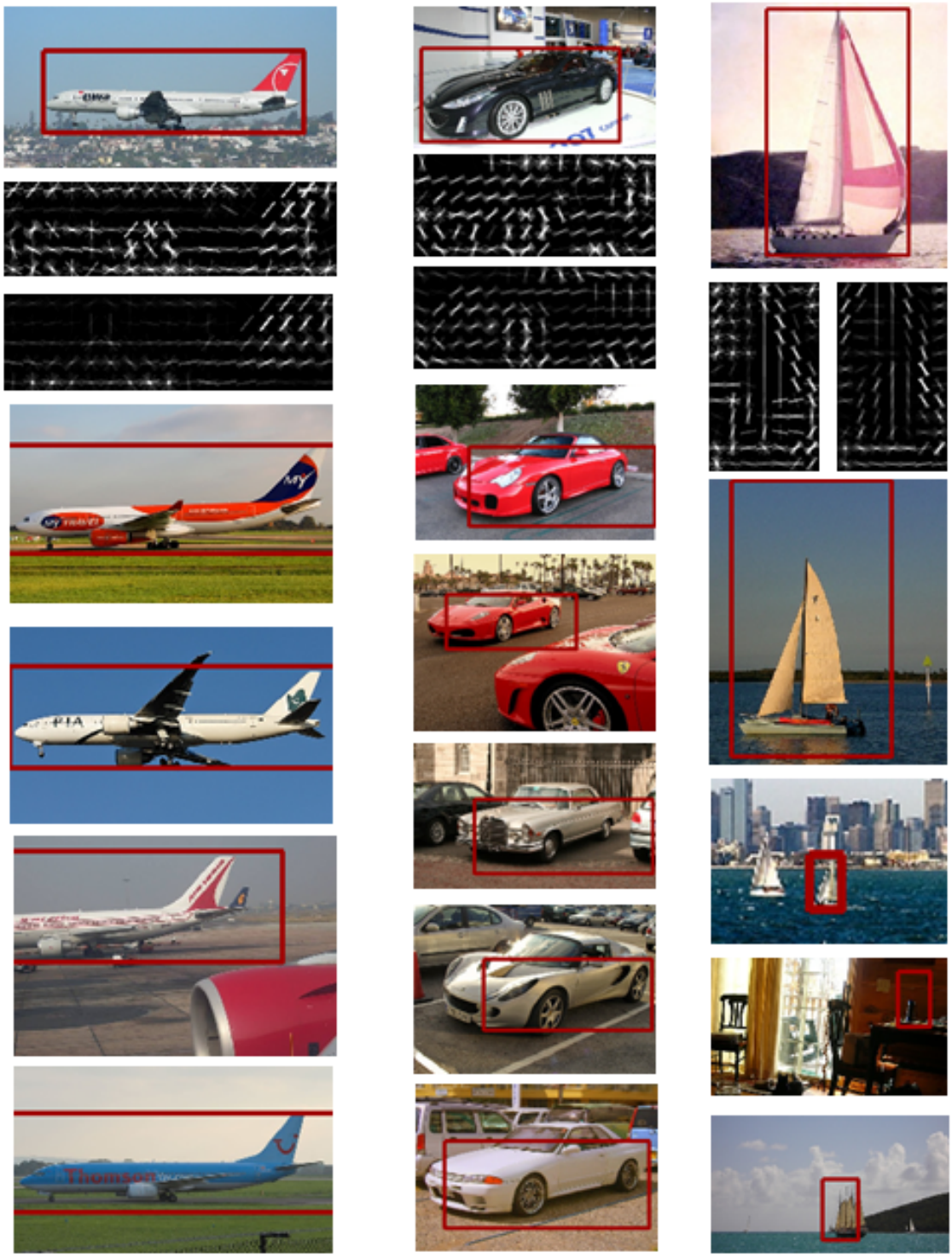


Figure 6.8: **Visualization of detection results for a few E-SVM queries.** Note the pose and the type similarity between the query and the retrieved results. These results are retrieved in less than a second.

Chapter 7

Conclusion

In this chapter we summarize the achievements of the presented thesis and discuss directions of future research.

7.1 Achievements

The overall achievement of this thesis is to develop and apply transfer learning methodologies for object category detection. The proposed methods are mainly demonstrated using HOG templates (Linear SVMs over HOG features (Dalal and Triggs, 2005) or DPMs (Felzenszwalb et al., 2008)) which are the state of the art models for object category detection and have been for the last five years. Here we present a list of accomplishments.

Rigid transfer, i.e. using the source detector as it is, is particularly proposed for the cases where the source and target detection tasks are very similar. For instance, transferring

from a motorbike detector in order to learn a bicycle detector, or transferring from horse to donkey. Although it is quite successful, it has a narrow band of applications since it is hard to find a very similar class to transfer from for many cases. Two main methods are proposed: the first one is the application of Adaptive SVMs (Yang et al., 2007b) for object detection, and the second one is a new objective named as Projective Model Transfer SVM.

Deformable transfer, i.e. transforming the source detector during transfer, is introduced for the cases where small local transformations will better benefit the source detector. For example, while transferring from a motorbike detector to a bicycle detector it would be beneficial to enlarge the motorbike wheels during the transfer to better fit the bicycle class hence providing a more successful transfer. The proposed formulation, Deformable Adaptive SVM, has a convex objective and performs a transformation of the source for a better match with the target during the transfer.

Part level transfer is particularly appealing when transferring from a single class is not possible, either due to a mismatch in the size or aspect ratio of the classifier templates or when no similar class exists. For instance, while training a detector for an ambulance the source repository might not include any similar size vehicle, however, by transferring parts from trucks, vans and cars, a successful transfer can be accomplished. This type of transfer can also be visualized as constructing a pseudo-detector template (using the parts extracted from source detectors) to use as a prior model for regularizing the target detector. This method lies somewhere between rigid and deformable transfer since parts are transferred without any local deformation but they can be relocated in the target detector. Being motivated by the existing semantic relations between the parts (e.g. we

expect to see a front leg when a tail or a back leg part exist) we also proposed to transfer some meta-level information such as the co-occurrence statistics between the parts. In that respect, a convex method for enforcing co-occurrence statistics to the SVM objective is introduced.

Efficient and robust optimization for transfer learning objectives is introduced. Transfer learning formulations are transformed to a "classical" linear SVM objective in order to utilize the robust tools developed throughout the years for optimizing SVMs. The semantic implications of the proposed equivalence relations provided a better insight into transfer regularization and the use of feature augmentation.

One-shot generalization, i.e. improving Exemplar SVMs (Malisiewicz et al., 2011), is better performed with the help of part level transfer learning. It has been shown that exemplar svms can be improved for better detection and retrieval of the same class instances in a similar pose. Particularly, part level transfer is very well suited for training stronger detectors for unusual poses and compositions of objects.

Fast and scalable detection is proposed by building upon the part based transfer (classifier reconstruction) idea. A part based representation for both the object detectors and test images is introduced, and through the use of inverted index files immediate (~ 1 second) detection in large image collections is performed. This method is applied to both running object category detectors and Exemplar SVMs, obtaining very good detection results in the low recall regions approaching the quality of sliding window evaluations (exhaustive search) with a considerable decrease in the search time (~ 1.3 hours vs. ~ 1

second).

7.2 Future Research

In this section we discuss some potentially promising future research directions.

Answering the "where" question. One of the main obstacles preventing us from developing solutions that address the "where" question, i.e. deciding which materials from which classes will be transferred, was the lack of evaluation of the methods due to the small number of classes existing in the current object category detection datasets (e.g. PASCAL VOC has 20 classes). However, the release of large scale datasets and challenges for object category detection, such as the ImageNet detection challenge (ILSVRC2013), makes it possible to search for the answer of the "where" question. Several potential sources of information can be used for this purpose. For instance, semantic relations, such as hypernyms-hyponyms(i.e. "is a" and "kind of" relations) and meronyms (i.e. "part of" relations), can be utilized for finding visually similar classes, particularly within the categories of vehicles and animals. Biological taxonomies, and perhaps even the genomes of animals, can be utilized for identifying visual similarity relations. Considering that semantics do not always necessarily coincide with visual similarities, the similarity connections can be empirically validated through the use of available training data and learned detector models. Supervised or unsupervised mid-level representations (i.e. representing images and classes with attributes or common generic parts) can also play a significant role for establishing similarities and hence determining where to transfer from.

Instance-based transfer. Most of the methods described in this thesis were focused on transferring information between categories. However we can also transfer information between instances. Instance-based transfer is loosely introduced by Malisiewicz et al. (2011) in the exemplar SVMs work, where the aim is to create similarity connections between instances using the available visual information and then selectively transfer some other properties (e.g. segmentation masks, 3D geometry, semantic labels, etc.) that exist in the source instance but not in the target one. For instance, the scene completion work by Hays and Efros (2007) can be thought of as instance-based transfer since the similarity connection is established using the general visual context of the image, and then the unwanted part of the target image is overwritten by transferring the similar part from the retrieved image. After establishing similarity connections via semantic hierarchical relations in ImageNet, Guillaumin and Ferrari (2012) transferred bounding box locations from one image to the other. Tighe and Lazebnik (2013) also transferred the segmentation masks from a database of objects in order improve pixel level labelling of the scenes. Wang et al. (2013) identified the similarity connection by matching 3D point clouds and transfers the semantic labels. The release of RGB-D datasets, such as (Lai et al., 2011), and other multi-model datasets, create the opportunity to establish similarities by using one modality (visual appearance (RGB)) and transfer information for annotating the instance for the other modality (depth information in the form of 3D point clouds). Similarly annotations in the form of bounding boxes, finer part-level annotations, semantic labels can all be interpreted as other modalities. With the introduction of our fast detection method discussed in chapter 6, well registered similarity connections between a

target instance and potential source instances can be achieved much faster even using a large scale image repository. This enables many other potential applications which were not possible before (e.g. object impainting, data driven object sketching, etc.) due to the search complexity.

Bibliography

Y. Amit, M. Fink, N. Srebro, and S. Ullman. Uncovering shared structures in multiclass classification. In *ICML*, pages 17–24, 2007.

R. K. Ando and T. Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853, 2005.

A. Argyriou, T. Evgeniou, and M. Pontil. Multi-task feature learning. In *NIPS*, pages 41–48, 2006.

A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

Y. Aytar, M. Shah, and J.B. Luo. Utilizing semantic word similarity measures for video retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

I. Biederman. *Recognition by components: A theory of human image understanding.* Psychological Review, 1987.

C. Bishop. *Pattern Recognition and Machine Learning.* Springer, New York, 2006.

R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

C. C. Chang and C. J. Lin. *LIBSVM: a library for support vector machines*, 2001. URL `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2011.

K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *Proceedings of the British Machine Vision Conference*, 2011.

O. Chum and A. Zisserman. An exemplar model for learning object classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis*, 2007.

W. Dai, Q. Yang, G. Xue, and Y. Yu. Boosting for transfer learning. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 193–200, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-793-3. doi: http://doi.acm.org/10.1145/1273496.1273521.

W. Dai, Q. Yang, G. Xue, and Y. Yu. Self-taught clustering. In *ICML '08: Proceedings of the 25th international conference on Machine learning*, pages 200–207, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4. doi: http://doi.acm.org/10.1145/1390156.1390182.

N. Dalal and B Triggs. Histogram of Oriented Gradients for Human Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

H. Daumé III. Frustratingly easy domain adaptation. *CoRR*, 2009.

T. Dean, M. Ruzon, M. Segal, J. Shlens, S. Vijayanarasimhan, and J. Yagnik. Fast, accurate detection of 100,000 object classes on a single machine. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

J. Deng, A. Berg, S. Satheesh, H. Su, and F. F. Khosla, A. Li. Imagenet large scale visual recognition challenge 2012 (ilsvrc2012), 2012.

C. Desai, D. Ramanan, and C. C. Fowlkes. Discriminative models for multi-class object layout. *International Journal of Computer Vision*, 95(1):1–12, 2011.

J. Donahue, J. Hoffman, E. Rodner, K. Saenko, and T. Darrell. Semi-supervised domain adaptation with instance constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

M. Douze, A. Ramisa, and C. Schmid. Combining attributes and fisher vectors for efficient image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

L. Duan, I.W. Tsang, D. Xu, and S.J. Maybank. Domain transfer svm for video concept detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

L.X. Duan, D. Xu, I.W. Tsang, and J.B. Luo. Visual event recognition in videos by learning from web data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

C. Dubout and F. Fleuret. Exact acceleration of linear object detectors. In *Proceedings of the European Conference on Computer Vision*, 2012.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html, 2007.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *International Journal of Computer Vision*, 88 (2):303–338, 2010a.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 2010b.

T. Evgeniou and M. Pontil. Regularized multi–task learning. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data*

mining, pages 109–117, New York, NY, USA, 2004. ACM. ISBN 1-58113-888-1. doi: http://doi.acm.org/10.1145/1014052.1014067.

R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.

P. Felzenszwalb, D. Mcallester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

P. F. Felzenszwalb, R. B. Grishick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010a.

P.F. Felzenszwalb, R.B. Girshick, and D. McAllester. Cascade object detection with deformable part models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2241–2248, 2010b.

S. Fidler, M. Boben, and A. Leonardis. Evaluating multi-class learning strategies in a generative hierarchical framework for object detection. In *NIPS*, 2009.

Michael Fink. Object classification from a single example utilizing class relevance metrics. In *Advances in Neural Information Processing Systems*, 2004.

FLICKR. http://www.flickr.com/.

T. Gao, M. Stark, and D. Koller. What makes a good detector? structured priors for learning from few examples. In *Proceedings of the European Conference on Computer Vision*, 2012.

R. Girshick, H.O. Song, and T. Darrell. Discriminatively activated sparselets. In *Proceedings of the International Conference on Machine Learning*, 2013.

G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007. URL `http://authors.library.caltech.edu/7694`.

M. Guillaumin and V. Ferrari. Large-scale knowledge transfer for object localization in imagenet. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

M. Hansen. *3D face recognition using photometric stereo*. PhD thesis, University of the West of England, 2012.

Z. Harchaoui, M. Douze, M. Paulin, M. Dudik, and J. Malick. Large-scale image classification with trace-norm regularization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

B. Hariharan, J. Malik, and D. Ramanan. Discriminative decorrelation for clustering and classification. In *Proceedings of the European Conference on Computer Vision*, 2012.

R. Haskell. *Transfer of learning : cognition, instruction, and reasoning.* Academic Press, San Diego Calif., 2001.

J. Hays and A. G. Efros. Scene completion using millions of photographs. In *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, 2007.

S. Helfenstein. *Transfer: review, reconstruction, and resolution.* PhD thesis, University of Jyvskyl, 2005.

M. Jain, H. Jégou, and P. Gros. Asymmetric hamming embedding. In *ACM Multimedia*, 2011.

H. Jégou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Proceedings of the European Conference on Computer Vision*, 2008.

H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

H. Jégou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

J. Jiang and C. Zhai. Instance weighting for domain adaptation in nlp. In *Annual Meeting of the Association for Computational Linguistics (ACL'07)*, 2007.

T. Joachims. Making large-scale svm learning practical. *Advances in Kernel Methods - Support Vector Learning*, 1999.

Z. Kang, K. Grauman, and F. Sha. Learning with whom to share in multi-task feature learning. In *ICML*, pages 521–528, 2011.

I. Kokkinos. Rapid deformable object detection using dual-tree branch-and-bound. In *Advances in Neural Information Processing Systems*, 2011.

B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, 2011.

A. Kumar and H. Daumé III. Learning task grouping and overlap in multi-task learning. In *ICML*, 2012.

M. P. Kumar, P. H. S. Torr, and A. Zisserman. Efficient discriminative learning of parts-based models. In *Proceedings of the International Conference on Computer Vision*, 2009.

I. Kuzborskij, F. Orabona, and B. Caputo. From n to n+1: Multiclass transfer incremental learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Graph cut based inference with co-occurrence statistics. In *Proceedings of the European Conference on Computer Vision*, 2010.

K. Lai, L. Bo, X. Ren, and D. Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Proceedings of the International Conference on Robotics and Automation*, 2011.

C. H. Lampert. Detecting objects in large image collections and videos by efficient subimage retrieval. In *Proceedings of the International Conference on Computer Vision*, pages 987–994, 2009.

C. H. Lampert and M. B. Blaschko. Structured prediction by joint kernel support estimation. *Machine Learning*, 2009.

S. Lee, V. Chatalbashev, D. Vickrey, and D. Koller. Learning a meta-level prior for feature relevance from multiple related tasks. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 489–496, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-793-3. doi: http://doi.acm.org/10.1145/1273496.1273558.

B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *Workshop on Statistical Learning in Computer Vision, ECCV*, May 2004.

L. Li, H. Su, Y. Lim, and F. F. Li. Objects as attributes for scene classification. In *ECCV Workshops (1)*, 2010a.

L. Li, H. Su, E. P. Xing, and F. F. Li. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *Advances in Neural Information Processing Systems*, 2010b.

X. Li. *Regularized Adaptation: Theory, Algorithms and Applications*. PhD thesis, University of Washington, USA, 2007.

J. J. Lim, R. Salakhutdinov, and A. Torralba. Transfer learning by borrowing examples for multiclass object detection. In *Advances in Neural Information Processing Systems*, 2011.

J. Liu, J. Sun, and H. Shum. Paint selection. In *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, 2009.

J. Luo, T. Tommasi, and B. Caputo. Multiclass transfer learning from unconstrained priors. In *Proceedings of the International Conference on Computer Vision*, 2011.

T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-SVMs for object detection and beyond. In *Proceedings of the International Conference on Computer Vision*, 2011.

M. Marszalek and C. Schmid. Spatial weighting for bag-of-features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

Z. Marx, M. Rosenstein, T. Dietterich, and L. P. Kaelbling. Transfer learning with an ensemble of background tasks. In *NIPS Workshop on Inductive Transfer*, 2005.

Mosek: Optimization Toolkit. http://www.mosek.com, 2010.

J. Mutch and D. G. Lowe. Multiclass object recognition with sparse, localized features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

G. Obozinski, B. Taskar, and M. I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252, 2010.

E.S. Olivas, J.D.M. Guerrero, M.M. Sober, J.R.M. Benedito, and A.J.S. Lopez. *Handbook Of Research On Machine Learning Applications and Trends: Algorithms, Methods and Techniques*. 2009.

A. Opelt, A. Pinz, and A. Zisserman. A boundary-fragment-model for object detection. In *Proceedings of the 9th European Conference on Computer Vision, Graz, Austria*, 2006.

F. Orabona, C. Castellini, B. Caputo, A.E. Fiorilla, and G. Sandini. Model adaptation with least-squares svm for adaptive hand prosthetics. In *Proceedings of the International Conference on Robotics and Automation*, 2009.

P. Ott and M. Everingham. Shared parts for deformable part-based models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

S.J. Pan and Q. Yang. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22(10):1345 –1359, oct. 2010. ISSN 1041-4347. doi: 10.1109/TKDE.2009.191.

M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization

with deformable part-based models. In *Proceedings of the International Conference on Computer Vision*, 2011.

S. Parameswaran and K. Q. Weinberger. Large margin multi-task metric learning. In *NIPS*, pages 1867–1875, 2010.

S.N. Parizi, J. Oberlin, and P. Felzenszwalb. Reconfigurable models for scene recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

M. Pedersoli, A. Vedaldi, and J. Gonzalez. A coarse-to-fine approach for fast deformable object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

D. N. Perkins and G. Salomon. Transfer of learning. *International Encyclopedia of Education (2nd ed.)*, 1992.

F. Perronnin, Y. Liu, J. Sanchez, and H. Poirier. Large-scale image retrieval with compressed fisher vectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis*, 2007.

A. Quattoni, M. Collins, and T.J. Darrell. Transfer learning for image classification with

sparse prototype representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *Proceedings of the International Conference on Computer Vision*, 2007.

M. Raginsky and S. Lazebnik. Locality sensitive binary codes from shift-invariant kernels. In *Advances in Neural Information Processing Systems*, 2009.

M. Rastegari, C. Fang, and L. Torresani. Scalable object-class retrieval with approximate and top-k ranking. In *Proceedings of the International Conference on Computer Vision*, 2011.

M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where – and why? semantic relatedness for knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

M.A. Sadeghi and A. Farhadi. Recognition using visual phrases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *Proceedings of the European Conference on Computer Vision*, 2010.

R. Salakhutdinov, A. Torralba, and J. B. Tenenbaum. Learning to share visual appearance for multiclass object detection. In *CVPR*, 2011.

P. Sharma and R Nevatia. Efficient detector adaptation for object detection in a video.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros. Data-driven visual similarity for cross-domain image matching. *ACM Trans. Graph.*, 2011.

S. Singh, A. Gupta, and A.A. Efros. Unsupervised discovery of mid-level discriminative patches. In *Proceedings of the European Conference on Computer Vision*, 2012.

J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, volume 2, pages 1470–1477, 2003.

H.O. Song, S. Zickler, T. Althoff, R.B. Girshick, M. Fritz, C. Geyer, P.F. Felzenszwalb, and T. Darrell. Sparselet models for efficient multiclass object detection. In *Proceedings of the European Conference on Computer Vision*, 2012.

Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan. Contextualizing object detection and classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

M. Stark, M. Goesele, and B. Schiele. A shape-based object class model for knowledge transfer. In *Proceedings of the International Conference on Computer Vision*, 2009.

E. L. Thorndike and R. S. Woodworth. The influence of improvement in one mental function upon the efficiency of other functions. *Psychological Review 8*, pages 247–564, 1901.

J. Tighe and S. Lazebnik. Finding things: Image parsing with regions and per-exemplar detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

T. Tommasi. *Learning to Learn by Exploiting Prior Knowledge*. PhD thesis, COLE POLYTECHNIQUE FDRALE DE LAUSANNE, 2013.

T. Tommasi and B. Caputo. The more you know, the less you learn: from knowledge transfer to one-shot learning of object categories. In *Proceedings of the British Machine Vision Conference*, 2009.

T. Tommasi, F. Orabona, and B. Caputo. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC*, 2004.

A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: a large dataset for non-parametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classemes. In *Proceedings of the European Conference on Computer Vision*, 2010.

I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research (JMLR)*, 6:1453–1484, September 2005.

A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008.

A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

A. Vedaldi and A. Zisserman. Sparse kernel approximations for efficient classification and detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *Proceedings of the International Conference on Computer Vision*, 2009.

P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 511–518, 2001.

G. Wang, D.S. Forsyth, and D. Hoiem. Comparative object similarity for improved recognition with few or no examples. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

M. Wang and X. Wang. Automatic adaptation of a generic pedestrian detector to a specific traffic scene. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

X. Wang, G. Hua, and T. X. Han. Detection by detections: Non-parametric detector adaptation for a video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

Y. Wang, R. Ji, and S. F. Chang. Label propagation from imagenet to 3d point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

Y. Weiss, A. Torralba, and R. Fergus. Spectral hashing. In *Advances in Neural Information Processing Systems*, 2008.

P. Wu and T.G. Dietterich. Improving svm accuracy by training on auxiliary data sources. In *ICML '04: Proceedings of the twenty-first international conference on Machine learning*, page 110, New York, NY, USA, 2004. ACM. ISBN 1-58113-828-5. doi: http://doi.acm.org/10.1145/1015330.1015436.

F. Xu, Y. Liu, C. Stoll, J. Tompkin, G. Bharaj, Q. Dai, H. P. Seidel, J. Kautz, and C. Theobalt. Video-based characters - creating new human performances from a multi-view video database. In *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, 2011.

J. Yang, R. Yan, and A.G. Hauptmann. Cross-domain video concept detection using adaptive svms. In *ACM Multimedia*, 2007a.

J. Yang, R. Yan, and A.G. Hauptmann. Adapting svm classifiers to data with shifted distributions. In *ICDM Workshops*, 2007b.

B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. J. Guibas, and F. F. Li. Human action recognition by learning bases of action attributes and parts. In *Proceedings of the International Conference on Computer Vision*, 2011.

K. Yu, V. Tresp, and A. Schwaighofer. Learning gaussian processes from multiple tasks. In *ICML*, pages 1012–1019, 2005.

C. Zhang, R. Hamid, and A. Zhang. Taylor expansion based classifier adaptation: Application to person detection. In *CVPR*, 2008.

W. Zhong and J. T. Kwok. Convex multitask learning with flexible task clusters. In *ICML*, 2012.

A. Zweig and D. Weinshall. Exploiting object hierarchy: Combining models from different category levels. In *Proceedings of the International Conference on Computer Vision*, 2007.