**THESE**

pour obtenir le grade de
**DOCTEUR DE L'ECOLE CENTRALE DE LYON**
Spécialité : Informatique

présentée et soutenue publiquement par

**Chao ZHU**

le 3 avril 2012

# Effective and Efficient Visual Description based on Local Binary Patterns and Gradient Distribution for Object Recognition

Ecole Doctorale InfoMaths

**Directeur de thèse: Liming CHEN**
**Co-directeur de thèse: Charles-Edmond BICHOT**

**JURY**

| | | |
|---|---|---|
| Prof. Matthieu CORD | Université Pierre et Marie Curie | Rapporteur |
| Prof. Jenny BENOIS-PINEAU | Université Bordeaux 1 | Rapporteur |
| Dr. Cordelia SCHMID | INRIA Grenoble Rhône-Alpes | Examinateur |
| Prof. Liming CHEN | Ecole Centrale de Lyon | Directeur de thèse |
| Dr. Charles-Edmond BICHOT | Ecole Centrale de Lyon | Co-directeur de thèse |

Numéro d'ordre: 2012-05

## Acknowledgments

I would like to express my gratitude here to many people who have been helping me during my thesis work since 2008. This thesis could not be accomplished without their help.

First of all, I would like to greatly thank my director of the thesis, **Prof. Liming CHEN**, for offering me the opportunity to work in his research team and supporting me with his instructive guidance during the whole thesis work. I have also been educated by his elegant demeanor and profound knowledge.

I am also greatly thankful to my co-director of the thesis, **Dr. Charles-Edmond BICHOT**, for his valuable advice, gentle care and encouragement, not only during my thesis work, but also in my everyday life. I am very grateful to be his first PhD student.

I would like to express my special thanks to **Prof. Matthieu CORD** and **Prof. Jenny BENOIS-PINEAU** for their precious time and hard work to review my thesis, and giving me valuable remarks to improve it. Special thanks to **Dr. Cordelia SCHMID** as well for being the president of the jury and examining my thesis.

It is a great pleasure for me to work in the research team of the LIRIS laboratory at the Department of Mathematics and Informatics in Ecole Centrale de Lyon. I would like to thank all the persons in the team, with whom I have passed the memorable last three and half years. My colleagues have often enlightened me during my research through exchange of opinions and ideas, while the personnel have helped me a lot in many problems concerning the administration, my life in France and other intractable situations.

At the end, I want to specially thank my family, who are the most important people for me in this world: my father **Yaozheng ZHU** and my mother **Shuling WANG**, for all their love, care and support.

# Contents

# List of Tables

# List of Figures

# Abstract

This thesis is dedicated to the problem of machine-based visual object recognition, which has become a very popular and important research topic in recent years because of its wide range of applications such as image/video indexing and retrieval, security access control, video monitoring, etc. Despite a lot of efforts and progress that have been made during the past years, it remains an open problem and is still considered as one of the most challenging problems in computer vision community, mainly due to inter-class similarities and intra-class variations like occlusion, background clutter, changes in viewpoint, pose, scale and illumination. The popular approaches for object recognition nowadays are feature & classifier based, which typically extract visual features from images/videos at first, and then perform the classification using certain machine learning algorithms based on the extracted features. Thus it is important to design good visual description, which should be both discriminative and computationally efficient, while possessing some properties of robustness against the previously mentioned variations. In this context, the objective of this thesis is to propose some innovative contributions for the task of visual object recognition, in particular to present several new visual features / descriptors which effectively and efficiently represent the visual content of images/videos for object recognition. The proposed features / descriptors intend to capture the visual information from different aspects.

Firstly, we propose six multi-scale color local binary pattern (LBP) features to deal with the main shortcomings of the original LBP, namely deficiency of color information and sensitivity to non-monotonic lighting condition changes. By extending the original LBP to multi-scale form in different color spaces, the proposed features not only have more discriminative power by obtaining more local information, but also possess certain invariance properties to different lighting condition changes. In addition, their performances are further improved by applying a coarse-to-fine image division strategy for calculating the proposed features within image blocks in order to encode spatial information of texture structures. The proposed features capture global distribution of texture information in images.

Secondly, we propose a new dimensionality reduction method for LBP called the orthogonal combination of local binary patterns (OC-LBP), and adopt it to construct a new distribution-based local descriptor by following a way similar to SIFT. Our goal is to build a more efficient local descriptor by replacing the costly gradient information with local texture patterns in the SIFT scheme. As the extension of our first contribution, we also extend the OC-LBP descriptor to different color spaces and propose six color OC-LBP descriptors to enhance the discriminative power and the photometric invariance property of the intensity-based descriptor. The proposed descriptors capture local distribution of texture information in images.

Thirdly, we introduce DAISY, a new fast local descriptor based on gradient distribution, to the domain of visual object recognition. It is well known that

gradient-distribution-based local descriptors such as SIFT, GLOH and HOG obtain the state-of-the-art performances in object recognition, while their drawback is relatively high computational cost. To deal with this, there are usually two ways: one is to replace the costly gradient information with other more efficient features, as what we did in the case of OC-LBP; the other is to find more efficient methods to calculate the gradient information. The DAISY descriptor was initially designed for wide-baseline stereo matching problem, and has shown good robustness against many photometric and geometric transformations. It has never been used in the context of visual object recognition, while we believe that it is very suitable for this problem. DAISY provides a fast way to capture the first order gradient information in images.

Fourthly, we propose a novel local descriptor called histograms of the second order gradients (HSOG) for visual object recognition. It captures the second order gradient information in images, which, to the best of our knowledge, is seldom investigated in the literature for the purpose of object recognition. Intuitively, the second order gradients applied to a gray level image capture the acceleration information on local variations of pixel gray values. They should not only offer certain discriminative power to distinguish different object classes, but also tend to be complementary to the description provided by the first order gradients. Thus we believe that both the first and second order gradient information is required to comprehensively describe the visual content of an image. Therefore, we propose the HSOG descriptor as a complement to the existing first order gradient descriptors, and further improve its performance by using multi-scale extension.

The proposed features / descriptors have been validated and evaluated through comprehensive experiments conducted on several popular datasets such as PASCAL VOC 2007, Caltech 101, and so on. The experimental results clearly show that (1) the multi-scale color LBP features outperform the original LBP and other popular texture features; (2) the gray and color OC-LBP descriptors obtain comparable or superior performances compared to the state-of-the-art descriptors such as SIFT and color SIFT while being more computationally efficient as well; (3) the DAISY descriptor outperforms the state-of-the-art SIFT in terms of both recognition accuracy and computational efficiency; (4) the HSOG descriptor obtains superior performance compared to the existing first order gradient based descriptors such as SIFT, CS-LBP and DAISY, and also provides complementary information to these descriptors.

**Keywords:** visual description; local descriptor; feature extraction; object recognition; scene classification; SIFT; DAISY; second order gradients; local binary patterns (LBP); color LBP descriptor; CS-LBP; orthogonal combination of LBP (OC-LBP).

# Résumé

Cette thèse est consacrée au problème de la reconnaissance visuelle des objets basé sur l'ordinateur, qui est devenue un sujet de recherche très populaire et important ces dernières années grâce à ses nombreuses applications comme l'indexation et la recherche d'image et de vidéo , le contrôle d'accès de sécurité, la surveillance vidéo, etc. Malgré beaucoup d'efforts et de progrès qui ont été fait pendant les dernières années, il reste un problème ouvert et est encore considéré comme l'un des problèmes les plus difficiles dans la communauté de vision par ordinateur, principalement en raison des similarités entre les classes et des variations intra-classe comme occlusion, clutter de fond, les changements de point de vue, pose, l'échelle et l'éclairage. Les approches populaires d'aujourd'hui pour la reconnaissance des objets sont basé sur les descripteurs et les classifieurs, ce qui généralement extrait des descripteurs visuelles dans les images et les vidéos d'abord, et puis effectue la classification en utilisant des algorithmes d'apprentissage automatique sur la base des caractéristiques extraites. Ainsi, il est important de concevoir une bonne description visuelle, qui devrait être à la fois discriminatoire et efficace à calcul, tout en possédant certaines propriétés de robustesse contre les variations mentionnées précédemment. Dans ce contexte, l'objectif de cette thèse est de proposer des contributions novatrices pour la tâche de la reconnaissance visuelle des objets, en particulier de présenter plusieurs nouveaux descripteurs visuelles qui représentent effectivement et efficacement le contenu visuel d'image et de vidéo pour la reconnaissance des objets. Les descripteurs proposés ont l'intention de capturer l'information visuelle sous aspects différents.

Tout d'abord, nous proposons six caractéristiques LBP couleurs de multi-échelle pour traiter les défauts principaux du LBP original, c'est-à-dire, le déficit d'information de couleur et la sensibilité aux variations des conditions d'éclairage non-monotoniques. En étendant le LBP original à la forme de multi-échelle dans les différents espaces de couleur, les caractéristiques proposées non seulement ont plus de puissance discriminante par l'obtention de plus d'information locale, mais possèdent également certaines propriétés d'invariance aux différentes variations des conditions d'éclairage. En plus, leurs performances sont encore améliorées en appliquant une stratégie de l'image division grossière à fine pour calculer les caractéristiques proposées dans les blocs d'image afin de coder l'information spatiale des structures de texture. Les caractéristiques proposées capturent la distribution mondiale de l'information de texture dans les images.

Deuxièmement, nous proposons une nouvelle méthode pour réduire la dimensionnalité du LBP appelée la combinaison orthogonale de LBP (OC-LBP). Elle est adoptée pour construire un nouveau descripteur local basé sur la distribution en suivant une manière similaire à SIFT. Notre objectif est de construire un descripteur local plus efficace en remplaçant l'information de gradient coûteux par des patterns de texture locales dans le régime du SIFT. Comme l'extension de notre première contribution, nous étendons également le descripteur OC-LBP aux différents

espaces de couleur et proposons six descripteurs OC-LBP couleurs pour améliorer la puissance discriminante et la propriété d'invariance photométrique du descripteur basé sur l'intensité. Les descripteurs proposés capturent la distribution locale de l'information de texture dans les images.

Troisièmement, nous introduisons DAISY, un nouveau descripteur local rapide basé sur la distribution de gradient, dans le domaine de la reconnaissance visuelle des objets. Il est bien connu que les descripteurs locaux basés sur la distribution de gradient tels que SIFT, GLOH et HOG obtenir les performances de l'état-de-l'art dans la reconnaissance des objets, tandis que leur coût de calcul est relativement élevé. Pour faire face à cela, il y a généralement deux façons: l'une est de remplacer l'information de gradient coûteux par d'autres caractéristiques plus efficaces, comme nous l'avons fait dans le cas d'OC-LBP; l'autre est de trouver des méthodes plus efficaces pour calculer l'information de gradient. Le descripteur DAISY a été initialement conçu pour le problème d'appariement stéréo de grande base, et a démontré une bonne robustesse contre les nombreuses transformations photométriques et géométriques. Il n'a jamais été utilisé dans le contexte de la reconnaissance visuelle des objets, tandis que nous croyons qu'il est très approprié pour ce problème. DAISY offre un moyen rapide pour capturer l'information de gradient du premier ordre dans les images.

Quatrièmement, nous proposons un nouveau descripteur local appelé histogrammes des gradients du second ordre (HSOG) pour la reconnaissance visuelle des objets. Il capture l'information de gradient du second ordre dans les images, qui, au meilleur de notre connaissance, est rarement étudiés dans la littérature aux fins de la reconnaissance des objets. Intuitivement, les gradients du second ordre appliqués à une image aux niveaux de gris capturent l'information d'accélération sur les variations de la valeur de gris des pixels locaux. Ils doivent non seulement offrir certaine puissance discriminante pour distinguer les différentes classes d'objet, mais ont aussi tendance à être complémentaires à la description fournie par les gradients du premier ordre. Ainsi nous pensons que l'information de gradient du premier et second ordre est nécessaire pour décrire complètement le contenu visuel d'une image. Par conséquent, nous proposons le descripteur HSOG comme un complément aux descripteurs existants de gradient du premier ordre, et améliorons encore sa performance en utilisant l'extension de multi-échelle.

Les descripteurs proposés ont été validés et évalués à travers des expériences complètes effectuées sur plusieurs bases de données populaires comme le PASCAL VOC 2007, Caltech 101, etc.

**Mots-clés:** description visuelle; descripteur local; l'extraction de caractéristiques; la reconnaissance des objets; la classification de scène; SIFT; DAISY; les gradients du second ordre; local binaire patterns (LBP); descripteur de LBP couleur; CS-LBP; la combinaison orthogonale de local binaire patterns (OC-LBP).

Chapter 1

# Introduction

Contents



## 1.1 Context

With the rapid development of digital technology, the world is currently experiencing a digital revolution. Particularly, because of the speedy popularization of digital cameras and camera phones, more and more information presented around us nowadays are changing from text-based to multimedia-based, especially in the form of images and videos. For example, the very famous online photo sharing website "Flickr" [1] reported in August 2011 that it was hosting more than 6 billion photos already and this number continues to grow with a speed of more than 1 billion per year. Another famous social networking website "Facebook" [2] announced in October 2011 that it was hosting about 140 billion images and thus becomes the largest album in the world.

Facing such huge amounts of data, the need for solutions of how to efficiently manage them and access to appropriate content becomes more and more urgent. Traditionally, one could first annotate images manually using keywords and then

[1] http://www.flickr.com/
[2] http://www.facebook.com/

carry out the search by matching their annotations with the required keywords, just as the cases of the most popular image search engines nowadays like Google Images [3], Yahoo Images [4] and Picsearch [5]. Technically, this kind of search method relies not on the image content directly, but on the textual information associated with images, e.g. file name, keywords, labels or tags. However, this method quickly becomes inconceivable nowadays because tremendous amount of time and labor is required for annotating such huge amounts of data. Moreover, there exist some other problems for manual annotations:

- When the annotation rules change, the annotation process must be manually performed again on the whole database.

- Since manual annotation might be subjective, there is no guarantee that two different persons would produce the same annotation for the same image, which however is generally expected in most applications.

- Since the annotations are in the form of text, choosing language is important for annotating and searching, while most of available annotations are only for a limited number of languages.

In such context, the current trend is to find out effective and efficient methods to realize automatic image annotation, which means that single or multiple labels could be assigned to an image automatically by computers according to its visual content. Another way is to skip the annotation step and to realize the content-based image retrieval directly. For these purposes, more and more attentions in recent years have been paid to machine-based visual object recognition and image classification, which serves as the fundamental problem and could greatly be beneficial to the mentioned applications.

[3] http://images.google.com/
[4] http://images.search.yahoo.com/
[5] http://www.picsearch.com/

## 1.2 Problems and objective

Machine-based visual object recognition aims at automatically predicting whether at least one or several objects of given categories are present in an image by computers based on its visual content. More precisely, only categories of objects or generic concepts are taken into account as the goal of object recognition systems. For example, given an image, we aim to find out if there exists any person or any building in it, rather than a particular person or a particular building. Figure 1.1 shows some instances of generic object categories "Car", "Aeroplane", "Cat" and "Sofa" respectively.

In fact, visual object recognition is a fundamental problem in computer vision and pattern recognition. It has a wide range of possible applications besides automatic image annotation, such as video monitoring, video coding systems, security access control, robot localization, automobile driving support and content-based image / video indexing and retrieval. Therefore, it has become a very popular and important research topic in computer vision community in recent years, and many different methods have been proposed and applied for the recognition of generic object categories such as vehicles, animals, person, plants, buildings, and so on [Sivic & Zisserman 2003] [Csurka *et al.* 2004] [Marszalek & Schmid 2006] [Marszalek & Schmid 2007] [Lazebnik *et al.* 2006] [Hegerath *et al.* 2006] [Lowe 2004] [Zhang *et al.* 2007] [van de Sande *et al.* 2010] [Zhang *et al.* 2006] [Chevalier *et al.* 2007] [Yang *et al.* 2009b] [Gorisse *et al.* 2010] [Wang *et al.* 2009a] [Guillaumin *et al.* 2010] [Harzallah *et al.* 2009] [van Gemert *et al.* 2010]. Despite a lot of efforts and progress that have been made during the past years [Everingham *et al.* 2010] [Smeaton *et al.* 2009], visual object recognition remains an open problem and is still considered as one of the most challenging problems in computer vision. The main reason lies in the difficulties for computers to cope with various intra-class variations, including appearance deformation, occlusion, background clutter, changes in viewpoint, pose, scale and illumination, etc., which although are much easier problems for human. The typical intra-class variations of object are illustrated by the horse images in Figure 1.2.

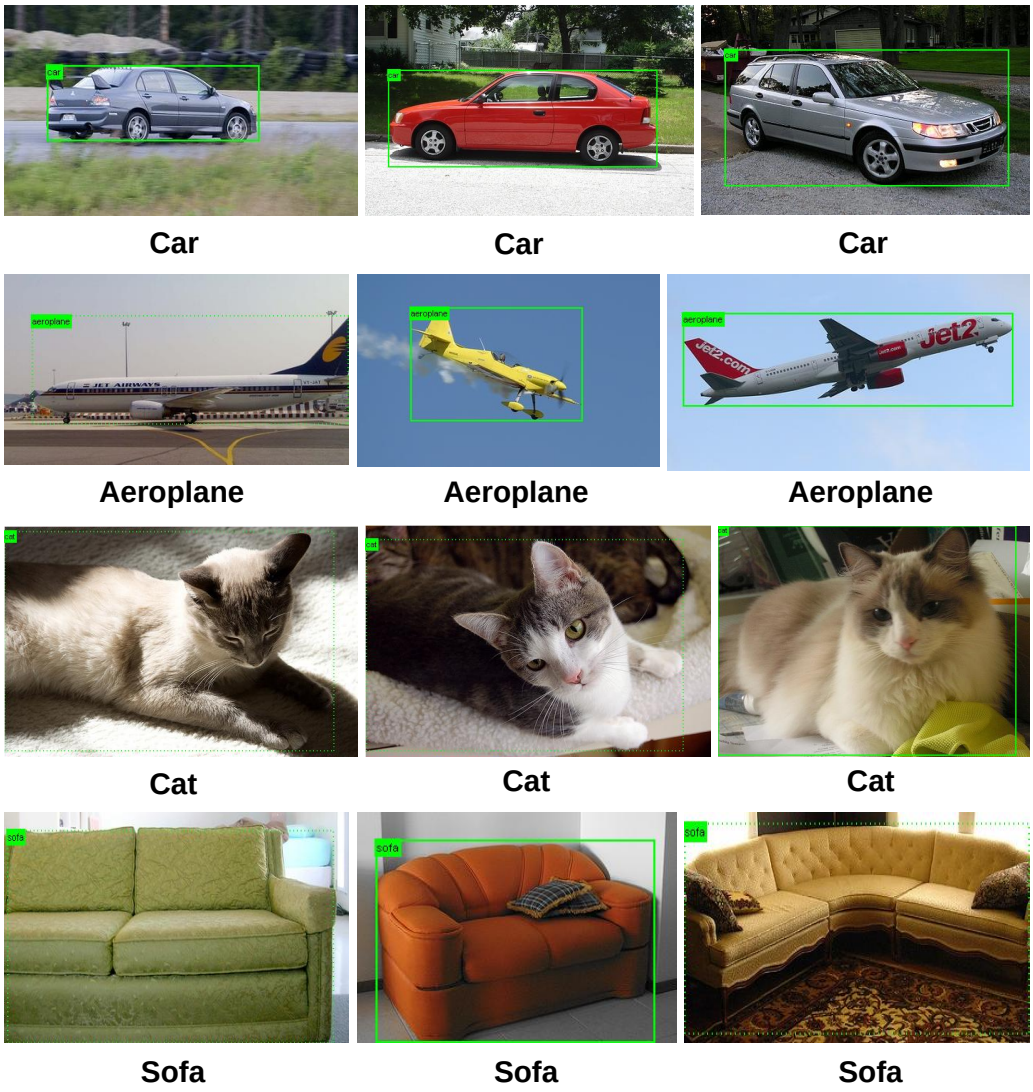



Figure 1.1: Different instances of generic object categories (example images from PASCAL VOC 2007 database)

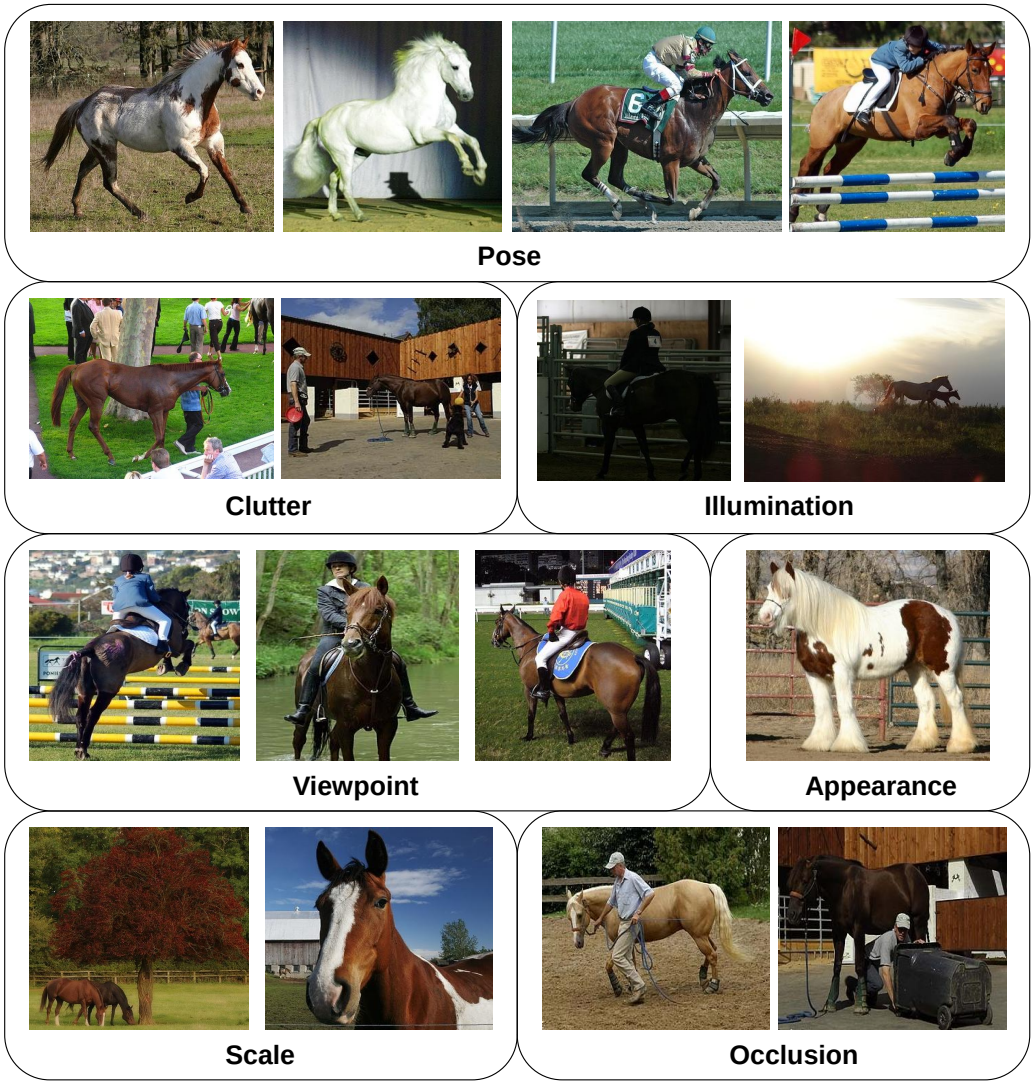



Figure 1.2: An illustration of various variations of object in the same category (example images of the category "horse" from PASCAL VOC 2007 database)

In this context, the objective of this thesis is to propose some innovative contributions for visual object recognition task, in particular concerning several new visual features / descriptors to effectively and efficiently represent the visual content of objects in images for recognition. The proposed approaches have been validated through comprehensive experiments conducted on several popular datasets.

## 1.3 Approaches and contributions

As we stated, visual object recognition is a very challenging problem, and a lot of factors need to be considered to construct a successful system. Generally speaking, the most important factors lie in two main steps: (1) image feature extraction and (2) image classification. Image feature extraction aims at extracting compact and informative feature vectors or descriptors rather than using the raw data from an image to represent its visual content. This is the very first but also important step because the raw data of an image are usually too huge and impractical to be used directly for the following classification step. Considering the difficulties mentioned in the previous section, we hold that a good image feature / descriptor should be both discriminative enough and computationally efficient, while possessing some properties of robustness to changes in viewpoint, scale and lighting conditions. Many different image features / descriptors have been proposed in the literature, and the most successful ones are distribution-based local descriptors, such as SIFT [Lowe 2004], GLOH [Mikolajczyk & Schmid 2005], HOG [Dalal & Triggs 2005], Shape context [Belongie *et al.* 2002], etc., because of their good performances. Image classification aims at constructing a robust classifier which could effectively classify an image or object into given categories based on the extracted image feature vectors or descriptors. Many different classifiers have also been proposed in the past years, such as Support Vector Machines (SVM) [Cortes & Vapnik 1995], K-Nearest Neighbors (KNN) [Cover & Hart 1967], Artificial Neural Networks (ANN) [Bishop 1995], Decision Trees (DT) [Quinlan 1993], Adaboost [Freund & Schapire 1997], etc., where the most popular one nowadays is SVM.

In this thesis, we mainly focus on image feature extraction by proposing sev-

eral new image features / descriptors for the task of object recognition, and then apply the SVM classifier on the proposed features / descriptors to obtain the final classification results. The proposed features / descriptors intend to capture an object's information from different aspects, including global texture distribution, local texture distribution, the first order gradients and the second order gradients. Our contributions are summarized as follows.

Our first contribution lies in proposing six multi-scale color local binary pattern features for visual object recognition. The local binary pattern (LBP) operator [Ojala *et al.* 2002b] is a computationally efficient yet powerful feature for analyzing image texture structures, and has been successfully applied to the applications as diverse as texture classification [Mäenpää *et al.* 2000a] [Mäenpää *et al.* 2000b] [Ojala *et al.* 2002b], texture segmentation [Ojala & Pietikäinen 1999], face recognition [Ahonen *et al.* 2004] [Ahonen *et al.* 2006] and facial expression recognition [Zhao & Pietikäinen 2007] [Shan *et al.* 2009]. However, it has been rarely used for the task of visual object recognition [6]. We hold that the main reasons lie in two aspects. On one hand, the LBP operator ignores all color information (its calculation is based on gray image), while color is an important clue for distinguishing objects, especially in natural scenes. On the other hand, there can be various changes in lighting and viewing conditions in real-world scenes, leading to large illumination variations of object's appearance, which further complicate the recognition task. According to its definition, the LBP operator is only invariant to gray-level monotonic light changes, and thus has difficulty to deal with the mentioned variations. Therefore, in order to incorporate color information, as well as to enhance the discriminative power and the photometric invariance property of the original LBP, we propose, in chapter 4, six multi-scale color LBP features which are more suitable for visual object recognition task. Moreover, we apply a coarse-to-fine image division strategy for calculating the proposed features within image blocks in order to encode spatial information of texture structures, thereby further improving their performances.

Our second contribution consists of proposing a new dimensionality reduction

[6] at the time when we started our work in 2008, while being more popular now

method for LBP called the orthogonal combination of local binary patterns (denoted as OC-LBP), and several new local descriptors based on OC-LBP for image region description. Nowadays, distribution-based local descriptors, such as SIFT and its extensions or refinements, have become the dominant features in the state-of-the-art recognition / classification systems. However, the downside of these descriptors is their high computational cost, especially when the size of image or the scale of dataset significantly increases. Therefore, it is highly desirable that local image descriptors offer both high discriminative power and computational efficiency. As we mentioned earlier, the LBP operator is a well known texture feature which has several interesting properties. First of all, it is simple and fast to compute. Moreover, it offers strong discriminative power for describing texture structures while staying robust to monotonic lighting changes. All these advantages make LBP a good candidate for constructing a local descriptor. However, the LBP operator tends to produce high dimensional feature vectors, especially when the number of considered neighboring pixels increases. The so-called "curse of dimensionality" is a barrier for using it directly to construct a local descriptor. Thus, a key issue of making use of LBP as a local descriptor is to reduce its dimensionality. For this purpose, we propose, in chapter 5, a new dimensionality reduction method for LBP, denoted as the orthogonal combination of local binary patterns (OC-LBP), which proves much more effective compared to the other popular methods such as "uniform patterns" [Ojala *et al.* 2002b] and CS-LBP operator [Heikkilä *et al.* 2009], because our method produces the LBP features with the smallest dimensions while still offering high discriminative power of local texture patterns. The proposed OC-LBP operator is then adopted to construct a distribution-based local image descriptor, denoted as the OC-LBP descriptor, by following a way similar to SIFT. Our aim is to build a more efficient local descriptor by replacing the costly gradient information with local texture patterns in the SIFT scheme. Moreover, since color plays an important role for object recognition and classification especially in natural scenes, as we declared in the first contribution, we further extend our OC-LBP descriptor to different color spaces and propose six color OC-LBP descriptors to enhance the photometric invariance property and the discriminative power of intensity-based descriptor. This

work could thus be considered as the extension of our first contribution.

Our third contribution is introducing the DAISY descriptor to the task of visual object recognition. There is now a trend in computer vision community that the scale of the benchmark datasets used for object recognition / image classification becomes larger year by year. However, it is well known that the most popular and state-of-the-art features are gradient-distribution-based local descriptors such as SIFT, GLOH and HOG, whose drawback is their relatively high computational cost. Thus, more computationally efficient and discriminative local descriptors are urgently demanded to deal with large scale datasets such as ImageNet [Deng *et al.* 2009] and TRECVID [Smeaton *et al.* 2006]. Usually, there are two ways to do this. One way is to replace the costly gradient information with other more efficient features, as what we did in the case of the OC-LBP descriptor. The other way is to find more efficient methods to calculate the gradient information. The DAISY descriptor [Tola *et al.* 2010], which was initially designed for wide-baseline stereo matching problem, is a new fast local descriptor based on gradient distribution, and has shown good robustness against many photometric and geometric transformations. It has never been used in the context of visual object recognition, while we believe that it is very suitable for this problem, and could well meet the mentioned demand. Therefore, we investigate the DAISY descriptor, in chapter 6, for the task of visual object recognition by evaluating and comparing it with SIFT both in terms of recognition accuracy and computation complexity on two standard image benchmarks. DAISY provides a fast way to calculate the gradient information and proves very promising for the task of visual object recognition.

Our fourth contribution lies in proposing a novel local image descriptor called histograms of the second order gradients (HSOG) for visual object recognition. In the literature, the first order gradient information is the most effective feature for characterizing an object's appearance or the content of an image, since it can reflect the pixel intensity changes for different directions in a small neighborhood around each pixel. Thus, many successful and state-of-the-art descriptors, such as SIFT, GLOH, HOG and DAISY, are constructed based on the first order gradient distribution (histogram) in a local region. However, to the best of our knowledge, local

descriptors focusing on the second order gradients are seldom investigated in the literature for the purpose of object recognition. Intuitively, the second order gradient information should not only possess certain discriminative power to distinguish different objects, but also tends to be complementary to the information provided by the first order gradients. This hypothesis is motivated by a physical analogy of object motion. Velocity and acceleration of an object are both needed to comprehensively describe a motion process within an unit displacement, which is better than using only velocity. Connecting these concepts to an image, within a predefined distance between two pixels, the first order gradients simulate the velocity of pixel intensity changes, while the second order gradients imitate its acceleration. In order to ameliorate the quality of visual content representation, both the first and second order gradient information is valuable. Therefore, we propose, in chapter 7, a novel local image descriptor called histograms of the second order gradients (HSOG) for the task of visual object recognition. Its construction consists of first computing several first order oriented gradient maps and then building the second order oriented gradient histograms based on these maps. A DAISY-style spatial pooling arrangement is adopted for taking into account the spatial information, and the principal component analysis (PCA) [Jolliffe 2002] is applied for dimensionality reduction. The performance of the proposed descriptor is further improved by using multi-scale strategy, which combines the descriptors computed from several concentric local regions with different size by late fusion.

## 1.4 Organization of the thesis

The rest of this thesis is organized as follows.

- In chapter 2, a review of related work on visual object recognition is presented. More attention is paid to the feature & classifier based approaches, which include image feature extraction; image representation (modelling); classification algorithms; and fusion strategies.

- In chapter 3, we introduce several standard datasets and popular benchmarks

available in computer vision community for object recognition and image / video classification tasks. Some of them will be used to carry out experiments in the following chapters.

- In chapter 4, we give the details of the proposed multi-scale color local binary pattern features, together with the analysis of their invariance properties, and show their effectiveness on the PASCAL VOC 2007 benchmark.

- In chapter 5, we first introduce the orthogonal combination of local binary patterns (OC-LBP) which is proposed as a new dimensionality reduction method for LBP. Its effectiveness is shown by comparing with other two popular methods on a standard texture classification dataset. Then we give the details of the proposed gray and color OC-LBP descriptors, and show their effectiveness in three different applications by comparing with the state-of-the-art SIFT and color SIFT descriptors both in terms of accuracy and computational cost.

- In chapter 6, we first present the details of the DAISY descriptor, and then introduce our approach of using DAISY for visual object recognition. Based on two standard image datasets, the Caltech 101 and the PASCAL VOC 2007, we compare DAISY with SIFT both in terms of recognition accuracy and computation complexity. Furthermore, the influence of different parameters in DAISY is analyzed.

- In chapter 7, we give the details of how to compute and construct the proposed histograms of the second order gradients (HSOG) descriptor, and show its effectiveness on the Caltech 101 dataset. The influence of different parameters in HSOG is also experimentally analyzed.

- In chapter 8, we give our conclusions as well as some perspectives for future research directions.

Chapter 2

# Literature Review

## Contents



In this chapter, we give a review of main approaches and related work for visual object recognition in the literature. First of all, we briefly introduce main approaches proposed for the problem of object recognition by generally dividing them into 4 categories according to the timeline: (1) geometry & matching based; (2) appearance & sliding window based; (3) parts & structure based; and (4) feature & classifier based. Then, since feature & classifier based approaches have become the most popular nowadays, a more detailed introduction of them is presented, including image feature (global or local) extraction; image representation (or modelling);

and image classification (generative or discriminative classifiers). In addition, we introduce different fusion strategies which aim to improve recognition performance by fusing different features, since they may carry complementary information to each other.

## 2.1 Introduction of main approaches for object recognition

The recognition of object categories in images and videos is a challenging problem in computer vision, especially when the number of categories is large. The main reasons are due to both high intra-class variations and inter-class similarities. Objects within the same category may look very different, while objects from different categories may look quite similar (see Figure 2.1 and 2.2 for illustrations). Moreover, depending on different viewpoint, scale and illumination, the same object may even appear dissimilar in images. Background clutter and partial occlusion also increase the difficulties of object recognition (see Figure 1.2 for an illustration).

In order to address this challenging problem, a lot of attention and efforts have been paid during the past decades by the researchers in computer vision community, and many approaches have been proposed in the literature. These approaches can be generally divided into 4 categories according to the timeline:

- Geometry & matching based approaches
- Appearance & sliding window based approaches
- Parts & structure based approaches
- Feature & classifier based approaches

### 2.1.1 Geometry & matching based approaches

The earliest attempts on object recognition mainly focused on using geometric models to represent objects. The main idea is that geometric descriptions of a three-dimensional (3D) object allow the projected shape to be accurately predicated in a

Figure 2.1: An illustration of intra-class variations. Examples are all from the class "chair" of the Caltech image dataset, but have very different appearances.

Figure 2.2: An illustration of inter-class similarities. Examples in the first row are from the class "bike" of the Caltech image dataset, while the ones in the second row are from the class "motorbike" of the same dataset. They are quite similar in appearance.

two-dimensional (2D) image under perspective projection, therefore the recognition of geometric descriptions can be achieved by using edge or boundary information, which is invariant to certain illumination changes [Mundy 2006]. L.G. Roberts with his blocks world model [Roberts 1963] is considered as the origin of computer vision and object recognition. The blocks world model is a simplification of the real world where objects are restricted to polyhedral shapes on a uniform background. Polyhedra have simple and easily represented geometry and the projection of polyhedra into images under perspective can be straightforwardly modeled with a projective transformation. Roberts carefully considered how polyhedra project into perspective images and established a generic library of polyhedral components that could be assembled into a composite structure. While the blocks world model only considers straight lines and flat surfaces as shown in Figure 2.3(a), Guzman [Guzman 1971] extended it to deal with curved surfaces and boundaries. He avoided difficult scene rendering issues by restricting the problem to line drawings, and focused on what happens when curved surfaces intersect. An example of line drawing for curved objects is shown in Figure 2.3(b). The drawback of this method is the restriction to ideal line drawings, which is far away from the real vision problem. Subsequently, a new geometric representation, the generalized cylinder (GC), was developed by Binford with his students [Binford 1971] [Agin 1972] [Nevatia & Binford 1977] to extend the blocks world to composite curved shapes in 3D. Their key idea is that many curved shapes can be expressed as a sweep of a variable cross section along a curved axis. Figure 2.3(c) gives an example. A lot of attention was also paid to extract geometric primitives such as lines, circles, etc., which are invariant to certain viewpoint and illumination changes [Mundy & Zisserman 1992].

To work with geometric models, the dominant object recognition approach during this period was based on alignment and matching, which means that two objects are directly compared by matching their geometric models after alignment to decide how similar they are. The work of Huttenlocher and Ullman [Huttenlocher & Ullman 1987] is considered as a representative, where an object is first aligned with an image using a small number of model pairs and image features, and then the aligned model is compared directly against the image to check

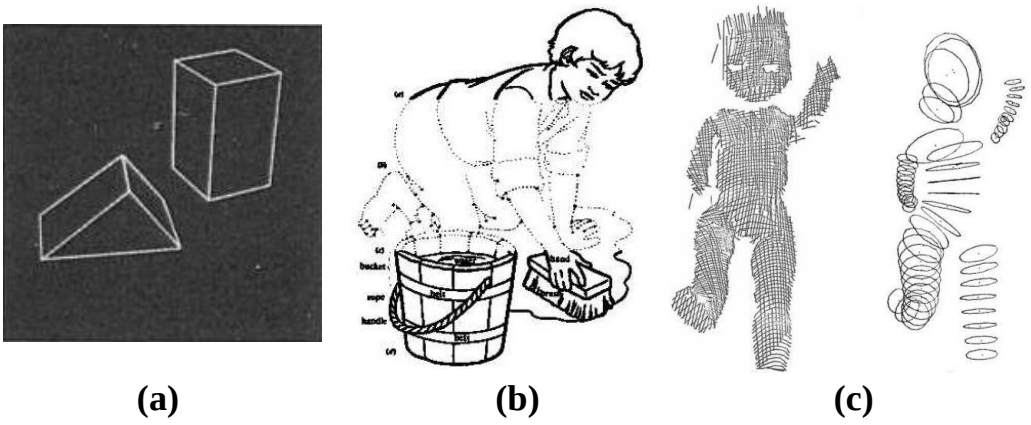



Figure 2.3: Geometry-based object recognition: (a) A 3D polyhedral description of the blocks world scene [Roberts 1963]. (b) The feature analysis of a line drawing for describing curved objects [Guzman 1971]. (c) A range image of a doll and the resulting set of generalized cylinders [Agin 1972].

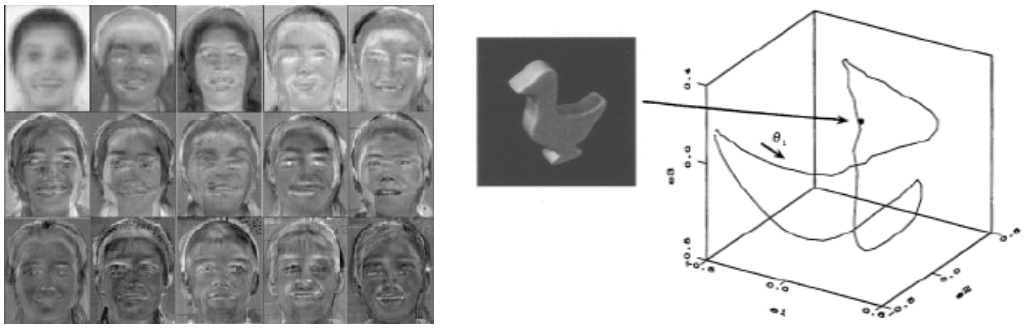
if the expected features are present. This method is able to detect transformations not only in scale and illumination conditions, but also in viewing angle. Thus it is able not only to identify the viewed object, but also to estimate the actual pose and 3D position of the object. However, this approach is computationally very expensive because the stored models are usually 3D internal representations of the object and the image features are formed exhaustively. A comprehensive review of geometry-based object recognition can be found in [Mundy 2006].

### 2.1.2 Appearance & sliding window based approaches

At the time when geometry-based approaches reached the end of their active period, more efforts had started to be focused on appearance-based techniques. The most representative methods of appearance-based techniques are eigenfaces [Turk & Pentland 1991a] and appearance manifolds [Murase & Nayar 1995]. Turk and Pentland proposed in 1991 the eigenfaces method [Turk & Pentland 1991a] which is considered as one of the first face recognition systems that are both computationally efficient and relatively accurate. Their approach treats the face recognition problem as an intrinsically 2D recognition problem rather than requiring 3D geometry recovery. The main idea is to project face images into a feature space that

spans the significant variations among the known face images. A set of vectors are first generated to represent each of the known face images by their gray-level pixel values, the eigenvectors are then computed by selecting the principal components from this set of vectors. These eigenvectors, denoted as eigenfaces, capture main variance among all the vectors, and a small set of eigenvectors could capture almost all the appearance variations of the face images in the training set. For a particular face image, its pixel value vector is projected into a feature space spanned by a set of eigenvectors so that it can be represented by a weighted sum of the eigenfaces with minimum error, and its recognition thus consists of comparing these weights with those of the known faces to find its nearest neighbor. Some examples of eigenfaces are shown in Figure 2.4(a). The idea of eigenfaces was then adopted and extended by Murase and Nayar in 1995 to recognize generic 3D objects with different viewpoints [Murase & Nayar 1995]. They proposed a compact representation of object appearance which is parameterized by viewpoint and illumination. For each object of interest, a large set of images is obtained by automatically varying viewpoint and illumination. This image set is compressed to obtain a low-dimensional continuous subspace, called the eigenspace, where the object is represented as a manifold. For an unknown input object, it is first projected into the eigenspace, and the recognition is then achieved by finding its closest manifold using Euclidean distance. The exact position of the projection on the manifold determines the viewpoint of the object, as illustrated in Figure 2.4(b).

As appearance-based methods generally require to only focus on the object part and not on the other disturbing parts such as background clutter, the "sliding window" technique is widely applied to cooperate with them. Its basic idea is to slide a window across the image at different scales and to recognize each sub-window as containing the target object or not. This technique was first applied on face recognition problems [Turk & Pentland 1991b] [Belhumeur *et al.* 1997] [Viola & Jones 2001], and then extended to generic object recognition [Papageorgiou & Poggio 2000] [Agarwal & Roth 2002] [Schneiderman & Kanade 2004]. The potential advantage of these sliding window-based techniques is their ability of achieving object recognition and localization at the same time. Their drawback lies in the failure of



(a) (b)

Figure 2.4: Appearance-based object recognition: (a) Some example images of eigenfaces (http://www.geop.ubc.ca/CDSST/eigenfaces.html/). (b) An illustration of 3D object recognition based on appearance manifolds [Murase & Nayar 1995].

detecting non-rigid deformable objects or objects that can not be shaped by a rectangle. While appearance-based methods have achieved promising results in object recognition tasks, they are not capable enough of handling occlusion, as well as pose and illumination change. In addition, a large set of samples needs to be collected to learn the appearance characteristics and thus requires a high computational cost. All these limitations have encouraged researchers to pay more attention to the parts and structure based approaches.

### 2.1.3 Parts & structure based approaches

The idea of parts and structure based approaches comes from the observation that most objects generally consist of several individual parts which are arranged in certain geometric structures. For example, a face consists of two eyes, one nose and one mouth, while an airplane consists of two wings, one fuselage and one tail. The parts-based deformable models were thus proposed to exploit this observation by decomposing an object into connected parts. For an object, each part encodes its local visual properties, while the deformable configuration is represented by connections between certain pairs of parts to define its global geometric structure. The recognition is achieved by finding the best match of such a parts-based model to an input image. The best match can be found by minimizing an energy function which

measures both a match cost for each part and a deformation cost for each pair of connected parts.

The application of parts-based deformable models for object recognition can trace back to the work of Fischler and Elschlager in 1973 [Fischler & Elschlager 1973], and has attracted renewed attention in early 2000s [Weber *et al.* 2000] [Ullman *et al.* 2001] [Fergus *et al.* 2003] [Bouchard & Triggs 2005] [Felzenszwalb & Huttenlocher 2005]. In [Fischler & Elschlager 1973], the authors proposed a parts-based model for face consisting of hair, eyes, nose, mouth and left/right edges, along with spring-like connections between certain pairs of parts, as depicted in Figure 2.5(a). In [Weber *et al.* 2000], objects are represented as flexible constellations of rigid parts which are automatically identified by applying a clustering algorithm on the training set. A statistical shape model is then learned on these parts by a maximum likelihood unsupervised algorithm to get the recognition results. In [Ullman *et al.* 2001], objects within a class are represented in terms of common image fragments that are used to build blocks for representing a large variety of different objects in a common class. The fragments are selected from a training image set based on a criterion of maximizing the mutual information between the fragment and its class. For recognition, the algorithm detects the fragments of different types and combines the evidence of the detected fragments to make the final decision. In [Fergus *et al.* 2003], the authors followed the work of [Weber *et al.* 2000], and proposed a number of improvements to its constellation model and learning algorithm, such as taking the variability of appearance into account, learning appearance simultaneously with shape, and extending the learning algorithm to efficiently learn new object categories. The examples of the learned models for motorbike and car are shown in Figure 2.5(b). In [Bouchard & Triggs 2005], the authors extended the work of [Fergus *et al.* 2003], and proposed a two-level hierarchical generative model for coding the geometry and appearance of visual object categories. The model is a collection of loosely connected parts containing more rigid assemblies of subparts. They also simplified the correspondence problem by using greedy nearest-neighbor matching in location-appearance space to deal with

many more subparts. Some examples of their models for motorbike and aeroplane are shown in Figure 2.5(c). In [Felzenszwalb & Huttenlocher 2005], the authors proposed a computationally efficient framework for parts-based modeling and object recognition which was motivated by the pictorial structure models introduced in [Fischler & Elschlager 1973]. They represented an object by a collection of parts arranged in a deformable configuration using spring-like connections between pairs of parts, and demonstrated the techniques by learning models that represent face and human body. Figure 2.5(d) shows some examples of the learned models for human body.

Parts and structure based approaches have several advantages. Firstly, while the global appearance of an object may significantly vary within a category, the appearance and spatial relationship of its local parts can often still be stable to provide important cues. Secondly, many natural object categories, such as human and animals, have relatively rigid global shape, but with significant shape variability, and parts-based models can easily represent this kind of covariance structure. However, most approaches can not handle large viewpoint variations or severe object deformations. Moreover, parts-based models require an exponentially growing number of parameters as the number of parts increases. Learning and inference problems for spatial relations also remain very complex and computationally expensive. The recent trend is to apply parts-based models for object detection and localization, rather than for object recognition. A successful example is the discriminatively trained deformable part model [Felzenszwalb *et al.* 2008] [Felzenszwalb *et al.* 2010], which has become the dominant approach in object detection task of the famous PASCAL VOC Challenge [Everingham *et al.* 2010].

### 2.1.4 Feature & classifier based approaches

Feature and classifier based approaches have become popular for object recognition since late 1990s, because of the great development of advanced image features / descriptors and pattern recognition algorithms in the community. Particularly, using local descriptors, e.g. SIFT [Lowe 2004], together with the Bag-of-Features (BoF) representation [Csurka *et al.* 2004] followed by discriminative classifiers such

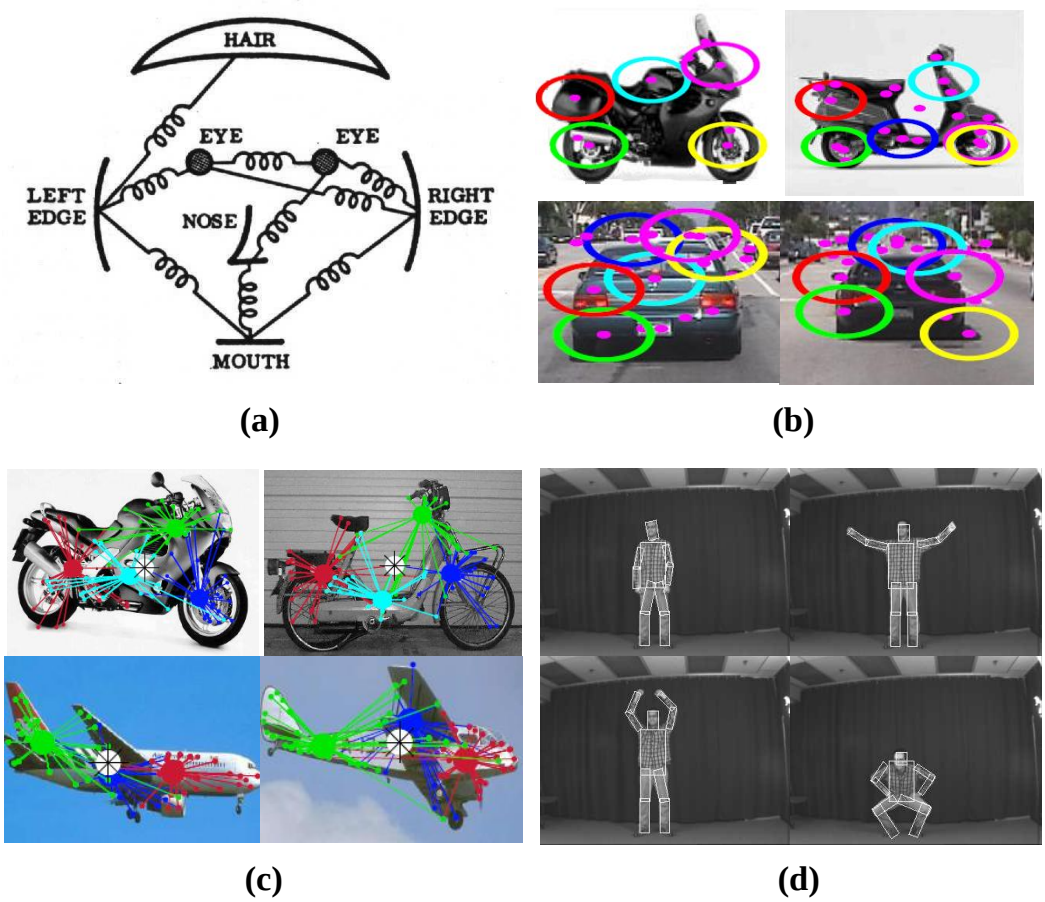



Figure 2.5: Parts-based object recognition: (a) The parts-based deformable model for face from [Fischler & Elschlager 1973]. (b) The parts-based deformable models for motorbike and car from [Fergus *et al.* 2003]. (c) The parts-based deformable models for motorbike and aeroplane from [Bouchard & Triggs 2005]. (d) The parts-based deformable models for human body from [Felzenszwalb & Huttenlocher 2005].

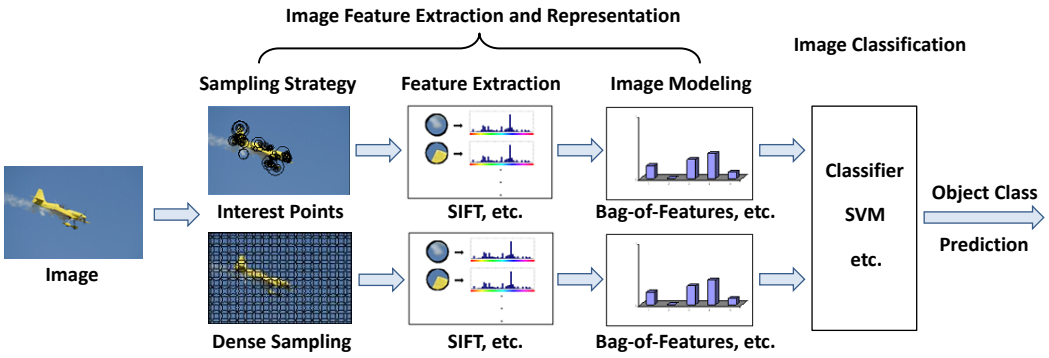



Figure 2.6: An overview of feature and classifier based object recognition (revised from Figure 2 & 3 in [van de Sande *et al.* 2010])

as Support Vector Machine (SVM) [Cortes & Vapnik 1995] has become the dominant paradigm since 2004. Generally speaking, feature and classifier based approaches consist of two main steps, as depicted in Figure 2.6. The first step is image feature extraction and representation, which aims to extract a set of feature vectors, or descriptors, from an image to describe its visual content, and to transform the extracted features into more compact and informative representations by applying certain image modelling methods. The second step is image classification, which accepts the image representations based on the extracted features and performs the final classification by utilizing certain pattern recognition algorithms (classifiers). In addition, as different features may carry complementary information to each other, fusion strategies are also required to further improve the recognition performance. The following sections will focus on these three aspects.

## 2.2 Image feature extraction and representation

The first step of image analysis for object recognition is to transform an image into the input data for subsequent process. A direct way is to concatenate gray or color values of all the pixels within an image. However, this will result in a very high-dimensional vector with a lot of redundant information. It is also very sensitive to any image variations. Therefore, image feature extraction is required, aiming at transforming the content of an image into a set of feature vectors, or descriptors,

which are expected to be discriminative, computationally efficient, with reasonable size, and possessed of some robustness properties to image variations (viewpoint, scale, illumination, etc.). After this step, the following process will no longer rely on the image itself, but only on the information carried by the extracted features. Thus, feature extraction is a very important step to ensure the final good performance of object recognition, and can be considered as the basis of the whole process.

A lot of feature extraction methods have been proposed in the literature, and we could summarize them into two main categories: global features and local features.

### 2.2.1 Global features and corresponding representations

Early work in this domain has mainly utilized global features as image description. These features are extracted directly from the whole image, and generally take the form of a single vector or histogram based on the statistical analysis of an image pixel by pixel. They thus encode global visual content of an image. Different global features have been proposed in the literature, and we present here several ones that we have studied and investigated in our work. We choose these features since they are the most popular ones among global features. An evaluation and comparison of different global features in the context of object recognition is given in Appendix B. These global features could be divided into three categories: (1) color, (2) texture and (3) shape.

#### 2.2.1.1 Color features

Color is perhaps the most direct and expressive of all the visual features. Color features aim at capturing color information, such as color distribution, relationship between different colors, etc., contained in an image.

- **Color Histogram** [Swain & Ballard 1991]: Color histogram is the simplest and most common way for expressing the color characteristics of an image. It is a representation of the color distribution of image pixels. Generally, each channel of the image's color space, such as RGB or HSV, is first quantized into an appropriate number of color ranges (called "bins"), and a histogram is

then built by counting the number of image pixels located in each bin. The more number of bins are selected, the more detailed color distribution could be obtained, but the higher dimensional histogram will be generated. The number of bins is thus a trade-off between feature information and size. Color histogram is invariant to translation and rotation of the viewing axis, and robust to viewpoint change, but with no spatial information.

- **Color Moments** [Stricker & Orengo 1995]: Color moments characterize the color distribution of an image into a very compact vector containing the mean, variance and skewness, which are respectively the moments of the 1st order, the 2nd order and the 3rd order as shown in (2.1), (2.2) and (2.3), for each channel of the image's color space.

$$E_i = \frac{1}{N}\sum_{j=1}^{N} p_{ij} \tag{2.1}$$

$$\sigma_i = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(p_{ij} - E_i)^2} \tag{2.2}$$

$$S_i = \sqrt[3]{\frac{1}{N}\sum_{j=1}^{N}(p_{ij} - E_i)^3} \tag{2.3}$$

where $i$ is the index of each channel, $N$ is total number of image pixels, and $p_{ij}$ is the value of the $j$-th pixel in channel $i$. Color moments have the same invariance properties and drawbacks as color histogram.

- **Color Coherence Vectors** [Pass *et al.* 1996]: Color coherence vectors propose to consider the coherent colors and the incoherent colors separately in an image. A color is defined as coherent if its proportion of pixels located in a spatial neighborhood area is bigger than a predefined threshold, otherwise it is defined as incoherent. Color coherence vectors first classify each pixel in a given color bin as either coherent or incoherent, then build two histograms by counting the number of the coherent and incoherent pixels with each color respectively. The final feature vector is the concatenation of these two histo-

grams. Its main advantage is the combination of color histogram with spatial information, while the main drawback is its high computational cost.

- **Color Correlogram / Color Auto-Correlogram** [Huang *et al.* 1997]: Color correlogram can be understood as a 3-dimensional matrix with size of $(n \times n \times r)$, where $n$ is the number of color bins in an image and $r$ is the maximal distance between two considered pixels. This matrix is indexed by color pairs, where the $k$-th entry for $(i, j)$ specifies the probability of finding a pixel of color $i$ at a distance $k$ away from a pixel of color $j$ in the image. The final feature is obtained by decomposing this matrix into a single vector. As the size of color correlogram is usually too large due to its three dimensions, color auto-correlogram is also proposed to only consider the pair of pixels with the same color $i$ at a distance $k$, thus resulting in a more compact representation. Their advantages are that they integrate the spatial correlation of colors and robustly tolerate large changes in appearance, viewing position and camera zoom. High computational cost is also their main drawback.

There also exist other color features in the literature, such as Dominant Color, Scalable Color, Color Layout, Color Structure, etc. [Manjunath *et al.* 2001].

#### 2.2.1.2 Texture features

Texture is also an important aspect to describe the content of an image. It has no precise definition, but can be intuitively considered as the repeated patterns of local variation of pixel intensities, thereby quantifying the properties such as smoothness, coarseness and regularity in an image.

- **Texture Co-occurrence Matrix** [Tuceryan & Jain 1998]: Gray Level Co-occurrence Matrix (GLCM) is a measurement of how often different combinations of gray level pixel values occur in an image. It estimates image properties of the second order texture statistics by considering the relationship between groups of two neighboring pixels in the image. Given a displacement vector $d = (dx, dy)$, GLCM $P_d$ of size $N \times N$ for $d$ is calculated in such a way that

Table 2.1: Some texture features extracted from gray level co-occurrence matrix (GLCM)

| Texture feature | Formula |
|---|---|
| Energy | $\sqrt{\sum_i \sum_j P_d^2(i,j)}$ |
| Entropy | $-\sum_i \sum_j P_d(i,j) \ln P_d(i,j)$ |
| Contrast | $\sum_i \sum_j (i-j)^2 P_d(i,j)$ |
| Homogeneity | $\sum_i \sum_j \frac{P_d(i,j)}{1+(i-j)^2}$ |

the entry $(i, j)$ of $P_d$ is the occurrence number of the pair of gray levels $i$ and $j$ which are at a distance $d$ apart. Here $N$ denotes the number of gray levels considered in the image. Usually, the matrix $P_d$ is not directly used in an application and a set of more compact features are computed instead from this matrix, as shown in Table 2.1. The main problem of GLCM is that there is no well established method for selecting the optimal displacement vector $d$. In the practice, four displacement vectors are commonly used: $d = (1, 0)$, $d = (0, 1)$, $d = (1, 1)$ and $d = (1, -1)$.

- **Texture Auto-Correlation** [Tuceryan & Jain 1998]: The basic principle of texture auto-correlation is to compare the original image with a shifted one. It measures the coarseness of an image by evaluating the linear spatial relationships between texture primitives. Suppose the displacements according to each axis as $dx$ and $dy$, then the auto-correlation function can be defined as follows:

$$f(dx, dy) = \frac{MN}{(M - dx)(N - dy)} \frac{\sum_{i=1}^{M-dx} \sum_{j=1}^{N-dy} I(i,j) I(i + dx, j + dy)}{\sum_{i=1}^{M} \sum_{j=1}^{N} I^2(i,j)} \quad (2.4)$$

where $M \times N$ is the size of the image and $I(i, j)$ is the gray value of the pixel at position $(i, j)$. It can be noticed that large primitives give rise to coarse texture (e.g. rock surface) and small primitives give rise to fine texture (e.g. silk surface). If texture primitives are large, the auto-correlation will decrease slowly while increasing the distance, whereas it will decrease rapidly if texture consists of small primitives. Moreover, if texture primitives are periodic, then

the auto-correlation will increase and decrease periodically with the distance.

- **Gabor** [Daugman 1988]: Gabor filters (or Gabor wavelets) are widely adopted texture features for image analysis. Basically, Gabor filters are a group of wavelets, with each wavelet capturing energy at a specific frequency and a specific direction. They have been found to be particularly appropriate for texture representation and discrimination because frequency and orientation representations of Gabor filters are similar to those of human visual system. A 2D Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave. Expanding a signal using this basis provides a localized frequency description, therefore capturing local texture properties of the signal. The mean and standard deviation of the transformed coefficients are used to represent the texture feature. Gabor feature has been proven very effective for describing texture [Manjunath & Ma 1996] [Zhang *et al.* 2000], but with disadvantage of high computational complexity because of the substantial convolution, which means it is more suitable for dealing with small images like faces, but will be very time and memory consuming to work on large images, such as natural scenes.

- **Local Binary Patterns** [Ojala *et al.* 2002b]: Local Binary Pattern (LBP) operator was firstly introduced as a complementary measure for local image contrast [Ojala *et al.* 1996], and then becomes a computationally efficient yet powerful feature for texture analysis. The detailed introduction of LBP will be in chapter 4 and 5, since our work presented in these two chapters is based on the LBP feature.

There also exist other texture features in the literature, such as Homogenous Texture, Texture Browsing, etc. [Manjunath *et al.* 2001].

#### 2.2.1.3 Shape features

The shape of an object is also an important clue for recognition, especially for rigid objects. Shape is a geometrical description of the external boundary of an

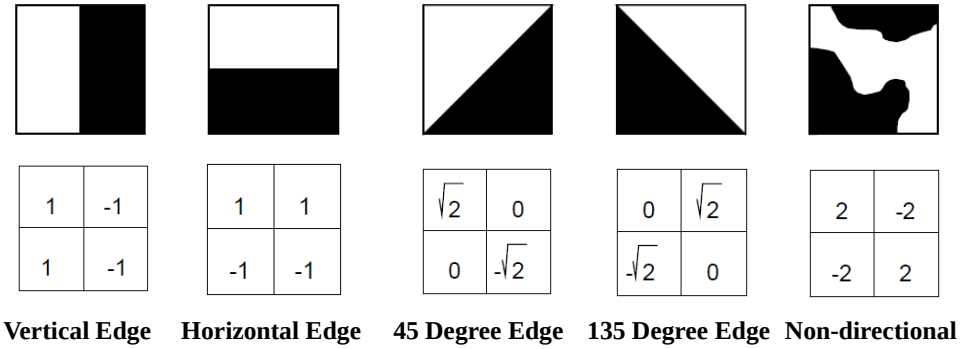



Figure 2.7: Five types of edge and the corresponding filters for edge detection used in edge histogram

object, and can be described by basic geometry units such as points, lines, curves and planes. The popular shape features mainly focus on the edge or contour of an object to capture its shape information.

- **Edge Histogram** [Park *et al.* 2000]: Edge histogram describes edge information with a histogram based on edge distribution in an image. Five types of edges, namely vertical, horizontal, 45-degree diagonal, 135-degree diagonal and non-directional, are considered as shown in Figure 2.7. To compute edge histogram, an image is first divided into $4 \times 4$ non-overlapping blocks, resulting in 16 equal-sized sub-images regardless of the size of the original image. In each of the sub-images, a histogram of edge distribution with 5 bins corresponding to 5 types of edges is computed, leading to a final histogram with $16 \times 5 = 80$ bins after concatenation. An extended version of edge histogram is also proposed by partitioning the image into $4 \times 1$, $1 \times 4$ and $2 \times 2$ sub-images in order to integrate the information of edge distribution in different scales.

- **Line Segments** [Pujol & Chen 2007]: Pujol and Chen proposed line segment based edge feature using Enhanced Fast Hough Transform (EFHT), which is a reliable and computationally efficient way of extracting line segments from an edge image. Once all the line segments are identified by EFHT, line segment based edge feature is extracted as a histogram of line segments' lengths and orientations. In order to obtain the invariant properties for scaling, translation

and rotation, all the lengths are divided by the longest line segment and then an average orientation is computed so that all the angles can be expressed with respect to it. The size of the histogram is determined experimentally and set to 6 bins for orientation and 4 bins for length. Compared to the edge histogram feature, the proposed feature can provide structure information through edge connectivity while still keeping a relatively low computational complexity.

There also exist other shape features in the literature, such as Region Shape, Contour Shape and Shape 3D, which are included in the MPEG-7 standard [1].

The previously introduced global features are all in the form of a single histogram or feature vector, which also keeps the consistent dimensionality regardless of the size of the input image. Therefore, no further modelling methods are required to transform these descriptions.

A comparison of different global features, regarding their category, invariance property and computational cost, is shown in Table 2.2. A detailed comparison of their performances in the context of object recognition is given in Appendix B. The main downside of these global features is their great sensitivity to background clutter, image occlusion, and illumination variations. Moreover, these global methods implicitly assume that the objects of interest should occupy most of the region in images. However, this assumption is hard to be satisfied in real situations, where background noises always exist, particularly in the case that the object of interest is very small compared to the image size. All these limitations make global features gradually give their way to local image features.

### 2.2.2 Local features and corresponding representations

Local image features have received a lot of attention in recent years, and they have already gained the popularity and dominance in object recognition / classification tasks nowadays. Instead of operating on the whole image, the key idea of local features is to extract distinctive information from local image regions centered either

[1] http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm/

Table 2.2: Comparison of the popular global features in the literature (Rotat.=Rotation; Viewp.=Viewpoint; Illum.=Illumination; Inva.=Invariance; Compu.=Computation)

| Feature | Category | Rotat. Inva. | Viewp. Inva. | Illum. Inva. | Compu. Cost |
|---|---|---|---|---|---|
| Color Histogram | Color | Yes | Yes | No | Low |
| Color Moments | Color | Yes | Yes | No | Low |
| Color Coherence Vectors | Color | Yes | Yes | No | High |
| Color (Auto-)Correlogram | Color | Yes | No | No | High |
| Co-occurrence Matrix | Texture | No | No | No | Medium |
| Auto-Correlation | Texture | No | No | No | Low |
| Gabor filters | Texture | No | No | Yes | High |
| Edge Histogram | Shape | No | No | Yes | High |
| Line Segments | Shape | No | No | Yes | Medium |

on some sparse keypoints with certain invariance properties, for instance with respect to scale and viewpoint change, or simply on a dense sampling grid. By this way, local features could be more discriminative and robust to image variations, compared to the global ones. Generally, local feature extraction consists of two main steps: (1) local keypoint/region detection and (2) local descriptor extraction.

#### 2.2.2.1 Local keypoint/region detection

Local features are extracted from local image regions, thus it is important to first detect such regions in a highly repetitive manner. To do so, one could apply certain region detector on images to directly get the output regions. Also, one could first apply certain point detector to get keypoints in images and then fix appropriate regions around these keypoints. There are mainly three strategies for local keypoint/region detection: (1) interest points/regions; (2) dense sampling; and (3) random sampling.

- **Interest Points/Regions**: Interest points are usually keypoints located on edges or corners. Interest regions are usually regions containing a lot of information about image structures like edges and corners, or local blobs with uniform brightness. Many interest point/region detectors have been proposed in the literature: Harris and Stephens [Harris & Stephens 1988] proposed Har-

ris corner detector which is based on the second moment matrix and responds to corner-like features. It is invariant to rotation. Hessian blob detector was proposed by Beaudet [Beaudet 1978] based on the Hessian matrix. It gives strong responses on blobs and ridges because of the second order derivatives. It is also invariant to rotation. Lindeberg [Lindeberg 1998] developed Laplacian blob detector which is scale-invariant, and a blob is defined by a maximum of the normalized Laplacian in scale-space. Harris-Laplace detector [Mikolajczyk & Schmid 2001] was proposed as an extension of the original Harris detector by adding the scale-invariant property. The points are first detected by the scale-adapted Harris function and then selected in scale-space by the Laplacian of Gaussian operator. It is thus invariant to both rotation and scale changes. Another scale-invariant detector is Difference of Gaussian (DoG) proposed by Lowe [Lowe 1999] [Lowe 2004]. DoG is an approximation of the normalized Laplacian scale by calculating differences of Gaussian blurred images at several adjacent local scales. It can also be calculated in a pyramid way which makes it much faster than the Laplacian scale space while keeping comparable results. Harris-Affine and Hessian-Affine detectors [Mikolajczyk & Schmid 2002] [Mikolajczyk & Schmid 2004] were proposed to further extend the scale-invariant detector to obtain invariance against image affine transformations. The affine adaptation is based on the shape estimation properties of the second moment matrix. Maximally Stable Extremal Regions (MSER) [Matas *et al.* 2002] is a watershed-like algorithm based on intensity value connected component analysis of an appropriately thresholded image. The obtained regions are of arbitrary shape and they are defined by all the border pixels enclosing a region, where all the intensity values within the region are consistently lower or higher with respect to the surrounding. There also exist other detectors in the literature such as entropy based salient region detector [Kadir & Brady 2001], edge based region detector (EBR) and intensity based region detector (IBR) [Tuytelaars & Gool 2000] [Tuytelaars & Gool 2004]. The comprehensive review and evaluation of interest point/region detectors can be found

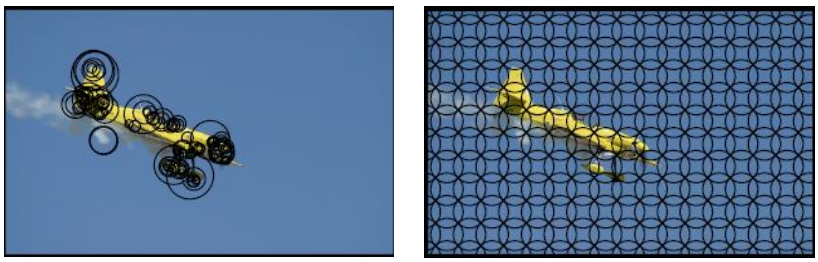



Figure 2.8: Comparison of interest points/regions and dense sampling strategies for local keypoint/region detection (examples from [van de Sande *et al.* 2010])

in [Schmid *et al.* 2000] and [Mikolajczyk *et al.* 2005].

- **Dense Sampling**: Several studies [Winn *et al.* 2005] [Li & Perona 2005] [Agarwal & Triggs 2006] [Furuya & Ohbuchi 2009] have shown experimentally that extracting local features on a dense sampling grid outperforms that of using interest point/region detectors.

- **Random Sampling**: Other studies [Marée *et al.* 2005] [Nowak *et al.* 2006] have proposed to use random sampling strategy for localizing keypoints/regions. As the name implies, keypoints/regions are randomly selected in images for local descriptor extraction.

Figure 2.8 shows the comparison of interest points/regions and dense sampling strategies for local keypoint/region detection. It is worth noticing that combining different strategies may provide further improvements. The winning system of the PASCAL VOC challenge 2007 [Everingham *et al.* 2007] demonstrated that the combination of interest points detector and dense sampling strategy performs clearly better than either of the two separately.

#### 2.2.2.2 Local descriptor extraction

After local keypoint/region detection, the detected regions or local neighborhood around the detected keypoints are described by local image descriptors, which should be discriminative, computationally efficient, and robust against various image vari-

ations such as scaling, affine distortions, viewpoint and illumination changes. Many different local descriptors have been proposed in the literature, and the most popular ones are distribution-based descriptors, which represent region properties by histograms. The most popular local descriptors applied to the domain of object recognition are listed as follows:

- **SIFT** [Lowe 1999] [Lowe 2004]: Lowe proposed Scale Invariant Feature Transform (SIFT), which is a 3D histogram of gradient locations and orientations, as shown in Figure 2.9(a). The location is quantized into a $4 \times 4$ location grid and the gradient angle is quantized into 8 orientations, resulting in a 128-dimensional descriptor. The contributions to the gradient orientations are weighted by the gradient magnitudes and a Gaussian window overlaid over the region, thereby emphasizing the gradients near the region center. SIFT is highly distinctive, in the sense that a single feature can be correctly matched with high probability against a large database of features from many images. Moreover, it is invariant to image scaling and rotation, and also provides robust matching ability across a substantial range of affine distortion, minor viewpoint change, noise disturbance and illumination variance. All these properties ensure its great success in computer vision community, especially for visual object recognition tasks.

- **PCA-SIFT** [Ke & Sukthankar 2004]: Ke and Sukthankar proposed PCA-SIFT, which applies Principal Component Analysis (PCA) technique [Jolliffe 2002] on the normalized gradient patches to enhance the distinctiveness and reduce the dimensionality of the original SIFT. A typical patch is $41 \times 41$ pixels, resulting in a 3042-dimensional vector, which is created by concatenating the horizontal and vertical gradient maps for the patch. The final dimension of the descriptor is reduced to 36 with PCA.

- **Color SIFT** [van de Sande *et al.* 2008] [van de Sande *et al.* 2010]: Van de Sande et al. proposed several color SIFT descriptors by extracting the SIFT feature in different color spaces, including HSV-SIFT, HueSIFT, OpponentSIFT, C-SIFT, RGB-SIFT, rgSIFT and transformed color SIFT. The

SIFT features computed in each individual channel are concatenated as the final color SIFT feature. The aim is to increase the photometric invariance property and the discriminative power of the original SIFT. Their performances were also evaluated and compared in the context of object recognition, and the results demonstrated that combining SIFT with color clues is a promising way to improve the recognition performance.

- **GLOH** [Mikolajczyk & Schmid 2005]: Mikolajczyk and Schmid proposed Gradient Location and Orientation Histogram (GLOH), which can be considered as the extension of the original SIFT to increase its robustness and distinctiveness. GLOH replaces the rectangular location grid used in SIFT with a log-polar one, and applies PCA to reduce the size of the descriptor. The location is divided into 17 bins (3 bins in radial direction and 8 bins in angular direction, the central bin is not divided) and the gradient orientations are quantized into 16 bins, resulting in a 272-dimensional vector. The final dimension of the descriptor is reduced to 128 with PCA.

- **HOG** [Dalal & Triggs 2005]: Dalal and Triggs proposed Histogram of Oriented Gradient (HOG), which is a 3D histogram of gradient locations and orientations. It is similar to both SIFT and GLOH, because it uses both rectangular and log-polar location grids, as shown in Figure 2.9(b). The main difference between HOG and SIFT is that HOG is computed on a dense grid of uniformly spaced cells, with overlapping local contrast normalization. This is for better invariance to illumination and shadowing, and can be done by accumulating a measure of local histogram "energy" over larger spatial blocks and then using the results to normalize all of the sub-images in each block. The standard HOG descriptor is of 36 dimensions.

- **SURF** [Bay *et al.* 2006] [Bay *et al.* 2008]: Bay et al. proposed Speeded-Up Robust Features (SURF), which is inspired by SIFT, but several times faster to compute. Instead of the gradient information in SIFT, SURF computes the Haar wavelet responses, and exploits integral images for computational efficiency. The input region around a keypoint is divided into $4 \times 4$ sub-regions,

within which the sum of the first order Haar wavelet responses in both $x$ and $y$ directions are computed, as shown in Figure 2.9(d). The standard SURF descriptor is of 64 dimensions.

- **Shape Context** [Belongie *et al.* 2002]: Belongie et al. proposed Shape Context, which is also similar to SIFT, but is based on edges. It is a 2D histogram of edge point locations, where the log-polar location grid is used, as shown in Figure 2.9(c). Its aim is to describe the distribution of edge points on a shape with respect to the reference point. The contour of shape can be detected by any edge detector, e.g. Canny edge detector, and edge points are regularly sampled over the whole shape contour. The location is divided into 5 bins in radial direction and 12 bins in angular direction, resulting in a 60-dimensional descriptor.

- **CS-LBP** [Heikkilä *et al.* 2009]: Heikkila et al. proposed Center-Symmetric Local Binary Pattern (CS-LBP) descriptor, which combines the strengths of both SIFT and LBP. It adopts the SIFT-like approach for descriptor construction, but replaces the gradient information used in SIFT with the CS-LBP feature, which is a modified version of the original LBP. Instead of comparing each neighboring pixel with the central one, CS-LBP only compares center-symmetric pairs of pixels, as shown in Figure 2.9(e). This could halve the number of comparisons, and reduce the size of the LBP histogram. The standard CS-LBP applies $4 \times 4$ location grid and 8 neighboring pixels for computation, resulting in a 256-dimensional descriptor.

The attributes of these descriptors are summarized in Table 2.3, including the representation type (sparse or dense), encoded information, spatial pooling scheme (neighborhood grid), computation method (comp.), and dimensionality (dim.). It should be noted that the items in the column of representation type and dimensionality can be changed according to different applications, and the ones listed in the table are directly cited from the original papers. A detailed comparison of some of these descriptors in the context of object recognition is given in Appendix B.

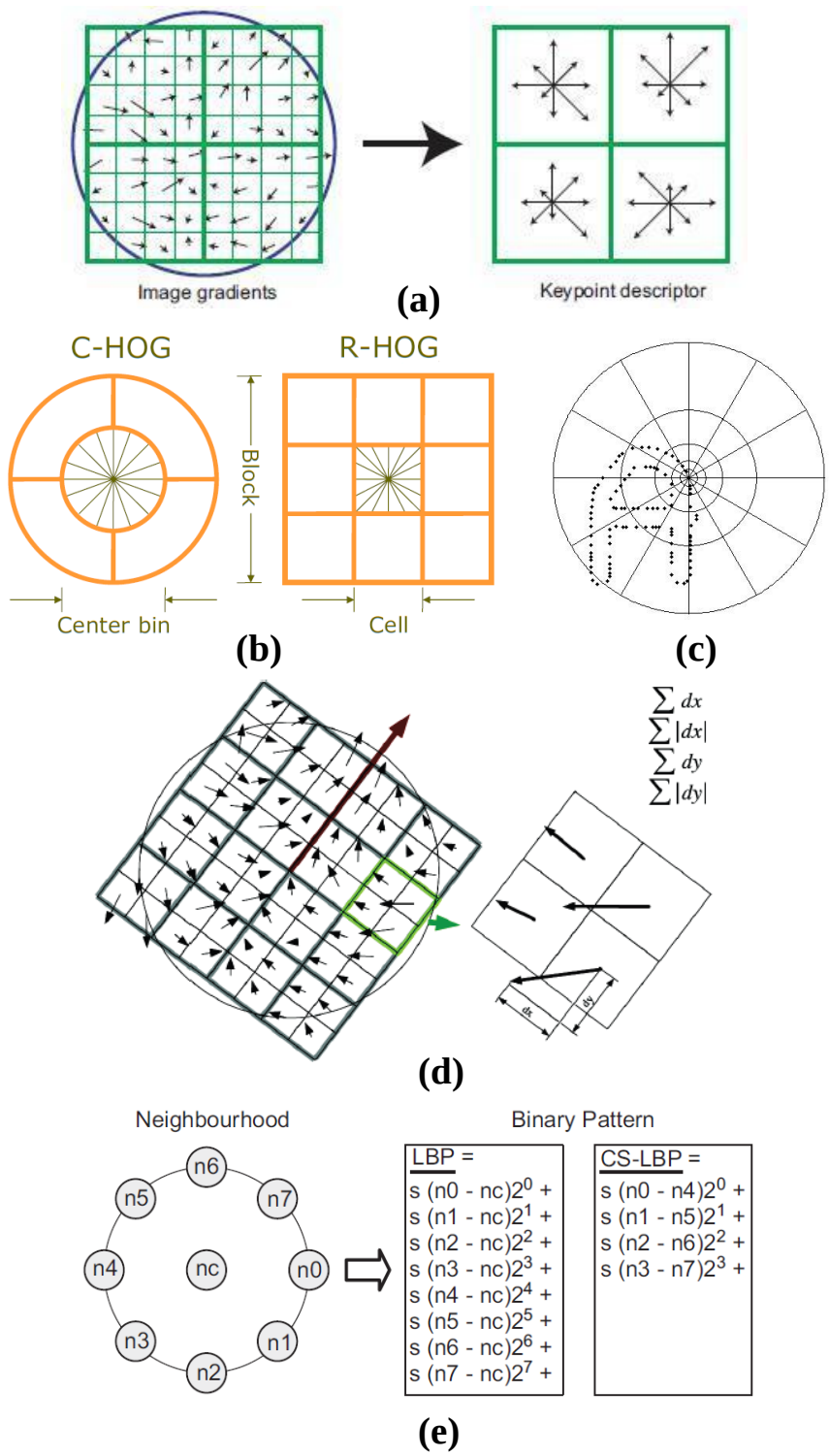



Figure 2.9: Illustrations of popular local image descriptors: (a) SIFT; (b) HOG; (c) Shape Context; (d) SURF; (e) CS-LBP (figures from the original papers)

Table 2.3: Attribute summary of main local image descriptors applied to object recognition in the literature

| Descriptor | Type | Information | Grid | Comp. | Dim. |
|---|---|---|---|---|---|
| SIFT | Sparse | Gradient | Rect. | Distr. | 128 |
| PCA-SIFT | Sparse | Gradient | Rect. | Distr. | 36 |
| Color SIFT | Sparse | Gradient | Rect. | Distr. | 384 |
| GLOH | Sparse | Gradient | Polar | Distr. | 128 |
| HOG | Dense | Gradient | Rect. & Polar | Distr. | 36 |
| SURF | Sparse | Wavelet response | Rect. | Filter | 64 |
| Shape Context | Sparse | Edge points | Polar | Distr. | 60 |
| CS-LBP | Sparse | Binary patterns | Rect. | Distr. | 256 |

In [Brown *et al.* 2011], the authors proposed a framework to learn local descriptors with different combinations of local features and spatial pooling strategies. The previously presented descriptors can thus be incorporated into their framework.

Besides these distribution-based descriptors, there also exist other types of local descriptors such as differential invariants [Koenderink & van Doorn 1987], steerable filters [Freeman & Adelson 1991], complex filters [Schaffalitzky & Zisserman 2002], moment invariants [Gool *et al.* 1996] and so on. Several studies [Mikolajczyk & Schmid 2005] [Zhang *et al.* 2007] [Li & Allinson 2008] [van de Sande *et al.* 2010] have been conducted to comprehensively evaluate and compare the performances of different local image descriptors, and they almost have given the consistent conclusions that distribution-based local descriptors perform the best, and therefore have been widely applied to the tasks of object recognition.

After local feature extraction, each image is represented by a set of local descriptors. It is unreasonable to feed them directly into a classifier. On one hand, the dimensions of these descriptors are relatively high because of the large number of keypoints/regions (normally around thousands) in images. On the other hand, the number of local descriptors in each image varies because the number of keypoints/regions changes from one image to another one. Thus, an efficient feature modelling method is required to transform these high dimensional and variable numbers of local descriptors into a more compact, informative and fixed-length repre-

sentation for further classification. Two leading modelling methods in the literature are Bag-of-Features (BoF) and Gaussian Mixture Model (GMM).

#### 2.2.2.3 Bag-of-Features (BoF) representation: discrete distribution

The "Bag-of-Features" (BOF) method (also called "Bag-of-Visual-Words" (BoVW)) [Sivic & Zisserman 2003] [Csurka *et al.* 2004] models an image as a discrete distribution. Its main idea is adapted from the "Bag-of-Words" (BoW) representation [Salton & McGill 1983] [McCallum & Nigam 1998] in text classification domain, and is to represent an image as an orderless collection of local descriptors based on an intermediate representation called "visual vocabulary". More precisely, it consists of two main steps: (1) visual vocabulary construction and (2) histogram encoding. A visual vocabulary is first constructed by applying a clustering algorithm on the training data, and each cluster center is considered as a "visual word" in the vocabulary. All the descriptors extracted from an image are then quantized to their closest visual word (hard assignment) or several close visual words (soft assignment) in an appropriate metric space by a certain encoding method. The number of the descriptors assigned to each visual word is accounted into a histogram as the final BoF representation. In other words, each image is characterized by a histogram of visual words frequencies. Figure 2.10 shows an illustration of this process. Some representative methods for each of these two steps are introduced in the following. As the BoF method discards all spatial information between the extracted local features, some approaches which reuse this useful information are also presented.

**Visual vocabulary construction** The visual vocabulary is constructed offline on the training data by unsupervised or supervised learning methods. The $k$-means clustering algorithm [MacQueen 1967] is the most popular one. It is an unsupervised clustering algorithm which proceeds by iterated assignments of points to their closest cluster centers and re-computation of the cluster centers. The number of the cluster centers $k$ is predefined empirically. The advantage of $k$-means is its simple and efficient implementation, while its drawback is that most of the cluster centers are drawn irresistibly towards dense regions of the sample distribution

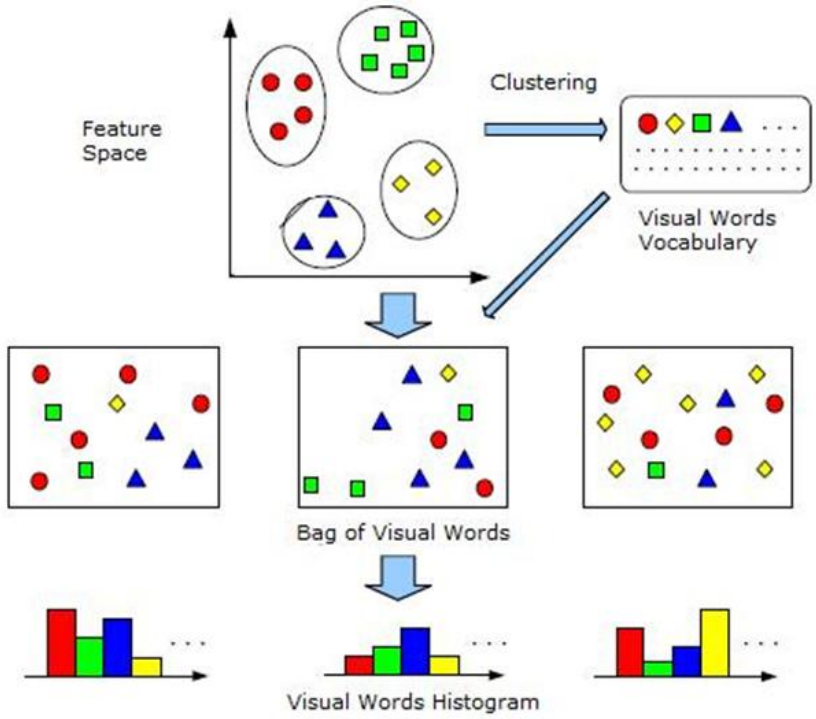



Figure 2.10: An illustration of the "Bag-of-Features" ("Bag-of-Visual-Words") method (example from [Yang *et al.* 2007])

which do not necessarily correspond to discriminative ones. [Jurie & Triggs 2005] proposed a radius-based clustering, which avoids setting all cluster centers into high density areas and assigns all features within a fixed radius of $r$ to one cluster. [Wu & Rehg 2009] proposed to use one-class SVM and the Histogram Intersection Kernel (HIK) instead of the popular Euclidean distance for clustering.

A drawback of the universal visual vocabulary generated by the unsupervised approaches is its deficient discriminative power due to the ignorance of category information. To address this problem, some studies departed from the idea of having one universal vocabulary for all the training data from the whole set of categories. In [Farquhar *et al.* 2005] [Zhang *et al.* 2007], category specific vocabularies were trained and agglomerated into a single vocabulary. Although substantial improvements were obtained, these approaches are impractical for a large number of categories as the size of the agglomerated vocabulary and the corresponding histogram representation grows linearly with the number of categories. Therefore, a compact visual vocabulary is preferred to provide a lower-dimensional representa-

tion and effectively avoid these difficulties. [Winn *et al.* 2005] [Fulkerson *et al.* 2008] [Lazebnik & Raginsky 2009] made use of the mutual information between the features and the categories to reduce the size of visual vocabulary without sacrificing its discriminative power. [Moosmann *et al.* 2006] proposed an efficient alternative, in which training examples are recursively divided using a randomized decision forest and the splits in the decision trees are the comparisons of a descriptor dimension to a threshold. [Perronnin *et al.* 2006] characterized images using a set of category specific histograms, where each histogram describes whether the content can be best modeled by the universal vocabulary or by its corresponding category vocabulary.

Another group of methods [Vogel & Schiele 2004] [Yang *et al.* 2008] [Liu *et al.* 2009] claimed that the semantic relations between features are useful for classification and attempted to bring the semantic information into visual vocabulary construction. In [Vogel & Schiele 2004], a semantic vocabulary was constructed by manually associating local image regions to certain semantic concepts such as "stone", "sky", "grass" and so on. However, the fact that it requires huge manual labor for labeling local image regions among large amount of training data makes it impractical in such cases. [Yang *et al.* 2008] proposed to unify the process of visual vocabulary generation and classifier training, and to encode an image by a sequence of visual bits which capture different aspects of image features and constitute the semantic vocabulary. The method proposed by [Liu *et al.* 2009] can automatically learn a semantic visual vocabulary using diffusion maps which capture the semantic and geometric relations of feature space.

**Histogram encoding** Once a visual vocabulary is constructed, a feature encoding method is needed to assign local descriptors to the visual words and characterize the visual content of an image by a histogram of visual words frequencies. Generally, there are two strategies for histogram encoding: (1) hard assignment and (2) soft assignment.

Hard assignment simply assigns the extracted local feature vectors to their single best (usually the nearest) visual word respectively, according to a certain distance

measure, as shown in equation (2.5):

$$HA(\omega) = \frac{1}{N}\sum_{n=1}^{N}\begin{cases}1 & \text{if } \omega = \arg\min_{v\in V}(D(v, r_n)) \\ 0 & \text{otherwise}\end{cases} \tag{2.5}$$

where $\omega$ is a visual word in the vocabulary $V$, $N$ is the number of local regions in an image, $r_n$ is the feature vector extracted from the $n$-th local region, and $D(v, r_n)$ is the distance between $r_n$ and each visual word $v$. The advantages of hard assignment include its computational simplicity and the fact that it leads to a sparse histogram. However, problems could occur for feature vectors located in ambiguous areas. In [van Gemert *et al.* 2008] [van Gemert *et al.* 2010], two different issues are considered: word uncertainty and word plausibility. Word uncertainty refers to the problem of selecting the correct visual word out of two or more relevant candidates, while word plausibility denotes the problem of selecting a visual word without any suitable candidate in the vocabulary, as illustrated in Figure 2.11. Soft assignment is thus proposed to address these issues.

There are two kinds of approaches for soft assignment. The first one consists in performing probabilistic clustering using typically a Gaussian Mixture Model (GMM) [Farquhar *et al.* 2005, Winn *et al.* 2005, Perronnin *et al.* 2006], and each feature vector contributes to multiple visual words according to its posterior probability of the Gaussian given each visual word. Although these works are able to deal with word uncertainty by considering multiple visual words, they ignore word plausibility. On the contrary, [Boiman *et al.* 2008] copes with word plausibility by using the distance to the single best neighbor in feature space without taking into account word uncertainty. [van Gemert *et al.* 2008] [van Gemert *et al.* 2010] made the assignment using a decreasing function of the Euclidean distance between feature vectors and word centroids, paired with a Gaussian kernel:

$$G_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp(-\frac{1}{2}\frac{x^2}{\sigma^2}) \tag{2.6}$$

where $\sigma$ is the smoothing parameter of kernel $G$. Three different formula were

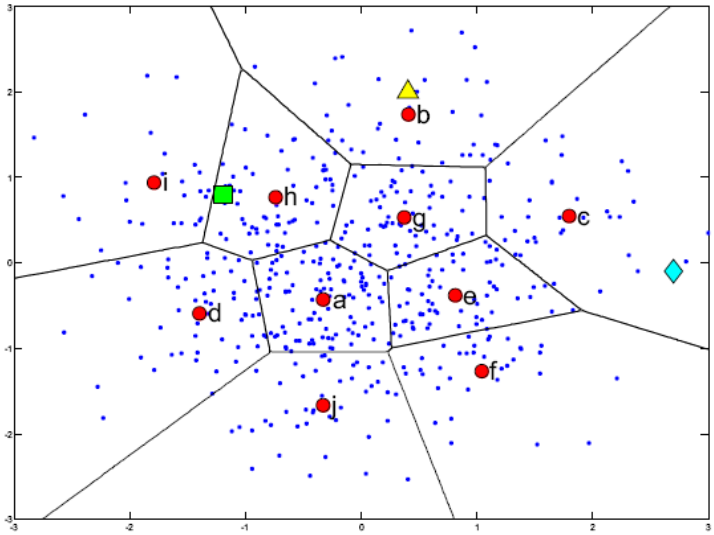



Figure 2.11: Illustration of visual word uncertainty and plausibility. The small dots represent image features, the labeled red circles are visual words found by unsupervised clustering. The triangle represents a data sample that is well suited to hard assignment approach. The difficulty with word uncertainty is shown by the square, and the problem of word plausibility is illustrated by the diamond. (example from [van Gemert *et al.* 2008])

proposed to cope with word uncertainty (UNC), word plausibility (PLA) and both of them (KCB) respectively:

$$UNC(\omega) = \frac{1}{N}\sum_{n=1}^{N}\frac{G_\sigma(D(\omega, r_n))}{\sum_{k=1}^{|V|} G_\sigma(D(v_k, r_n))} \tag{2.7}$$

$$PLA(\omega) = \frac{1}{N}\sum_{n=1}^{N}\begin{cases} G_\sigma(D(\omega, r_n)) & \text{if } \omega = \arg\min_{v\in V}(D(v, r_n)) \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

$$KCB(\omega) = \frac{1}{N}\sum_{n=1}^{N} G_\sigma(D(\omega, r_n)) \tag{2.9}$$

Recently, several new encoding methods, such as locality-constrained linear encoding [Wang *et al.* 2010], improved Fisher encoding [Perronnin *et al.* 2010], and super vector encoding [Zhou *et al.* 2010], have been proposed to improve on the standard histogram of quantized local features, and have reported very good results on the tasks of object recognition and image classification. A compara-

tive analysis and evaluation of these different encoding methods can be found in [Chatfield *et al.* 2011].

**Spatial information** The BoF method views images as orderless distributions of local image features, thus losing at the same time all the spatial relationships between these local features. However, we know intuitively that spatial information is important for image classification. Therefore, [Lazebnik *et al.* 2006] proposed the "spatial pyramid" method in order to take into account the spatial information of local features, inspired by pyramid match kernels introduced in [Grauman & Darrell 2005b] which build pyramid in feature space while discarding the spatial information. The "spatial pyramid" method consists of performing pyramid matching in two-dimensional image space and using the traditional clustering techniques in feature space.

Suppose we have $M$ types of features and each of them provides two sets of two-dimensional vectors, $X_m$ and $Y_m$, representing the coordinates of features of type $m$ found in the respective image. Then the final kernel is the sum of the separate kernels:

$$K^L(X, Y) = \sum_{m=1}^{M} \kappa^L(X_m, Y_m) \tag{2.10}$$

where $\kappa^L(X_m, Y_m)$ is the pyramid match kernel of feature type $m$. This approach has the advantage of maintaining continuity with the BoF paradigm. In fact, it reduces to a standard BoF method when $L = 0$. Figure 2.12 shows an example of constructing a three-level spatial pyramid.

The winning system [van de Sande *et al.* 2010] for object classification task in the PASCAL VOC Challenge [Everingham *et al.* 2010] provided some modifications of the standard "spatial pyramid" method. An image is first divided into $1 \times 1 + 2 \times 2 + 1 \times 3$ spatial levels, as shown in Figure 2.13, one unique vocabulary is then constructed for the whole image, and the BoF representations are computed using this vocabulary for each spatial level, which are fused later using the extended Gaussian kernel.

Another work [Marszalek & Schmid 2006] exploits spatial relations between fea-

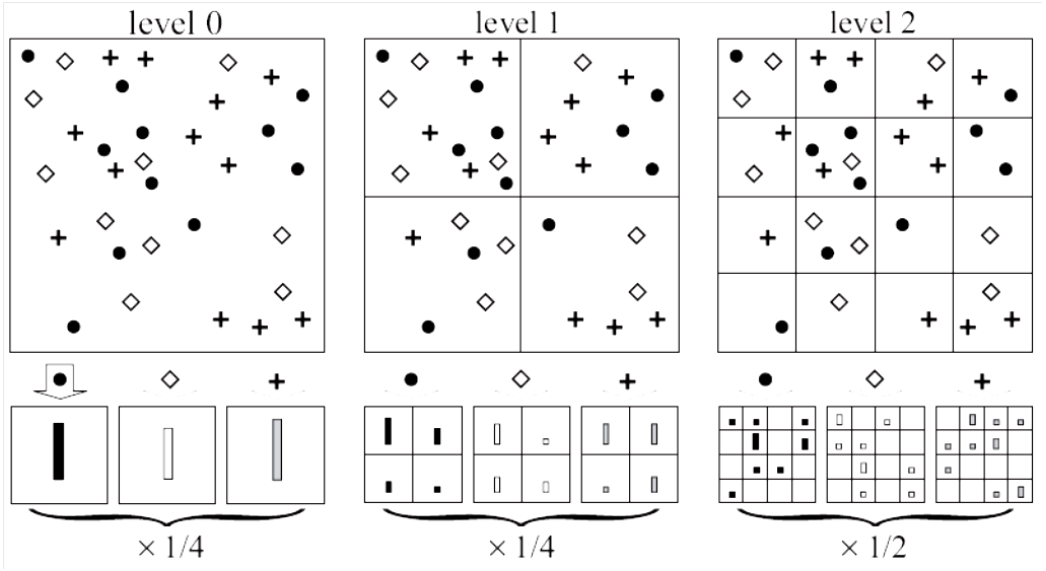



Figure 2.12: An example of constructing a three-level spatial pyramid. The image has three feature types, indicated by circles, diamonds, and crosses. At the top, the image is subdivided at three different levels of resolution. Next, for each level of resolution and each channel, the features that fall in each spatial bin are counted. Finally, each spatial histogram is weighted according to its level. (example from [Lazebnik *et al.* 2006])

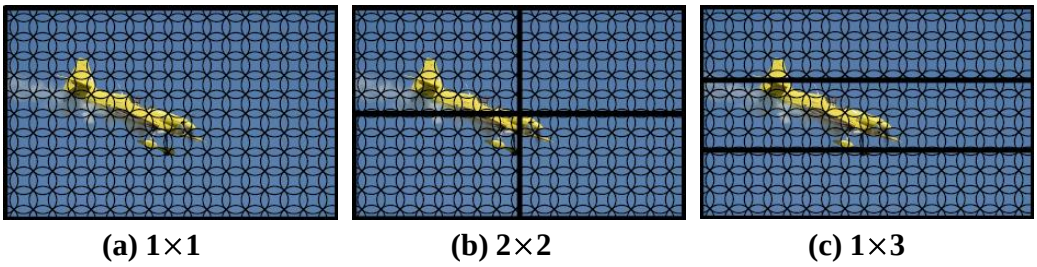



Figure 2.13: The spatial pyramid used in the winning system for object classification task in the PASCAL VOC Challenge (example from [van de Sande *et al.* 2010])

tures by making use of object boundaries provided during supervised training. They boost the weights of features that agree on the position and shape of the object and reduce the weights of background features, thus suitable to solve the problem of background clutter.

The BoF method effectively provides a mid-level representation which helps to bridge the semantic gap between low-level features extracted from an image and high-level concepts to be categorized. Its main limitation is the assumption that the distribution of feature vectors in an image can be known a priori. The optimal size of visual vocabulary, which is the basis of this approach, is also hard to be fixed.

**Bag-of-Regions** Recently, the Bag-of-Regions (BoR) representation has been proposed and applied on several different applications such as object recognition [Gu *et al.* 2009], image retrieval [Hu *et al.* 2011] [Vieux *et al.* 2012] and scene classification [Gokalp & Aksoy 2007]. The BoR approach extends the classical BoF method to be based not only on keypoint-based descriptors, but also on the features extracted from image regions. After region extraction by an image segmentation algorithm, a vast amount of different visual features could be computed from image regions, such as color, texture and shape, as introduced in section 2.2.1. Then, visual vocabulary construction and histogram encoding are performed by following the way similar to the BoF method. The final frequency histogram is used as the representation of an image. An example of the BoR representation is shown in Figure 2.14.

The BoR representation aims at using image regions because they have some pleasant properties: (1) they encode shape and scale information of objects naturally; (2) they specify the domains on which to compute various features, without being affected by clutter from outside the region [Gu *et al.* 2009]. However, the bottleneck of this approach lies in the difficulty of choosing a good image segmentation algorithm for region extraction, because image segmentation itself is still a very challenging problem and the results are not always satisfactory.

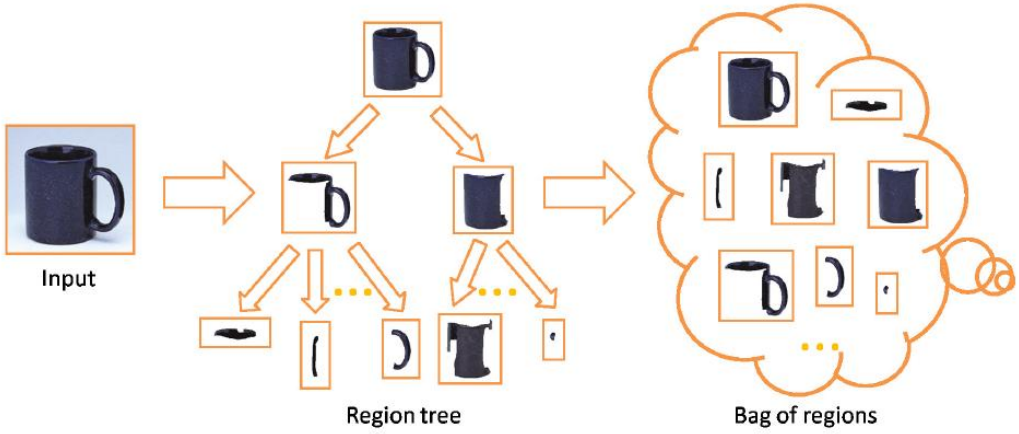



Figure 2.14: An example of the BoR representation (from [Gu *et al.* 2009])

#### 2.2.2.4 Gaussian Mixture Model (GMM) representation: continuous distribution

The Gaussian Mixture Model (GMM) method models an image as a continuous distribution. [Moreno *et al.* 2003] and [Farquhar *et al.* 2005] proposed to model an image as a single Gaussian distribution with full covariance. However, the monomodal assumption is generally too restrictive. Therefore, [Goldberger *et al.* 2003] [Vasconcelos 2004] [Vasconcelos *et al.* 2004] proposed to model an image as a mixture of Gaussian distributions, generally with diagonal covariance. Formally, a GMM is in the form:

$$\begin{aligned} p(x) &= \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \\ &= \sum_{k=1}^{K} \pi_k \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma_k|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)\right] \end{aligned} \tag{2.11}$$

where $\mu_k$ and $\Sigma_k$ are respectively mean and covariance of the $k$-th component of a GMM which contains a total of $K$ Gaussians, and $D$ is the dimensionality of the data. The parameters $\pi_k$ are called mixing coefficients and must satisfy:

$$0 \leq \pi_k \leq 1 \quad \text{together with} \quad \sum_{k=1}^{K} \pi_k = 1 \tag{2.12}$$

The GMM method has two main shortcomings. Firstly, the robust estimation of the GMM parameters may be difficult as the cardinality of the vector set is small. Secondly, it is expensive to compute the similarity between two GMMs. Therefore, we choose the BoF method for image modelling in our work presented in the following chapters.

## 2.3 Image classification

In order to perform the final classification based on image representations computed from the extracted features, certain pattern recognition algorithms (classifiers) are required. There exist two main kinds of approaches in the literature for making the final classification: (1) generative methods and (2) discriminative methods.

Generative methods produce a probability density model over all the variables and then adopt it to compute classification functions. Differently, discriminative methods directly estimate the posterior probabilities for classification without attempting to model the underlying probability distributions.

### 2.3.1 Generative methods

Suppose that $x$ is the set of features representing an image to be classified, and $C_m, m = 1, \ldots, M$ are a set of class labels, generative methods estimate the posterior probability $p(C_m|x)$ in a probabilistic framework, according to which $x$ will be classified into the target class. For instance, if we wish to minimize the number of misclassifications, $x$ will be assigned to the class with the largest posterior probability. According to the Bayes theorem, the posterior probability $p(C_m|x)$ can be expressed in the following form:

$$p(C_m|x) = \frac{p(x|C_m)p(C_m)}{p(x)} \tag{2.13}$$

where $p(C_m)$ is the prior probability of the class $C_m$, $p(x|C_m)$ is the probability density (also called likelihood) of the class $C_m$, and $p(x)$ is the probability density over all the classes. As $p(x)$ stays constant when considering the posterior probability

for each class, its computation is not necessary. Moreover, if we know that the prior probabilities are equal, or if we make this assumption, the decision can be realized only depending on the likelihood function $p(x|C_m)$ for each class.

The typical generative method relies on a GMM to model the distribution of the training samples. The set of the GMM parameters can be efficiently learned by using the Expectation Maximization (EM) algorithm. If we consider a GMM for modeling the specific class $C_m$, then the logarithm of the likelihood function is given by:

$$\begin{aligned} \ln(p(x|C_m)) = \ln(p(x|\mu, \Sigma, \pi)) &= \ln \prod_{n=1}^{N} \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k) \right\} \\ &= \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k) \right\} \end{aligned} \tag{2.14}$$

where $N$ is the number of feature vectors in $x$. Then, we can employ the EM algorithm to maximize this likelihood function for the class $C_m$ with respect to the parameters of the GMM, according to the following steps:

1. Initialize all the parameters and compute the initial value of the logarithm of the likelihood function.

2. **Expectation step (E-step):** Calculate the expected value of the logarithm of the likelihood function under the current estimation of the parameter values:

$$\gamma_n^k = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_k \mathcal{N}(x_n|\mu_j, \Sigma_j)} \tag{2.15}$$

3. **Maximization step (M-step):** Re-estimate all the parameters:

$$\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k x_n \tag{2.16}$$

$$\Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k (x_n - \mu_k^{new})(x_n - \mu_k^{new})^T \tag{2.17}$$

$$\pi_k^{new} = \frac{N_k}{N} \tag{2.18}$$

where $N_k = \sum_{n=1}^{N} \gamma_n^k$.

4. Evaluate the logarithm of the likelihood function $\ln(p(x|\mu, \Sigma, \pi))$ and check for convergence of either the parameters or the logarithm of the likelihood. If the convergence criterion is not satisfied, return to step 2.

After the optimized GMMs for all the classes are obtained, each new sample will be assigned to the class with the maximum value of the logarithm of the likelihood function.

Generative methods offer the advantage of easily adding new classes or new data for a certain class by training the model only for the concerned class rather than for all the classes. It can also deal with the situation of incomplete data. Its main drawback lies in high computational cost of learning process.

### 2.3.2 Discriminative methods

The objective of discriminative methods is to learn the precise boundaries between different classes of samples in a multi-dimensional space (usually the feature space) so that the classification can be performed by considering the position of the image projection in this space. Many discriminative classifiers are reported in the literature, and the kernel-based ones are the most popular.

#### 2.3.2.1 Support Vector Machines (SVM)

Among all the kernel-based discriminative classifiers, the Support Vector Machines (SVM) proposed by Vanpik [Cortes & Vapnik 1995] based on his statistical learning theory [Vapnik 1995] is the most famous and popular. SVM constructs a hyperplane in a high or infinite dimensional space to linearly separate the samples from different classes for classification. A good separation is achieved by constructing the hyperplane that has the maximum distance (margin) to the nearest training data samples of any class. Generally, the larger is the margin, the lower the generalization error of the classifier is. An example of good separation hyperplane is illustrated in Figure 2.15. New samples are then mapped into the same space and predicted to a class based on which side of the hyperplane they fall into.

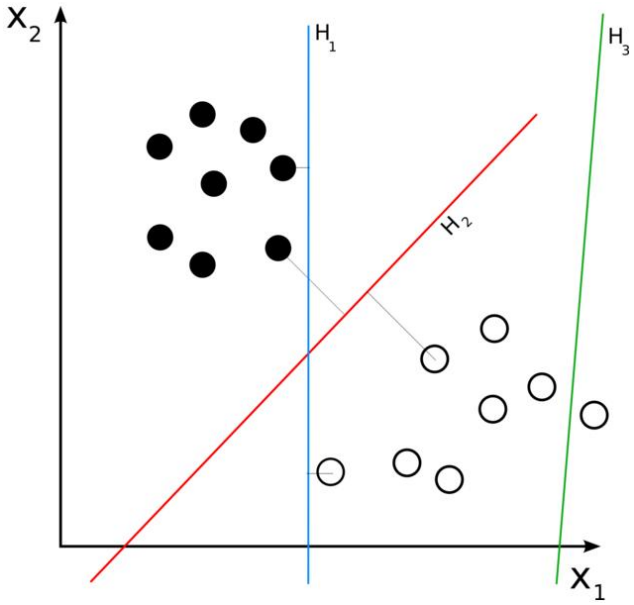



Figure 2.15: An illustration of different hyperplanes: H3 does not separate two classes; H1 does separate two classes, but with a small margin; H2 separates two classes with the maximum margin.

**Linear SVM** The standard SVM is a linear classifier for binary classification problem. Given a set of $N$ labelled training samples $(x_i, y_i), i = 1, \ldots, N$, where $x_i \in R^D$ are the feature vectors representing the samples with $D$ dimensions while $y_i \in \{-1, 1\}$ are the sample labels, SVM constructs a $D-1$-dimensional hyperplane with the maximum margin in the feature space to linearly separate these samples into two predefined classes, as illustrated in Figure 2.16, by solving the following optimization problem:

$$\begin{aligned} &\min_{\omega,b,\xi} \left\{ \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{N} \xi_i \right\} \\ \text{subject to} \quad & y_i(\omega \cdot x_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, N \\ & \xi_i \geq 0. \end{aligned} \tag{2.19}$$

where $\omega$ is the normal vector of the hyperplane, $b$ determines the offset of the hyperplane from the origin along the normal vector $\omega$, $\xi_i$ are slack variables which measure the degree of misclassification of the datum $x_i$, and $C$ is the penalty parameter of the error term which controls the penalty level of the misclassified samples.

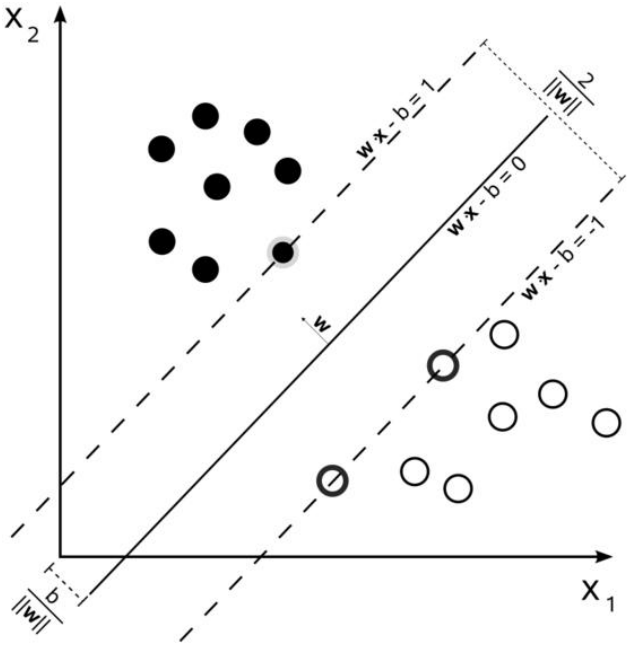



Figure 2.16: An illustration of maximum-margin hyperplane for an SVM trained with samples from two classes (samples on the margins are called the support vectors)

For a new sample $x$ to be classified, the final decision function is in the form:

$$f(x) = \text{sgn}\left\{\sum_{i=1}^{N} \alpha_i^* y_i (x_i \cdot x) + b^*\right\} \tag{2.20}$$

where $\alpha_i^*$ and $b^*$ are the optimized parameters obtained in the training process.

**Non-linear SVM** The original classification problem for the standard SVM is stated in a finite dimensional space (usually the feature space). However, it often happens that the samples to be classified are not linearly separable in the original space. For this reason, the non-linear SVM was proposed to map the samples from the original finite dimensional space into a higher or infinite dimensional space, in which these samples are supposed to be linear and the separation of them is much easier than in the original space. To keep the computational cost reasonable, the mapping used by the non-linear SVM is designed to ensure that the dot products of the samples in the mapped space can be easily computed in terms of a kernel function $K(*,*)$ in the original space.

For the training of the non-linear SVM classifier, the optimization problem in the linear SVM training as equation 2.19 is changed as:

$$\begin{aligned} &\min_{\omega,b,\xi} \left\{ \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{N}\xi_i \right\} \\ \text{subject to} \quad & y_i(\omega \cdot \phi(x_i) + b) \geq 1 - \xi_i, \quad i = 1, \ldots, N \\ & \xi_i \geq 0. \end{aligned} \tag{2.21}$$

where the training samples $x_i$ are mapped into a higher or infinite dimensional space by the mapping function $\phi$.

The final decision function for a new sample $x$ is thus changed as:

$$f(x) = \text{sgn} \left\{ \sum_{i=1}^{N} \alpha_i^* y_i K(x_i, x) + b^* \right\} \tag{2.22}$$

where

$$K(x_i, x) = \phi(x_i)^T \phi(x) \tag{2.23}$$

The kernel function $K(*,*)$ in equation (2.22) is a very important factor for the non-linear SVM to achieve a good classification performance. The choice of this kernel function and the tuning of its parameters will directly impact the final results. Unfortunately, to the best of our knowledge, the selection of kernels for a certain application is until now generally done empirically and experimentally, or by cross-validation in some cases. The commonly used kernel functions will be introduced in section 2.3.3.

**Multi-class SVM** The standard SVM is a binary classifier, whereas many classification problems involve multiple classes. Two common strategies are designed to extend SVM for dealing with multi-class problems: (1) one-against-all and (2) one-against-one. The "one-against-all" strategy constructs one SVM binary classifier for each class by taking the samples in the considered class as the positive samples and all the other samples as the negative ones. The "one-against-one" strategy constructs one SVM binary classifier for each pair of the classes, and the final classification is

done in a max-wins voting way: every classifier assigns the sample to one of the two classes, and the vote for the assigned class is then increased by one, and the sample is finally classified to the class with the most votes. Such strategy is adopted in C-SVC of the popular LibSVM implementation [Chang & Lin 2001].

#### 2.3.2.2 Multiple Kernel Learning (MKL)

The SVM classifier only uses single kernel for solving learning problems. Recently, some studies [Lanckriet *et al.* 2004] [Yang *et al.* 2009b] [Vedaldi *et al.* 2009] have demonstrated the effectiveness of using multiple kernels instead of a single one for improving the classification performance.

The combination of multiple kernels is defined as follows:

$$\begin{aligned} K(x_i, x) &= \sum_{m=1}^{M} \beta_m K_m(x_i, x) \\ \text{with} \quad \beta_m &\geq 0, \ \sum_{m=1}^{M} \beta_m = 1 \end{aligned} \tag{2.24}$$

where $M$ is the total number of kernels, and $\beta_m$ is the weight for each kernel which is optimized during the training process. Each basis kernel $K_m$ can either be different kernels with different parameter configurations or kernels computed from different sets of features. Therefore, MKL can also be interpreted as a kind of fusion technique in certain sense. The final decision function of MKL is in the following form, which is similar to the one of SVM except the combined kernels:

$$f(x) = \text{sgn} \left\{ \sum_{i=1}^{N} \alpha_i^* y_i \sum_{m=1}^{M} \beta_m K_m(x_i, x) + b^* \right\} \tag{2.25}$$

where $\alpha_i^*$ and $b^*$ are the optimized parameters obtained in the training process. Here $\alpha_i^*$ and $\beta_m$ can be learned in a joint optimization problem as in [Bach *et al.* 2004] [Rakotomamonjy *et al.* 2008].

An extension of the precedent simple MKL is presented in [Yang *et al.* 2009b] and called the Group-Sensitive MKL (GS-MKL). An intermediate notion of "group" between object categories and individual images has been introduced to the MKL

framework to seek a trade-off between capturing the diversity and keeping the invariance for each class in the training process. In GS-MKL, the weight of each kernel $\beta_m$ depends not only on the corresponding kernel functions, but also on the "groups" that two compared images belong to. Thus, the combined kernel in equation (2.24) and the final decision function in equation (2.25) are respectively rewritten as:

$$K(x_i, x) = \sum_{m=1}^{M} \beta_m^{c(x_i)} \beta_m^{c(x)} K_m(x_i, x) \tag{2.26}$$

$$f(x) = \text{sgn}\left\{ \sum_{i=1}^{N} \alpha_i^* y_i \sum_{m=1}^{M} \beta_m^{c(x_i)} \beta_m^{c(x)} K_m(x_i, x) + b^* \right\} \tag{2.27}$$

where $c(x_i)$ and $c(x)$ are the group indices of the sample $x_i$ and $x$ respectively.

Although GS-MKL is shown to be very effective for image classification according to the experiments on several datasets [Yang *et al.* 2009b], the optimal way to get the group index for each image remains debatable. The authors applied some clustering methods, namely $k$-means [MacQueen 1967] and probabilistic Latent Semantic Analysis (pLSA) [Hofmann 1999], to get a set of groups whose number is manually defined. It remains unclear how to choose the optimal number of groups and the corresponding clustering method.

#### 2.3.2.3 Other typical classifiers

Besides the kernel-based classifiers, we briefly present here several other typical discriminative classifiers.

- **Multilayer Perceptron** [Rosenblatt 1962]: It is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate output. It consists of multiple layers of nodes in a directed graph which is fully connected from one layer to the next. The back-propagation technique is usually used for training the network.

- **Decision Tree** [Quinlan 1986] [Quinlan 1993]: It is a classifier in the form of a tree structure, where each node is either a leaf node which indicates the class of samples, or a decision node which specifies some test to be carried out on a

single attribute value, with one branch and sub-tree for each possible outcome of the test. There are a variety of algorithms for building decision trees, such as ID3 [Quinlan 1986] and C4.5 [Quinlan 1993].

- **K-Nearest Neighbors** [Cover & Hart 1967]: It is an instance-based learning algorithm which classifies a sample by calculating the distances between this sample and the samples in the training set. Then, it assigns this sample to the class that is most common among its k-nearest neighbors.

- **Adaboost** [Freund & Schapire 1997]: It calls a weak classifier repeatedly in a series of rounds $t = 1, \ldots, T$. For each round, a weak classifier is forced to focus on the samples incorrectly classified by the previous weak classifier through increasing the weights for these hard samples. Finally, a strong classifier can be created by linearly combining these weak classifiers.

In conclusion, discriminative methods and generative methods are two different ways for classification. Given an observed variable $x$ and an unobserved variable $y$, discriminative methods model the conditional probability distribution $P(y|x)$, while generative methods model their joint distribution $P(x, y)$. For tasks such as classification or regression that do not require the joint distribution, discriminative methods generally yield superior performance. Moreover, discriminative methods are less computationally expensive than generative methods. Therefore, we adopt discriminative methods, in particular SVM and MKL, to perform classification in our experiments presented in the following chapters.

### 2.3.3 Similarity measurement between images

An important factor for image classification is how to measure the similarities between images. The resulting kernels are also important for the performance of the kernel-based discriminative classifiers such as SVM and MKL. According to different image representations, the similarity measurement between images can be divided into 3 categories: (1) kernel functions for model-free approaches; (2) kernel functions for discrete models; and (3) kernel functions for continuous models.

#### 2.3.3.1 Kernel functions for model-free approaches

The model-free approaches directly measure the similarity between two unordered feature sets. Assume that we have two feature sets $X = x_i, i = 1, \ldots, T_X$ and $Z = z_j, j = 1, \ldots, T_Z$. The simplest approach to define a similarity measurement between such two sets is the sum of the similarities between all possible pairs of feature vectors. Let $k(*, *)$ be a Positive Semi-Definite kernel (PSD), the summation kernel [Haussler 1999] is defined as:

$$K_S(X, Z) = \frac{1}{T_X}\frac{1}{T_Z}\sum_{i=1}^{T_X}\sum_{j=1}^{T_Z} k(x_i, z_j) \tag{2.28}$$

However, its discriminative ability is compromised as all possible matchings between features are combined with equal weights. The good matchings could be easily swamped by the bad ones.

[Wallraven *et al.* 2003] and [Boughorbel *et al.* 2004] both proposed a matching kernel that only considered the similarities of the best matched local features:

$$K_M(X, Z) = \frac{1}{2}\left[\frac{1}{T_X}\sum_{i=1}^{T_X}\max_{j=1,\ldots,T_Z} k(x_i, z_j) + \frac{1}{T_Z}\sum_{j=1}^{T_Z}\max_{i=1,\ldots,T_X} k(z_j, x_i)\right] \tag{2.29}$$

Unfortunately, the "max" operator makes this kernel non-Mercer (not PSD).

Lyu [Lyu 2005] proposed a Mercer kernel to quantify the similarities between feature sets. The kernel is a linear combination of the $p$-exponentiated kernels between local features:

$$K(X, Z) = \frac{1}{T_X}\frac{1}{T_Z}\sum_{i=1}^{T_X}\sum_{j=1}^{T_Z}[k(x_i, z_j)]^p \tag{2.30}$$

$p$ is the kernel parameter and $p > 1$ gives more influence to good matchings.

The Earth Mover's Distance (EMD) [Rubner *et al.* 2000] is a similarity measurement between feature sets and aims at finding an optimal matching that would be

required to transform one set into the other. It is defined as:

$$EMD = \max_{f_{ij}, i=1,\ldots,T_X, j=1,\ldots,T_Z} \sum_{i=1}^{T_X} \sum_{j=1}^{T_Z} k(x_i, z_j) f_{ij} \tag{2.31}$$

subject to the following constraints:

$$f_{ij} \geq 0 \tag{2.32}$$

$$\sum_{i=1}^{T_X} f_{ij} \leq 1 \tag{2.33}$$

$$\sum_{j=1}^{T_Z} f_{ij} \leq 1 \tag{2.34}$$

$$\sum_{i=1}^{T_X} \sum_{j=1}^{T_Z} f_{ij} = \min(T_X, T_Z) \tag{2.35}$$

$f_{ij}$ is the flow between $x_i$ and $z_j$. The computation of the EMD requires calculating a similarity between all pairs of components of two sets and optimizing a transportation problem whose complexity is cubic with the number of features.

To address the computational issue, [Grauman & Darrell 2005a] made use of an embedding of the EMD based on the work of [Indyk & Thaper 2003]. However, the approximation suffers from a high error when the feature dimension increases.

All the previous approaches have a high computational complexity: typically $O(T_X T_Z)$ with $T_X$ and $T_Z$ varying from a few hundreds to a few thousands.

#### 2.3.3.2 Kernel functions for discrete models

Typically, the discrete models are the representations obtained by the Bag-of-Features (BoF) modelling method, and therefore are in the form of histograms. Let $F$ and $F'$ (with the same dimension $n$) be the histograms of two images, there exist many different kernel functions to measure the similarity between them:

- **Linear**: $K(F, F') = F^T F'$
- **Polynomial**: $K(F, F') = (\gamma F^T F' + r)^p, \ \gamma > 0$
- **Radial Basis Function (RBF)**: $K(F, F') = \exp(-\gamma \|F - F'\|^2), \ \gamma > 0$

- **Sigmoid**: $K(F, F') = \tanh(\gamma F^T F' + r)$

- **Chi-square**: It is one of the most popular kernel functions applied for visual object recognition task. The Chi-square ($\chi^2$) distance between $F$ and $F'$ is first computed as equation (2.36):

$$dist_{\chi^2}(F, F') = \sum_{i=1}^{n} \frac{(F_i - F'_i)^2}{F_i + F'_i} \tag{2.36}$$

  Then, the kernel function based on this distance is computed as equation (2.37):

$$K_{\chi^2}(F, F') = e^{-\frac{1}{D} dist_{\chi^2}(F,F')} \tag{2.37}$$

  where $D$ is the parameter for normalizing the distances, and is usually set to the average value of distance between each pair of images in the training set.

- **Pyramid match** [Grauman & Darrell 2005b]: It works by placing a sequence of increasingly coarser grids over the feature space and taking a weighted sum of the number of matches that occur at each level of resolution. Let $H_F^l$ and $H_{F'}^l$ denote the histograms of $F$ and $F'$ at the resolution $l$ in which we have $2^l$ bins along each dimension, $l = 0, \ldots, L$, so that $H_F^l(i)$ and $H_{F'}^l(i)$ are the numbers of points from $F$ and $F'$ that fall into the $i$-th bin of the grid. Then the number of matches at level $l$ is given by the histogram intersection function as follows:

$$I(H_F^l, H_{F'}^l) = \sum_{i=1}^{2^{nl}} \min(H_F^l(i), H_{F'}^l(i)) \tag{2.38}$$

  if we abbreviate $I(H_F^l, H_{F'}^l)$ to $I^l$, finally we get the pyramid match kernel:

$$\begin{aligned} K^L(F, F') &= I^L + \sum_{l=0}^{L-1} \frac{1}{2^{L-l}} (I^l - I^{l+1}) \\ &= \frac{1}{2^L} I^0 + \sum_{l=1}^{L} \frac{1}{2^{L-l+1}} I^l \end{aligned} \tag{2.39}$$

Here, the above $\gamma, r, p$ and $L$ are all kernel parameters.

#### 2.3.3.3 Kernel functions for continuous models

Generally, the continuous models are the representations obtained by the Gaussian Mixture Model (GMM) method, and images are modeled as continuous distributions. The probabilistic kernels can be defined between the distributions, such as the Probability Product Kernel (PPK) and the Kullback-Leibler Kernel (KLK).

Assume that we have two continuous distributions $p$ and $q$ defined on the space $\mathbb{R}^D$ ($D$ is the dimensionality of image features). Jebara et al. [Jebara & Kondor 2003] [Jebara *et al.* 2004] proposed the PPK between two distributions:

$$K_{ppk}^{\rho}(p,q) = \int_{x\in\mathbb{R}^D} p(x)^{\rho} q(x)^{\rho} dx \tag{2.40}$$

where $\rho$ is a parameter.

The PPK has two special cases. When $\rho = 1$, the PPK takes the form of the expectation of one distribution under the other. This is referred as the Expected Likelihood Kernel (ELK):

$$K_{elk}(p,q) = \int_{x\in\mathbb{R}^D} p(x)q(x)dx = E_p[q(x)] = E_q[p(x)] \tag{2.41}$$

when $\rho = 1/2$, it is known as the Bhattacharyya Kernel (BHA):

$$K_{bha}(p,q) = \int_{x\in\mathbb{R}^D} \sqrt{p(x)}\sqrt{q(x)}dx \tag{2.42}$$

The Kullback-Leibler Divergence (KLD) [Kullback 1968] is defined as follows:

$$KL(p\|q) = \int_{x\in\mathbb{R}^D} p(x)\log\frac{p(x)}{q(x)}dx \tag{2.43}$$

The symmetric KL (SKL) is given by:

$$SKL(p,q) = KL(p\|q) + KL(q\|p) \tag{2.44}$$

The KLK [Moreno *et al.* 2003] can then be defined by exponentiating the SKL:

$$K_{klk} = \exp(-\gamma SKL(p, q)) \tag{2.45}$$

where $\gamma > 0$ is the kernel parameter.

## 2.4 Fusion strategies

The idea of "fusion" is usually adopted in the problem of multimedia data analysis [Ayache *et al.* 2007]. For example, there are generally three modalities which have to be handled in videos, namely the auditory modality, the textual modality, and the visual modality. Thus, a fusion step is necessary to combine the results of the analysis of each individual modalities to get the final results [Snoek *et al.* 2005]. The same idea can also be employed in the task of visual object recognition, since different types of features usually extract information in images from different aspects, which may be complementary to each other, and thus the fusion of them may improve the recognition performance. In order to extract comprehensive information, different types of features are computed from the same image to form several information channels. These channels need to be fused to make the final decision from different information sources. There are several different strategies for fusion:

- **Early fusion**: The features from all the channels are concatenated to build a single feature vector, which is then fed into a classifier for the final classification.

- **Late fusion**: The feature from each individual channel is first fed into a classifier to get its classification score, and the scores from all the channels are then combined into the final score according to a certain criterion, such as mean, max, min, and weighted sum. Suppose $S_i, i = 1, \ldots, N$ represent the scores from $N$ individual channels, the final score $S_{fusion}$ can be obtained as follows:

  - **Mean**: $S_{fusion} = \frac{1}{N} \sum_{i=1}^{N} S_i$

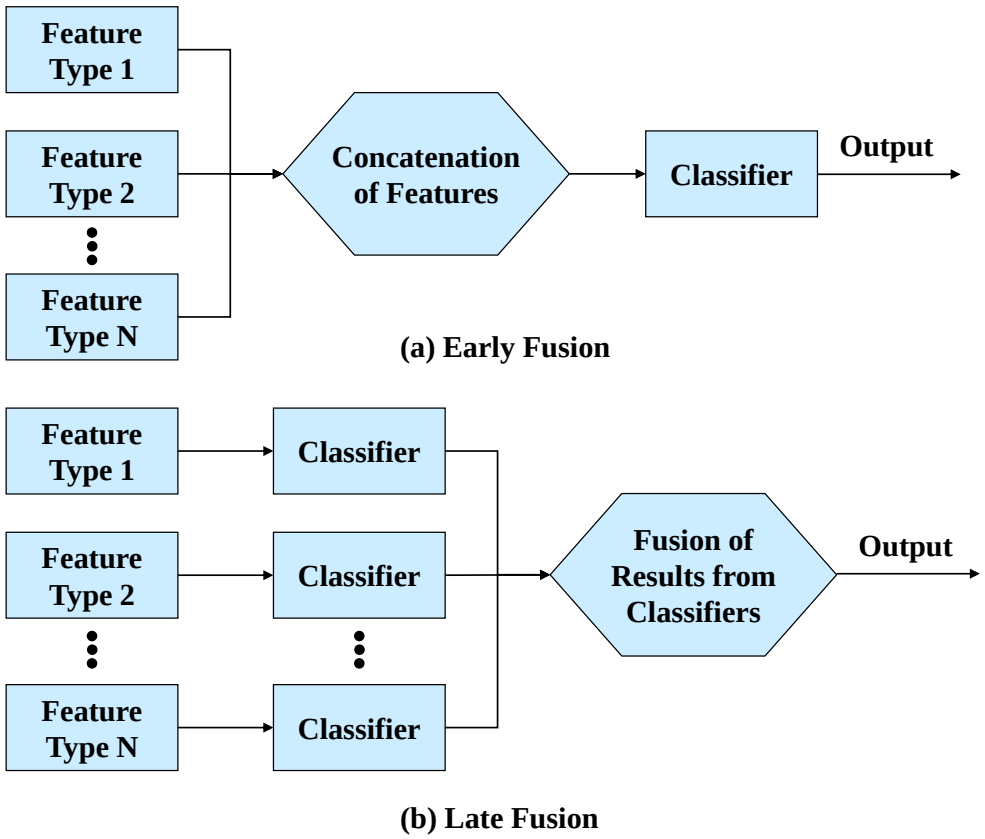



Figure 2.17: A comparison of early and late fusion strategies: (a) early fusion; (b) late fusion

- **Max**: $S_{fusion} = \max(S_1, \ldots, S_N)$
- **Min**: $S_{fusion} = \min(S_1, \ldots, S_N)$
- **Weighted sum**: $S_{fusion} = \frac{1}{N} \sum_{i=1}^{N} (\omega_i * S_i)$, where $\omega_i$ is the weight for the $i$-th channel.

- **Intermediate fusion**: As we stated in section 2.3.2.2, the Multiple Kernel Learning (MKL) method can also be interpreted as a kind of fusion technique. Different from both early and late fusion, MKL combines different features in the kernel level, and thus can be considered as a intermediate fusion strategy.

A comparison of early and late fusion strategies is illustrated in Figure 2.17.

## 2.5 Conclusions

In this chapter, a review of main approaches proposed in the literature for visual object recognition is presented. In particular, more attention is paid to the feature & classifier based approaches, because they have become the most popular framework for object recognition and classification tasks nowadays. Typically, this kind of approach consists of three steps: (1) extraction of image features (global or local); (2) image representation (or modelling); and (3) image classification (machine learning) algorithms. The popular methods adopted for each of these steps are reviewed in detail respectively. Moreover, several fusion strategies for combining different features are also introduced.

We apply the feature & classifier based approach for object recognition in this thesis, and we believe that the visual description (features) of images is a key step. Parikh and Zitnick have recently confirmed this point in their work [Parikh & Zitnick 2010]. Through statistical analysis on three main factors for visual recognition: (1) features; (2) amount of training data; and (3) learning algorithms, they have found that the main factor impacting the performance is the choice of features. Therefore, the following chapters of this thesis will focus on the visual description of images, and will propose several effective and efficient visual features for object recognition. Regarding to the other steps including image modelling and classification algorithms, we apply the most popular techniques such as the Bag-of-Features modelling and the SVM classifier.

Chapter 3

# Datasets and Benchmarks

Contents



In this chapter, we introduce several standard datasets and popular benchmarks available in computer vision community for object recognition and image / video classification tasks. Some of them will be used to carry out experiments in the following chapters.

## 3.1 PASCAL VOC

The PASCAL Visual Object Classes (VOC) challenge [1] consists of two components: (1) a publicly available dataset of images and annotations, together with standard evaluation procedures; and (2) an annual competition and workshop. Organized annually from 2005 to present, this challenge and its associated dataset has become accepted in computer vision and machine learning communities as a benchmark for visual object recognition and detection [Everingham *et al.* 2010].

[1] Website: http://pascallin.ecs.soton.ac.uk/challenges/VOC/

The goal of this challenge is to recognize objects from a number of visual object classes in realistic scenes (i.e. not pre-segmented objects). It is fundamentally a supervised learning problem in that a training set of labelled images is provided. The number of object classes considered was only 4 in the starting year of 2005, and then increased to 10 in 2006, and has further increased to 20 since 2007. The object classes that have been selected are:

- **Person:** person
- **Animal:** bird, cat, cow, dog, horse, sheep
- **Vehicle:** aeroplane, bicycle, boat, bus, car, motorbike, train
- **Indoor:** bottle, chair, dining table, potted plant, sofa, tv/monitor

There are two principal challenge tasks:

- **Classification:** For each of the twenty classes, predicting presence / absence of an example of that class in the test image.
- **Detection:** Predicting the bounding box and label of each object from the twenty target classes in the test image.

We participated in the PASCAL VOC challenge in 2009, 2010 and 2011. A brief introduction of our participation can be found in Appendix A.

Besides the challenge organized in each year, the PASCAL VOC 2007 dataset [Everingham *et al.* 2007] has become a standard benchmark for evaluating object recognition and detection algorithms, because all the annotations were made available in 2007 by the organizers but since then they have not made the test annotations publicly available. The PASCAL VOC 2007 dataset contains nearly 10 000 images of 20 object classes, which contain different number of images, from hundreds to thousands. The dataset is divided into a predefined training set (2501 images), validation set (2510 images) and test set (4952 images). The mean average precision (MAP) across all the classes is used as the evaluation criterion. Average precision (AP) measures the area under the precision-recall curve for each class, and a good

Table 3.1: Some state-of-the-art results achieved on the PASCAL VOC 2007 dataset in the literature ([1]: [Wang *et al.* 2009b]; [2]: [Khan *et al.* 2009]; [3]: [Marszalek *et al.* 2007]; [4]: [Yang *et al.* 2009b]; [5]: [Harzallah *et al.* 2009]; [6]: [Zhou *et al.* 2010]; [7]: [Perronnin *et al.* 2010]; [8]: [Wang *et al.* 2010]; [9]: [Chatfield *et al.* 2011])

| AP (%) | [1] | [2] | [3] | [4] | [5] | [6] | [7] | [8] | [9] |
|---|---|---|---|---|---|---|---|---|---|
| airplane | 65.0 | 65.0 | 77.5 | 79.4 | 77.2 | 79.4 | 75.7 | 74.8 | 79.0 |
| bicycle | 44.3 | 48.0 | 63.6 | 62.4 | 69.3 | 72.5 | 64.8 | 65.2 | 67.4 |
| bird | 48.6 | 44.0 | 56.1 | 58.5 | 56.2 | 55.6 | 52.8 | 50.7 | 51.9 |
| boat | 58.4 | 60.0 | 71.9 | 70.2 | 66.6 | 73.8 | 70.6 | 70.9 | 70.9 |
| bottle | 17.8 | 20.0 | 33.1 | 46.6 | 45.5 | 34.0 | 30.0 | 28.7 | 30.8 |
| bus | 46.4 | 49.0 | 60.6 | 62.3 | 68.1 | 72.4 | 64.1 | 68.8 | 72.2 |
| car | 63.2 | 70.0 | 78.0 | 75.6 | 83.4 | 83.4 | 77.5 | 78.5 | 79.9 |
| cat | 46.8 | 49.0 | 58.8 | 54.9 | 53.6 | 63.6 | 55.5 | 61.7 | 61.4 |
| chair | 42.2 | 50.0 | 53.5 | 63.8 | 58.3 | 56.6 | 55.6 | 54.3 | 56.0 |
| cow | 29.6 | 32.0 | 42.6 | 40.7 | 51.1 | 52.8 | 41.8 | 48.6 | 49.6 |
| table | 20.8 | 39.0 | 54.9 | 58.3 | 62.2 | 63.2 | 56.3 | 51.8 | 58.4 |
| dog | 37.7 | 40.0 | 45.8 | 51.6 | 45.2 | 49.5 | 41.7 | 44.1 | 44.8 |
| horse | 66.6 | 72.0 | 77.5 | 79.2 | 78.4 | 80.9 | 76.3 | 76.6 | 78.8 |
| motor | 50.3 | 59.0 | 64.0 | 68.1 | 69.7 | 71.9 | 64.4 | 66.9 | 70.8 |
| person | 78.1 | 81.0 | 85.9 | 87.1 | 86.1 | 85.1 | 82.7 | 83.5 | 85.0 |
| plant | 27.2 | 32.0 | 36.3 | 49.5 | 52.4 | 36.4 | 28.3 | 30.8 | 31.7 |
| sheep | 32.1 | 35.0 | 44.7 | 48.8 | 54.4 | 46.5 | 39.7 | 44.6 | 51.0 |
| sofa | 26.8 | 42.0 | 50.6 | 56.4 | 54.3 | 59.8 | 56.6 | 53.4 | 56.4 |
| train | 62.8 | 68.0 | 79.2 | 75.9 | 75.8 | 83.3 | 79.7 | 78.2 | 80.2 |
| monitor | 33.3 | 49.0 | 53.2 | 54.4 | 62.1 | 58.9 | 51.5 | 53.5 | 57.5 |
| mean | 44.9 | 50.2 | 59.4 | 62.2 | 63.5 | 64.0 | 58.3 | 59.3 | 61.7 |

AP value requires both high recall and high precision values. A detailed introduction of AP and MAP can be found in [Zhu 2004]. Some example images from each category are shown in Figure 3.2, and some state-of-the-art results achieved on this dataset in the literature are presented in Table 3.1.

## 3.2 Caltech 101

The Caltech 101 dataset [2] [Li *et al.* 2007] contains a total of 9146 images, split into 101 different object classes (including airplanes, animals, faces, vehicles, chairs, flowers, pianos, etc.) and an additional background category. The number of images in

[2]Website: http://www.vision.caltech.edu/Image_Datasets/Caltech101/

Table 3.2: Some state-of-the-art results (%) achieved on the Caltech 101 dataset in the literature

| Method \ Training Images | 5 | 10 | 15 | 20 | 25 | 30 |
|---|---|---|---|---|---|---|
| [Zhang *et al.* 2006] | 46.6 | 55.8 | 59.1 | 62.0 | – | 66.2 |
| [Lazebnik *et al.* 2006] | – | – | 56.4 | – | – | 64.6 |
| [Griffin *et al.* 2007] | 44.2 | 54.5 | 59.0 | 63.3 | 65.8 | 67.6 |
| [Boiman *et al.* 2008] | 56.9 | – | 72.8 | – | – | 79.1 |
| [Jain *et al.* 2008] | – | – | 61.0 | – | – | 69.1 |
| [Yang *et al.* 2009a] | – | – | 67.0 | – | – | 73.2 |
| [Wang *et al.* 2010] | 51.2 | 59.8 | 65.4 | 67.7 | 70.2 | 73.4 |
| [Gehler & Nowozin 2009] | 54.2 | 65.0 | 70.4 | 73.6 | 75.7 | 77.8 |
| [Yang *et al.* 2009b] | – | 65.1 | 73.2 | 80.1 | 82.7 | 84.3 |

each category varies from 31 to 800, and most categories have about 50 images. The dataset is not divided into a predefined training set and test set, and the common strategy for experiments is to randomly select (5,10,15,20,25,30) number of images from each class for training and the rest images for test. The average classification accuracy across all the classes is used as the evaluation criterion. Figure 3.1 shows some example images from the dataset, and Table 3.2 presents some state-of-the-art results achieved on this dataset in the literature.

## 3.3 ImageNet

ImageNet [3] [Deng *et al.* 2009] is a large scale image dataset organized according to the WordNet [Fellbaum 1998] hierarchy. Each meaningful concept in WordNet, possibly described by multiple words or word phrases, is called a "synonym set" or "synset". There are more than 100,000 synsets in WordNet, and majority of them are nouns (80,000+). The aim of ImageNet is to provide on average 1000 images to illustrate each synset. Images of each concept are quality-controlled and human-annotated. Currently, ImageNet contains about 15 millions of images for more than 20,000 synsets, and the number of images with bounding box annotations is more than 1 million. In its completion, ImageNet will offer tens of millions of cleanly sorted images for most of the concepts in the WordNet hierarchy.

[3] Website: http://www.image-net.org/

Starting from 2010, the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is organized based on a subset of ImageNet dataset. The aim of this competition is to estimate the content of images for the purpose of retrieval and automatic annotation. The general goal is to identify the main objects present in images. Given a subset of ImageNet for training and a set of images with no annotation for test, algorithms will have to produce labels specifying what objects are present in the images. In ILSVRC 2011, 1000 object categories are selected for recognition, and the training set contains 1.2 million images. The number of images included in the validation and test set are 50,000 and 100,000 respectively. Figure 3.5 shows some example images from the dataset for each of 1000 categories.

## 3.4 ImageCLEF

ImageCLEF [4] launched in 2003 as part of the Cross Language Evaluation Forum (CLEF) with the goal of providing an evaluation forum for the cross-language annotation and retrieval of images. Motivated by the need to support multilingual users from a global community accessing the growing amount of visual information, ImageCLEF aims to support the advancement of the field of visual media analysis, indexing, classification and retrieval by developing the necessary infrastructure for the evaluation of visual information retrieval systems operating in both monolingual, cross-language and language-independent contexts. There are four main tasks in ImageCLEF:

- Photo Annotation
- Medical Retrieval
- Plant Identification
- Wikipedia Retrieval

Among these tasks, photo annotation (also called visual concept detection and annotation) is closely related to object recognition. It aims at automatically annotating a large number of consumer photos with multiple annotations. The task

[4] Website: http://www.imageclef.org/

can be solved by following three different approaches: (1) visual information only; (2) Flickr user tags only; and (3) Multi-modal approaches combining visual information and Flickr user tags. The task uses a subset of the MIR Flickr 1 million image dataset for the annotation challenge. In ImageCLEF 2011, the training set consists of 8,000 photos annotated with 99 visual concepts, which describe the scene (indoor, outdoor, landscape, etc.), depicted objects (car, animal, person, etc.), the representation of image content (portrait, graffiti, art), events (travel, work, etc.), quality issues (overexposed, underexposed, blurry, etc.) or sentiments (happy, active, funny, etc.). The test set consists of 10,000 photos with EXIF data and Flickr user tags. The evaluation is conducted by the interpolated Average Precision and the example-based F-measure.

## 3.5 SIMPLIcity

The SIMPLIcity dataset [Wang *et al.* 2001] is a subset of the COREL image database. It contains totally 1000 images, which are equally divided into 10 different categories: African people, beach, building, bus, dinosaur, elephant, flower, horse, mountain and food. Half of the images are randomly chosen for training and the other half images are for test. The average classification accuracy is used as the evaluation criterion. Some example images from the dataset are shown in Figure 3.3.

## 3.6 OT Scene

The dataset from Oliva and Torralba [Oliva & Torralba 2001] is denoted as the OT scene dataset. It consists of 2688 color images from 8 scene categories: coast (360 samples), forest (328 samples), mountain (374 samples), open country (410 samples), highway (260 samples), inside city (308 samples), tall building (356 samples) and street (292 samples). Half of the images are randomly chosen for training and the other half are for test. The average classification accuracy is used as the evaluation criterion. Figure 3.4 shows some example images from the dataset for each category.

## 3.7 TRECVID

The TREC Video Retrieval Evaluation (TRECVID) challenge [5] [Smeaton *et al.* 2006] is organized annually by the National Institute of Standards and Technology (NIST) from 2001, and has become a popular and also very challenging benchmark in video domain. The main goal of this challenge is to promote progress in content-based analysis and retrieval from digital video via open, metrics-based evaluation. TRECVID uses video data of more than 400 hours from a small number of known professional sources — broadcast news, TV programs, and surveillance systems. These videos are characterized by a high degree of diversity in creator, content, style, production qualities, original collection device, language, etc. In TRECVID, the following tasks are evaluated:

- Semantic indexing
- Known-item search
- Event detection
- Instance search
- Content-based copy detection

Among these tasks, the semantic indexing task is closely related to object recognition. Its aim is to automatically analyze the meaning conveyed by videos and tag video segments (shots) with semantic concept labels. More precisely, given the test collection, master shot reference, and concept definitions, participants are required to return for each concept a list of at most 2000 shot IDs from the test collection ranked according to the possibility of detecting the concept. In TRECVID 2011, there are totally 346 concepts. The test set includes 200-hour video data with durations between 10 seconds and 3.5 minutes, while the development set contains 400-hour video data with durations just longer than 3.5 minutes. The mean extended inferred average precision (mean xinfAP) [Yilmaz *et al.* 2008] is used as the evaluation criterion.

[5] Website: `http://trecvid.nist.gov/`

Table 3.3: Attribute summary of main datasets and benchmarks available for object/concept recognition

| Dataset | Domain | Type | Class | Train | Val. | Test |
|---|---|---|---|---|---|---|
| PASCAL VOC 2007 | Image | Object | 20 | 2501 | 2510 | 4952 |
| Caltech 101 | Image | Object | 101 | 510-3060 | – | the rest |
| ImageNet 2011 | Image | Object | 1000 | 1.2M | 50K | 100K |
| ImageCLEF 2011 | Image | Concept | 99 | 8K | – | 10K |
| SIMPLIcity | Image | Object | 10 | 500 | – | 500 |
| OT Scene | Image | Scene | 8 | 1344 | – | 1344 |
| TRECVID 2011 | Video | Concept | 346 | 400hour | – | 200hour |

We participated in the TRECVID challenge in 2011. A brief introduction of our participation can be found in Appendix A.

The attributes of the presented datasets and benchmarks are summarized in Table 3.3, including the domain (image or video), type of recognition (object, concept, etc.), number of classes to be identified, and scale of data for training, validation and test respectively.

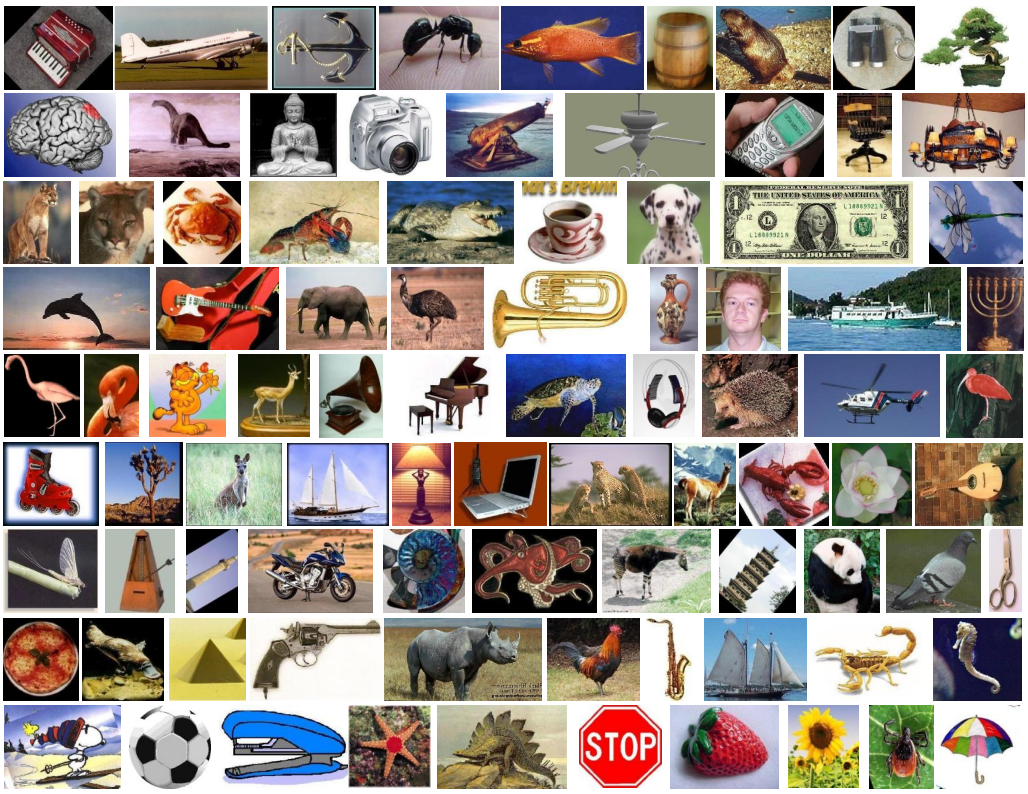



Figure 3.1: Example images of the Caltech 101 dataset

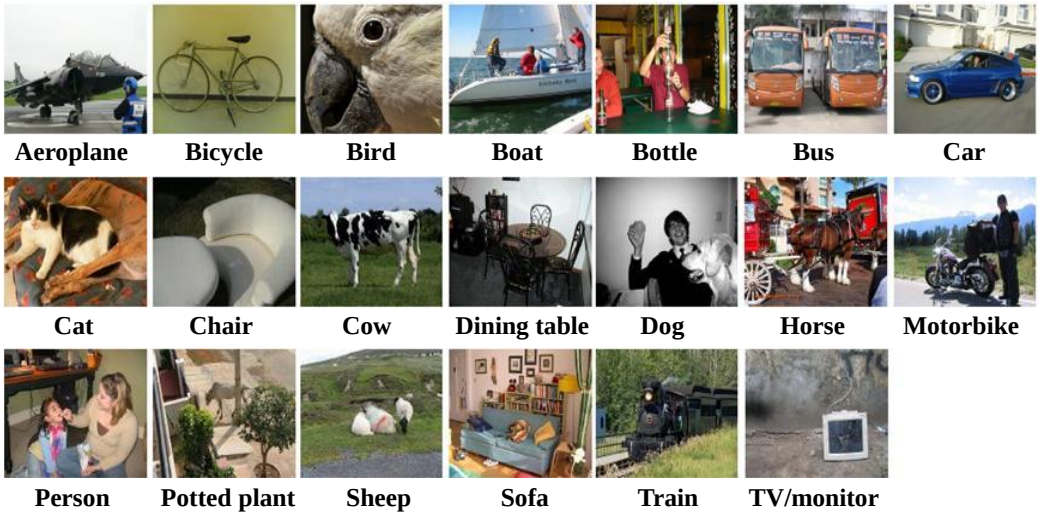



Figure 3.2: Example images of the PASCAL VOC 2007 dataset

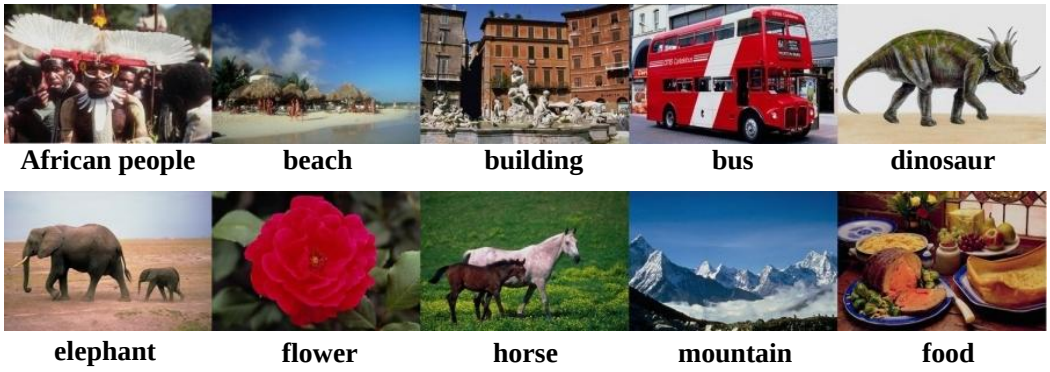



Figure 3.3: Example images of the SIMPLIcity dataset

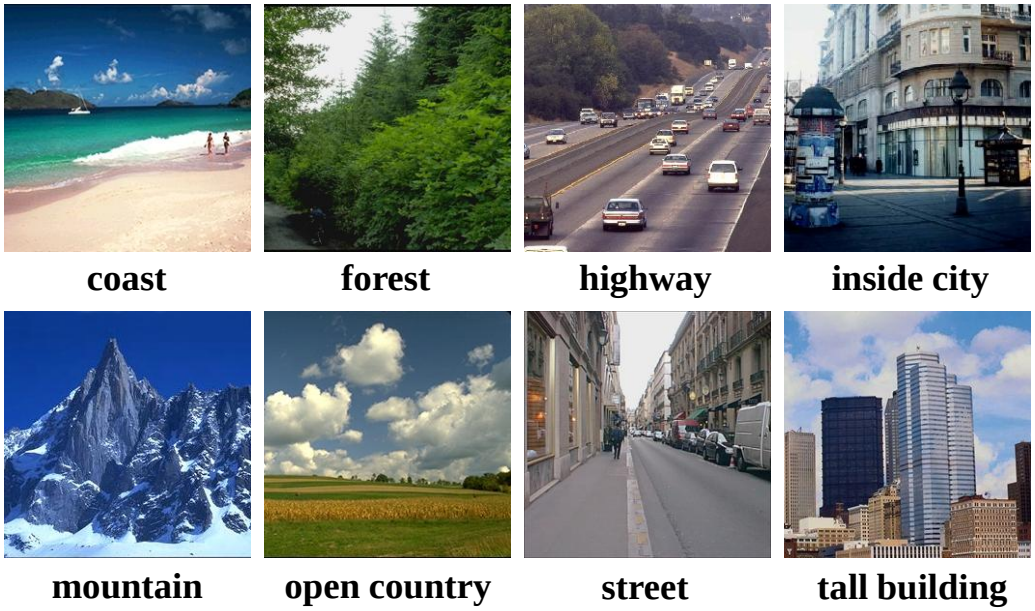



Figure 3.4: Example images of the OT Scene dataset

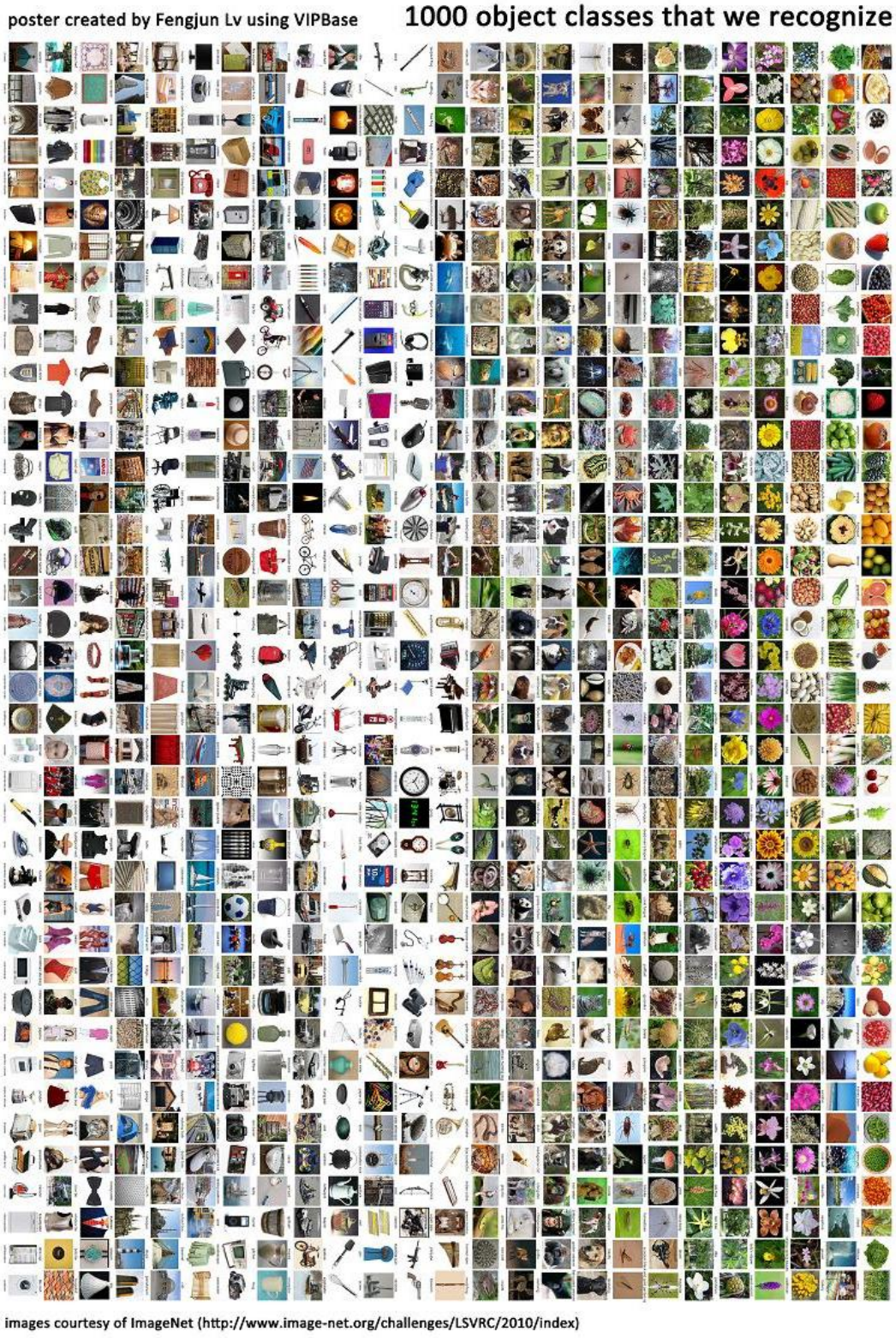



Figure 3.5: Example images of the ImageNet dataset

Chapter 4

# Multi-scale Color Local Binary Patterns for Object Recognition

Contents



## 4.1 Introduction

The Local Binary Pattern (LBP) operator [Ojala *et al.* 2002b] is a computationally efficient yet powerful texture feature. It was firstly introduced as a complementary measure for local image contrast [Ojala *et al.* 1996]. The histogram of the binary patterns computed over a region is generally used for texture description. It can be seen as a unified approach to statistical and structural texture analysis. The LBP operator describes each pixel by the relative gray levels of its neighboring pixels. Figure 4.1 illustrates the calculation of the LBP code for one pixel with 8 neighbors. Precisely, for each neighboring pixel, the result will be set to one if its value is no less

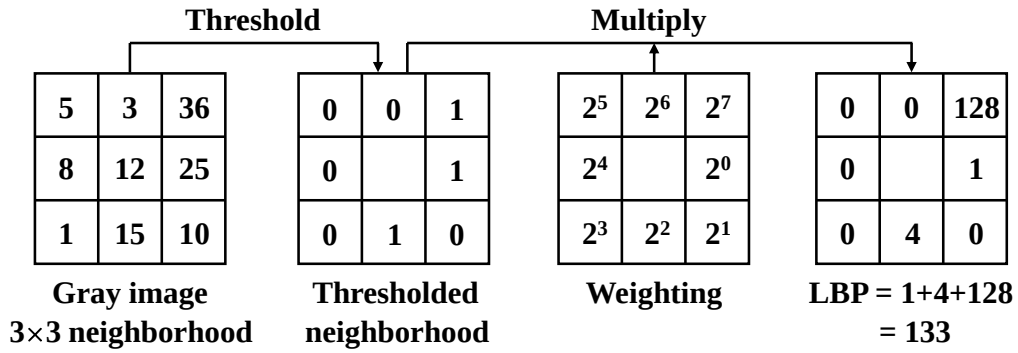



Figure 4.1: Calculation of the original LBP operator

than the value of the central pixel, otherwise the result will be set to zero. The LBP code of the central pixel is then obtained by multiplying the results with weights given by powers of two, and summing them up together. The final LBP feature of an image is generally distribution-based and consists of computing the LBP code for each pixel within the image and building a histogram based on these codes. It can be noticed that the LBP feature is very fast to calculate, and is invariant to monotonic illumination changes.

Because of its computational simplicity, and strong descriptive power for analyzing both micro and macro texture structures, the LBP feature has been successfully applied to many applications as diverse as texture classification [Mäenpää *et al.* 2000a] [Mäenpää *et al.* 2000b] [Ojala *et al.* 2002b], texture segmentation [Ojala & Pietikäinen 1999], face recognition [Ahonen *et al.* 2004] [Ahonen *et al.* 2006] and facial expression recognition [Zhao & Pietikäinen 2007] [Shan *et al.* 2009]. However, it has been rarely used in the domain of visual object recognition [1]. We hold that main reasons lie in two aspects. On one hand, the LBP feature ignores all color information (its calculation is based on gray image), while color is an important clue for distinguishing objects, especially in natural scenes. On the other hand, there can be various changes in lighting and viewing conditions in real-world scenes, leading to large illumination variations of object's appearance, which further complicate the recognition task. According to its definition, the LBP

[1] at the time when we started our work in 2008, while being more popular now

feature is only invariant to gray-level monotonic light changes, and thus is deficient in power to deal with the mentioned variations.

Therefore, in order to incorporate color information, as well as to enhance the discriminative power and the photometric invariance property of the original LBP, we propose, in this chapter, six multi-scale color LBP features which are more suitable for visual object recognition task. The performances of the proposed features are analyzed experimentally using the PASCAL VOC 2007 image benchmark [Everingham *et al.* 2007].

## 4.2 Model analysis for illumination changes

Changes in illumination can be expressed by the diagonal model as equation (4.1) and the diagonal-offset model as equation (4.2), where $u$ and $c$ represent respectively the values before and after illumination transformation:

$$\begin{pmatrix} R^c \\ G^c \\ B^c \end{pmatrix} = \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix} \begin{pmatrix} R^u \\ G^u \\ B^u \end{pmatrix} \tag{4.1}$$

$$\begin{pmatrix} R^c \\ G^c \\ B^c \end{pmatrix} = \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix} \begin{pmatrix} R^u \\ G^u \\ B^u \end{pmatrix} + \begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} \tag{4.2}$$

Based on these two models, different kinds of illumination changes can be expressed as follows [van de Sande *et al.* 2010]:

**Light intensity change.** Image values change by a constant factor in all channels ($a = b = c$):

$$\begin{pmatrix} R^c \\ G^c \\ B^c \end{pmatrix} = \begin{pmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{pmatrix} \begin{pmatrix} R^u \\ G^u \\ B^u \end{pmatrix} \tag{4.3}$$

**Light intensity shift.** Image values change by an equal offset in all channels

$(a = b = c = 1, O_1 = O_2 = O_3)$:

$$\begin{pmatrix} R^c \\ G^c \\ B^c \end{pmatrix} = \begin{pmatrix} R^u \\ G^u \\ B^u \end{pmatrix} + \begin{pmatrix} O_1 \\ O_1 \\ O_1 \end{pmatrix} \tag{4.4}$$

**Light intensity change and shift.** Image values change by combining two kinds of change above:

$$\begin{pmatrix} R^c \\ G^c \\ B^c \end{pmatrix} = \begin{pmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{pmatrix} \begin{pmatrix} R^u \\ G^u \\ B^u \end{pmatrix} + \begin{pmatrix} O_1 \\ O_1 \\ O_1 \end{pmatrix} \tag{4.5}$$

**Light color change.** Image values change in all channels independently ($a \neq b \neq c$), as equation (4.1).

**Light color change and shift.** Image values change in all channels independently with arbitrary offsets ($a \neq b \neq c$ and $O_1 \neq O_2 \neq O_3$), as equation (4.2).

## 4.3 Color LBP features and their properties

In order to incorporate color information into the original LBP, as well as to enhance its discriminative power and photometric invariance property for dealing with different kinds of illumination changes as described in section 4.2, six color LBP features are proposed in this chapter. The main idea is to calculate the original LBP operator independently over different channels of a certain color space, and then concatenate the resulting histograms to get the final color LBP feature, as shown in Figure 4.2.

The $RGB$, $HSV$, and $OPPONENT$ color spaces are chosen for calculating color LBP features because of their own characteristics. $RGB$ is the most popular color space used in electronic systems for sensing, representation and display of images. It uses additive color mixing with primary colors of red, green and blue to reproduce a broad array of colors. $HSV$ color space rearranges the geometry of $RGB$ so that it could be more relevant to human perception, because it is more natural

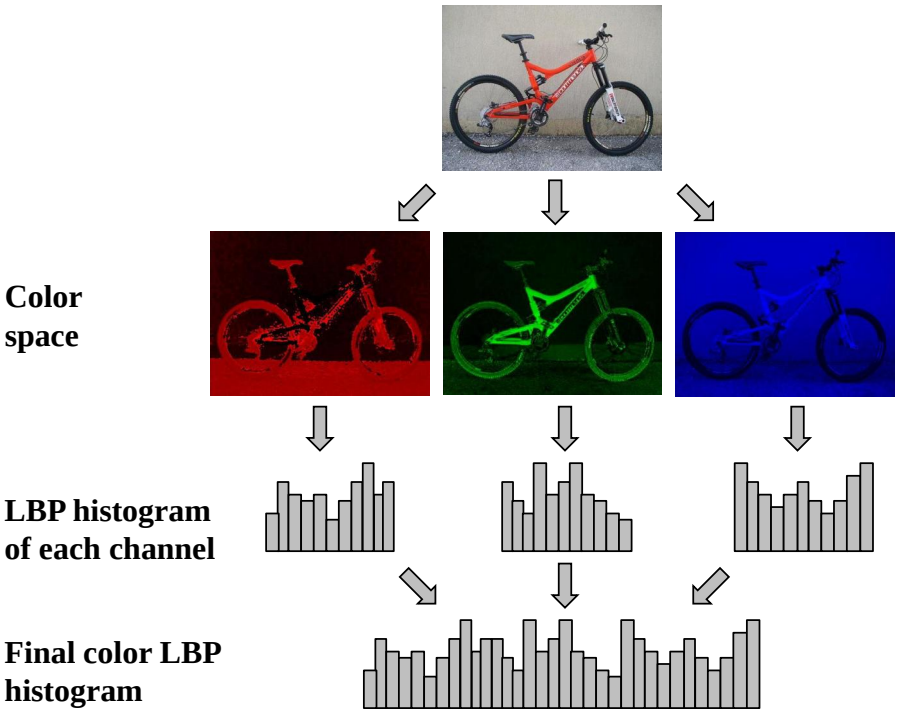



Figure 4.2: Calculation of color LBP feature

to think about a color in terms of hue and saturation than in terms of additive color components. *OPPONENT* color space is constructed to be consistent with human visual system, because it proves more efficient for human visual system to record differences between responses of cones, rather than each type of cone's individual response. Details of the proposed color LBP features and their properties are listed as follows:

**RGB-LBP.** This feature is obtained by computing LBP over all three channels of the *RGB* color space. It is invariant to monotonic light intensity change due to the property of the original LBP, and has no additional invariance properties.

**nRGB-LBP.** This feature is obtained by computing LBP over both $r$ and $g$ channels of the *normalized RGB* color space as equation (4.6) ($b$ channel is redundant because $r + g + b = 1$):

$$\begin{pmatrix} r \\ g \end{pmatrix} = \begin{pmatrix} R/(R+G+B) \\ G/(R+G+B) \end{pmatrix} \tag{4.6}$$

Due to the normalization, the change factors can be cancelled out if they are constant in all channels. This is proven as equation (4.7) (Let $a$ be the constant factor):

$$\begin{aligned}\begin{pmatrix} r \\ g \end{pmatrix} &= \begin{pmatrix} R/(R+G+B) \\ G/(R+G+B) \end{pmatrix} = \begin{pmatrix} aR'/(aR'+aG'+aB') \\ aG'/(aR'+aG'+aB') \end{pmatrix} \\ &= \begin{pmatrix} aR'/a(R'+G'+B') \\ aG'/a(R'+G'+B') \end{pmatrix} = \begin{pmatrix} R'/(R'+G'+B') \\ G'/(R'+G'+B') \end{pmatrix}\end{aligned} \tag{4.7}$$

Therefore, $r$ and $g$ channels are scale-invariant, which make this feature invariant to light intensity change as equation (4.3).

**OPPONENT-LBP.** This feature is obtained by computing LBP over all three channels of the $OPPONENT$ color space as equation (4.8):

$$\begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} = \begin{pmatrix} (R-G)/\sqrt{2} \\ (R+G-2B)/\sqrt{6} \\ (R+G+B)/\sqrt{3} \end{pmatrix} \tag{4.8}$$

Due to the subtraction in channel $O_1$ and $O_2$, the change offsets can be cancelled out if they are equal in all channels. This is proven as equation (4.9) (Let $a$ be the equal offset):

$$\begin{aligned}\begin{pmatrix} O_1 \\ O_2 \end{pmatrix} &= \begin{pmatrix} (R-G)/\sqrt{2} \\ (R+G-2B)/\sqrt{6} \end{pmatrix} \\ &= \begin{pmatrix} ((R'+a)-(G'+a))/\sqrt{2} \\ ((R'+a)+(G'+a)-2(B'+a))/\sqrt{6} \end{pmatrix} \\ &= \begin{pmatrix} (R'-G')/\sqrt{2} \\ (R'+G'-2B')/\sqrt{6} \end{pmatrix}\end{aligned} \tag{4.9}$$

Therefore, $O_1$ and $O_2$ channels are invariant to light intensity shift as equation (4.4). $O_3$ channel represents the intensity information, and has no invariance properties.

**nOPPONENT-LBP.** This feature is obtained by computing LBP over two

channels of the *normalized OPPONENT* color space as equation (4.10):

$$\begin{pmatrix} O'_1 \\ O'_2 \end{pmatrix} = \begin{pmatrix} \frac{O_1}{O_3} \\ \frac{O_2}{O_3} \end{pmatrix} = \begin{pmatrix} \frac{\sqrt{3}(R-G)}{\sqrt{2}(R+G+B)} \\ \frac{R+G-2B}{\sqrt{2}(R+G+B)} \end{pmatrix} \tag{4.10}$$

Due to the normalization by intensity channel $O_3$, $O'_1$ and $O'_2$ channels are scale-invariant, which make this feature invariant to light intensity change as equation (4.3).

**Hue-LBP.** This feature is obtained by computing LBP over the $Hue$ channel of the $HSV$ color space as equation (4.11):

$$Hue = \arctan(\frac{O_1}{O_2}) = \arctan(\frac{\sqrt{3}(R-G)}{R+G-2B}) \tag{4.11}$$

Due to the subtraction and the division, $Hue$ channel is scale-invariant and shift-invariant, therefore this feature is invariant to light intensity change and shift as equation (4.5).

**TC-LBP.** This feature is obtained by computing LBP over all three channels of the $transformed$ color space as equation (4.12) ($\mu$ is the mean and $\sigma$ is the standard deviation of each channel):

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} (R-\mu_R)/\sigma_R \\ (G-\mu_G)/\sigma_G \\ (B-\mu_B)/\sigma_B \end{pmatrix} \tag{4.12}$$

Due to the subtraction and the normalization, all three channels are scale-invariant and shift-invariant, which make this feature invariant to light intensity change and shift as equation (4.5). Furthermore, because each channel is operated independently, this feature is also invariant to light color change and shift as equation (4.2).

## 4.4 Multi-scale color LBP features

Another big limitation of the original LBP operator is that it only covers a fixed small neighborhood area (8 neighboring pixels as default), and thus can only get

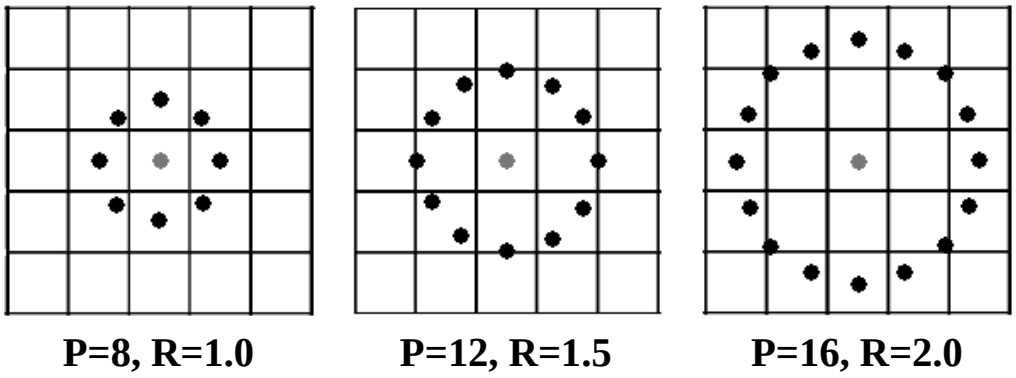



Figure 4.3: Multi-scale LBP operator

very limited local information. In order to obtain more local information by covering larger neighborhood area with different size, and therefore to increase its discriminative power, multi-scale LBP operator [Ojala *et al.* 2002b] is applied by combining different LBP operators which use a circular neighborhood with different radius and different number of neighboring pixels. Figure 4.3 gives an example.

Formally, the LBP code of the pixel at $(x_c, y_c)$ is calculated according to the following equation:

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} S(g_p - g_c) \times 2^p \tag{4.13}$$

$$S(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{4.14}$$

where $g_c$ is the value of the central pixel, $g_p$ corresponds to the gray values of the $P$ neighboring pixels equally located on a circle of radius $R$.

Therefore, the final multi-scale color LBP features can be obtained by extending color LBP features proposed in section 4.3 to their corresponding multi-scale forms respectively. By doing this, the proposed features are not only invariant to different illumination changes, but also scale-invariant to a certain extent.

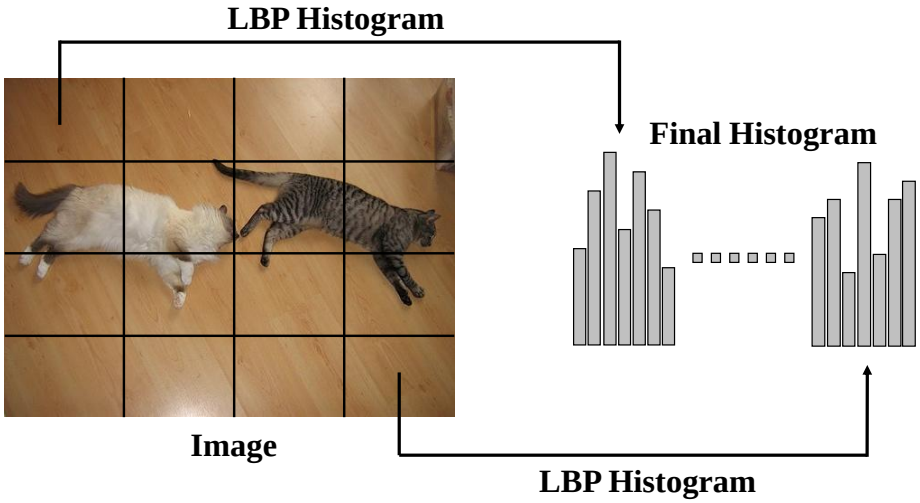



Figure 4.4: Computing color LBP features within image blocks

## 4.5 Computing color LBP features within image blocks

Usually, an image can be represented as a single histogram computed by applying each of the proposed color LBP features over the whole image. However, this only encodes the occurrences of the texture structures in images without any information about their locations.

Therefore, in order to include the coarse spatial relations of the texture structures, we equally divide an image into $M \times M$ non-overlapping blocks within which an LBP histogram is computed. The final LBP feature of the whole image is then the concatenation of the LBP histograms computed within all the blocks, as shown in Figure 4.4.

By changing the number of blocks dividing an image, we can obtain different levels of spatial information. Usually, the more blocks we divide, the more detailed spatial information we could obtain, and maybe the better recognition performance we could get. On the other hand, more number of blocks means larger feature vector dimensions, and more requirements for storage and computation cost. So the number of blocks should be chosen carefully as a trade-off between recognition performance and feature vector size.

We apply a coarse-to-fine strategy to evaluate the performances of the proposed color LBP features under different number of blocks. We found that finer division

gives better results until a peak reaches. And the features from different levels of division are not completely redundant, since combining them can further boost the recognition performance. The detailed analysis is given in section 4.6.2.3.

## 4.6 Experimental evaluation

The PASCAL VOC 2007 image benchmark [Everingham *et al.* 2007] is used to evaluate the performances of the proposed color LBP features. Its detailed introduction can be found in section 3.1. All the images in this dataset are taken from real-world scenes under variant lighting conditions, which makes it very suitable for evaluating the proposed features.

### 4.6.1 Experimental Setup

The same multi-scale configuration, as shown in Figure 4.3, is applied for all the proposed color LBP features: 8 neighboring pixels with radius 1, 12 neighboring pixels with radius 1.5, and 16 neighboring pixels with radius 2.

Three widely-used texture features are chosen to make comparisons, including: Gabor filters [Zhang *et al.* 2000], Grey Level Co-occurrence Matrix (GLCM) [Tuceryan & Jain 1998], and Texture Auto-Correlation (TAC) [Tuceryan & Jain 1998]. A detailed introduction of these features can be found in section 2.2.1. For Gabor filters, 5 scales and 8 orientations are used. For GLCM, 4 directions (horizontal, vertical and two diagonals) with 1 offset between two pixels are considered. For TAC, (0,2,4,6,8) are applied as position difference in both $x$ and $y$ directions.

The Support Vector Machine (SVM) algorithm is applied for classification. An introduction of SVM can be found in section 2.3.2.1. Here the LibSVM implementation [Chang & Lin 2001] is used. Once all the features are extracted from the dataset, the Chi-square ($\chi^2$) kernel is computed as equation (2.36) and (2.37) for the SVM training and prediction. The Chi-square ($\chi^2$) kernel is chosen for SVM because it is very suitable for computing similarities between features in terms of histogram, and has been proven to outperform other popular kernels such as linear,

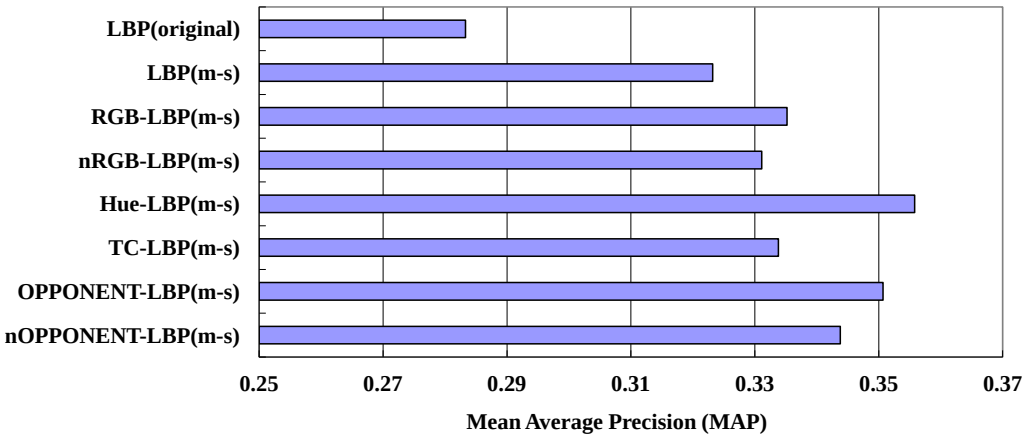



Figure 4.5: Comparison of the proposed multi-scale color LBP features and the original LBP ("m-s" is the abbreviation of "multi-scale")

quadratic and RBF (Radial Basis Function) [Zhang *et al.* 2007]. Finally, for each category, the precision-recall curve is plotted according to the output decision values of the SVM classifier, and the AP (Average Precision) value is computed based on the proportion of the area under this curve. We train the classifier on the training set, then tune the parameters on the validation set, and obtain the classification results on the test set.

### 4.6.2 Experimental Results

#### 4.6.2.1 Comparison with the original LBP

The proposed multi-scale color LBP features are first compared with the original LBP with 8 nearest neighbors.

From the results shown in Figure 4.5, it can be seen that intensity-based multi-scale LBP outperforms the original LBP by 14.1%, proving the importance of obtaining more local information and invariance to scaling. The proposed multi-scale color LBP features all further outperform intensity-based multi-scale LBP, with the improvements from 2.5% to 10.2% (17.0% to 25.8% if compared with the original LBP), which proves that the proposed features truly have more discriminative power benefitting from color information and the additional properties of illumination

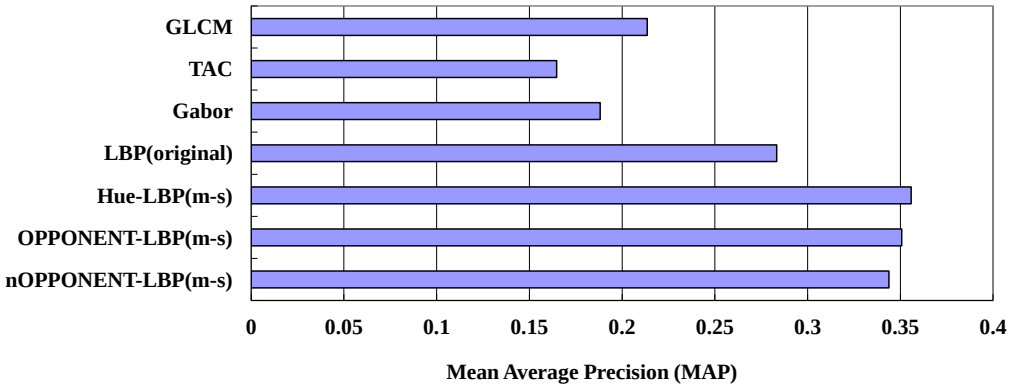



Figure 4.6: Comparison of the proposed multi-scale color LBP features and other popular texture features ("m-s" is the abbreviation of "multi-scale")

invariance.

It also can be noticed that among these features, Hue-LBP, OPPONENT-LBP and nOPPONENT-LBP have the best overall performance (improvement over 6% than intensity-based multi-scale LBP and over 20% than the original LBP), consistent with their strong properties of illumination invariance.

#### 4.6.2.2 Comparison with other popular texture features

As one kind of texture feature, the best three multi-scale color LBP features are also compared with other popular texture features, including Gabor, GLCM and TAC.

From the results shown in Figure 4.6, it can be seen that the original LBP already outperforms other popular texture features, proving its superior ability of describing texture structures. The best three multi-scale color LBP features further improve the performances to almost double of the other texture features, demonstrating their strong discriminative power which benefits from the properties of illumination-invariant and scale-invariant.

#### 4.6.2.3 Influence of image division strategy

The proposed multi-scale color LBP features are then evaluated under different image division strategies. For the number of blocks in images, we equally divide

Table 4.1: Mean Average Precision (MAP) of the proposed multi-scale color LBP features under different image division strategies ("m-s" is the abbreviation of "multi-scale")

| Feature \ Block(s) | $1 \times 1$ | $2 \times 2$ | $3 \times 3$ | $4 \times 4$ | $5 \times 5$ | Fusion |
|---|---|---|---|---|---|---|
| LBP(original) | 0.283 | 0.340 | 0.363 | 0.360 | 0.358 | **0.379** |
| LBP(m-s) | 0.323 | 0.346 | 0.374 | 0.365 | 0.361 | **0.403** |
| RGB-LBP(m-s) | 0.335 | 0.355 | 0.380 | 0.373 | 0.370 | **0.414** |
| nRGB-LBP(m-s) | 0.331 | 0.350 | 0.378 | 0.370 | 0.368 | **0.410** |
| Hue-LBP(m-s) | 0.356 | 0.374 | 0.392 | 0.385 | 0.380 | **0.425** |
| TC-LBP(m-s) | 0.334 | 0.353 | 0.380 | 0.374 | 0.370 | **0.415** |
| OPPONENT-LBP(m-s) | 0.351 | 0.370 | 0.390 | 0.382 | 0.378 | **0.424** |
| nOPPONENT-LBP(m-s) | 0.344 | 0.365 | 0.386 | 0.380 | 0.375 | **0.421** |

each image into $1 \times 1, 2 \times 2, \ldots, 5 \times 5$ non-overlapping blocks, and extract the proposed features respectively.

From the results shown in Table 4.1, it can be seen that extracting the proposed features within image blocks instead of the whole image is a simple, but efficient and effective way to improve their recognition performances. When the number of blocks increases from $1 \times 1$ to $2 \times 2$, the improvements of the MAP values are 20.1% for the original LBP, 7.1% for intensity-based multi-scale LBP, and 5.1% to 6.1% for multi-scale color LBP features respectively. When the number of blocks increases from $2 \times 2$ to $3 \times 3$, the improvements of the MAP values are 6.8% for the original LBP, 8.1% for intensity-based multi-scale LBP, and 4.8% to 8.0% for multi-scale color LBP features respectively. Then the MAP values start to decrease if the number of blocks continues to increase. This may be because the important texture structures of objects are broken into pieces if the block size is too small. Therefore, $3 \times 3$ could be an appropriate number of blocks for the proposed features with good performance and relatively low dimensions.

Furthermore, we found that the features from different levels of division are not completely redundant, since fusing them can further boost the recognition performance. The MAP values improve, after fusion of the features from all the five levels, 4.4% for the original LBP, 7.8% for intensity-based multi-scale LBP, and 8.4% to 9.2% for multi-scale color LBP features respectively.

Table 4.2: Fusion of different color LBP features in $3 \times 3$ blocks ("m-s" is the abbreviation of "multi-scale")

| Feature | Mean Average Precision (MAP) |
|---|---|
| Hue-LBP(m-s) | 0.392 |
| OPPONENT-LBP(m-s) | 0.390 |
| nOPPONENT-LBP(m-s) | 0.386 |
| **Fusion** | **0.411** |

#### 4.6.2.4 Fusion of different color LBP features

It is also worthy to notice that from the results shown in Table 4.2, further improvement (about 5%) on performance can be obtained by fusing the best three multi-scale color LBP features, proving that different color LBP features can provide complementary information to each other, and the fusion of them can boost the recognition performance.

## 4.7 Conclusions

In this chapter, we propose six multi-scale color LBP features to deal with the main shortcomings of the original LBP, namely deficiency of color information and sensitivity to non-monotonic lighting condition changes. The proposed features not only have more discriminative power by obtaining more local information, but also possess invariance properties to different lighting condition changes. They also keep the advantage of computational simplicity from the original LBP. In addition, we apply a coarse-to-fine image division strategy for calculating the proposed features within image blocks in order to encode spatial information of texture structures, thereby further improving their performances. The experimental results on the PASCAL VOC 2007 image benchmark prove that the proposed features can gain significant improvement on recognition accuracy, and thus are promising for real-world object recognition tasks.

Chapter 5

# Image Region Description Using Orthogonal Combination of Local Binary Patterns Enhanced with Color Information

Contents

## 5.1 Introduction

Machine-based automatic object recognition and scene classification is one of the most challenging problems in computer vision. The difficulties are mainly due to intra-class variations and inter-class similarities. Therefore, a key issue and the first important step when solving such problems is to generate good visual content descriptions, which should be both discriminative and computationally efficient, while possessing some properties of robustness to changes in viewpoint, scale and lighting conditions.

Local image descriptors have received a lot of attention in recent years, and have already gained the popularity and dominance in image analysis and understanding tasks nowadays. Many different local descriptors have been proposed in the literature (see section 2.2.2.2 for a more detailed introduction). Several comprehensive studies on local descriptors [Mikolajczyk & Schmid 2005] [Zhang *et al.* 2007] [Li & Allinson 2008] have shown that distribution-based descriptors perform significantly better than other features, and achieve the best results in tasks as diverse as image region matching, texture classification, object recognition and scene classification. Among them, SIFT [Lowe 2004] is considered as the most powerful and successful one, and has been widely applied as the dominant feature in the state-of-the-art recognition / classification systems [Everingham *et al.* 2010]. Moreover, since SIFT is an intensity-based descriptor without any color information, several color SIFT descriptors have been proposed [Abdel-Hakim & Farag 2006] [Bosch *et al.* 2008] [van de Weijer *et al.* 2006] [Burghouts & Geusebroek 2009] to enhance its discriminative power. In [van de Sande *et al.* 2010], the authors evaluated different color descriptors in a structured way, and recommended to use color SIFT descriptors for object and scene recognition because they outperform the original SIFT. However, the downside of color SIFT descriptors is their high computational cost, especially when the size of image or the scale of dataset significantly increases. Therefore, it is highly desirable that local image descriptors offer both high discriminative power and computational efficiency.

The Local Binary Pattern (LBP) operator [Ojala *et al.* 2002b] introduced in

chapter 4 is a well known texture feature which has been successfully applied to many applications. It has several interesting properties. First of all, it is simple and fast to compute. Moreover, it offers strong discriminative power for the description of texture structure while staying robust to monotonic lighting changes. All these advantages make LBP a good candidate for describing local image regions. However, the LBP operator tends to produce high dimensional feature vectors, especially when the number of considered neighboring pixels increases. The so-called "curse of dimensionality" is a barrier for using it directly as a local region descriptor. Thus, a key issue of making LBP a local region descriptor is to reduce its dimensionality. There exist in the literature two main works, namely "uniform patterns" [Ojala *et al.* 2002b] and center-symmetric local binary pattern (CS-LBP) operator [Heikkilä *et al.* 2009], which address this issue.

In this chapter, we propose a new dimensionality reduction method for LBP, denoted as the orthogonal combination of local binary patterns (OC-LBP), which is more effective and offers high discriminative power of local texture patterns. The basic idea is to first split the neighboring pixels of the original LBP operator into several non-overlapped orthogonal groups, then compute the LBP code separately for each group, and finally concatenate them together. The experimental results on a standard texture classification dataset show that our method is much more effective than both CS-LBP operator and "uniform patterns" in terms of dimension reduction, since our method produces the LBP features with the smallest dimensions while still keeping high classification accuracy.

The proposed OC-LBP operator is then adopted to build a distribution-based local image region descriptor, denoted as OC-LBP descriptor, by following a way similar to SIFT: given several local regions of an image, each region is firstly divided into small cells for spatial information; in each cell, the OC-LBP feature is then computed for each pixel and an LBP histogram is constructed; finally, all the histograms from the cells are concatenated and delivered as the final region descriptor. Our aim is to build a more efficient local descriptor by replacing the costly gradient information with local texture patterns in the SIFT scheme.

Furthermore, similar to the extension of SIFT to color SIFT, we also extend the

OC-LBP descriptor to different color spaces and propose six color OC-LBP descriptors in this chapter to increase the photometric invariance properties and enhance the discriminative power of the intensity-based descriptor. In chapter 4, we have proposed several color LBP features, which are based on the original LBP operator and serve as global features. Different from them, the proposed color OC-LBP descriptors in this chapter are based on the orthogonal combination of the LBP operator, and serve as local features. They could thus be considered as the extensions of our previous work in chapter 4. The experimental results in three different applications show that the proposed descriptors outperform the popular SIFT, HOG, SURF and CS-LBP descriptor, and achieve comparable or even better performances than the state-of-the-art color SIFT descriptors. Meanwhile, the proposed descriptors provide complementary information to SIFT, because a fusion of these two kinds of descriptors is found to perform clearly better than either of the two separately. Moreover, the proposed descriptors are more computationally efficient than color SIFT.

## 5.2 Dimensionality reduction of LBP

### 5.2.1 Original LBP operator

The original LBP operator was firstly introduced as a complementary measure for local image contrast [Ojala *et al.* 1996], and can be seen as a unified approach to statistical and structural texture analysis. The detailed introduction of the original LBP operator is given in chapter 4. The advantage of the LBP feature is that it is very fast to calculate, and is invariant to monotonic illumination changes. Thus it is a good candidate for local image region description.

However, the drawback of the LBP feature lies in the high dimensional histogram produced by the LBP codes. Let $P$ be the total number of neighboring pixels, then the LBP feature will have $2^P$ distinct values, resulting in a $2^P$-dimensional histogram. For example, the size of the LBP histogram will be 256/65536 if 8/16 neighboring pixels are considered. It will rapidly increase to a huge number if more neighboring pixels are taken into consideration. Thus, a dimensionality reduction

method for LBP is needed to address this problem.

### 5.2.2 Orthogonal combination of local binary patterns (OC-LBP)

To reduce the dimensionality of the LBP histogram, a straightforward way is to only consider fewer neighboring pixels. For example, the LBP operator with 8 neighbors is mostly used in the applications, and it produces a rather long (256-dimensional) histogram, see the left column of Figure 5.1 for an illustration. The size of the LBP histogram will significantly reduce to 16 if only 4 neighboring pixels are taken into account, as illustrated in the middle column of Figure 5.1. However, this brut reduction also decreases the discriminative power of the LBP feature because compared to 8 neighbors, only horizontal and vertical neighbors are considered, and the information of diagonal neighborhood is discarded. We need to find out a trade-off between the reduction of the LBP histogram dimensionality and its descriptive power.

In this chapter, we propose an orthogonal combination of local binary patterns, namely OC-LBP, which drastically reduces the dimensionality of the original LBP histogram while keeping its discriminative power. Specifically, given $P$ neighboring pixels equally located on a circle of radius $R$ around a central pixel $c$, OC-LBP is obtained by combining the histograms of $[P/4]$ different 4-orthogonal-neighbor operators, each of which consists of turning the previous 4 orthogonal neighbors by one position in a clockwise direction. The dimension of an OC-LBP based histogram is thus $2^4 \times [P/4]$ or simply $4 \times P$, which is linear with the number of neighboring pixels in comparison to $2^P$ for the original LBP-based scheme.

Figure 5.1 illustrates the construction process of an OC-LBP operator with 8 neighboring pixels. In this case, two regular 4-neighbor LBP operators are considered. The first one consists of the horizontal and vertical neighbors, and the second one consists of the diagonal neighbors. By concatenating these two LBP histograms, we obtain the OC-LBP histogram with 32 dimensions, which is 8 times more compact than the original 8-neighbor LBP histogram (256 dimensions). Meanwhile, this combination keeps quite well the discriminative power of the original LBP because it preserves the same number of distinct binary patterns ($2^4 \times 2^4$) as before ($2^8$).

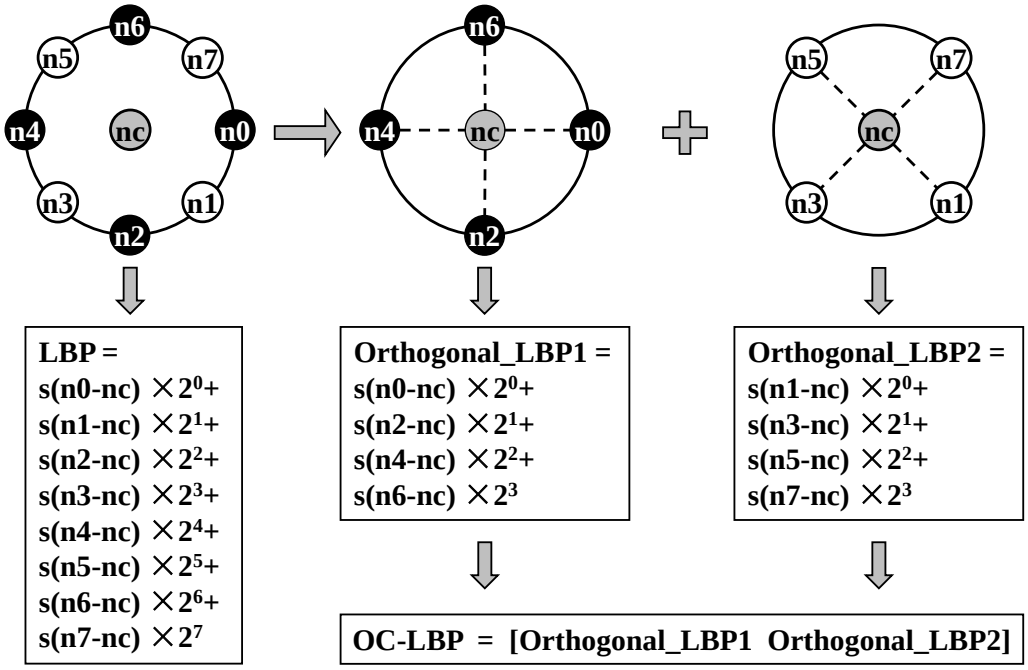



Figure 5.1: Calculation of the original LBP and OC-LBP operators with 8 neighboring pixels

This orthogonal combination of local binary patterns (OC-LBP) can also be generalized in different ways. For instance, the neighboring pixels of the original LBP can be firstly split into several non-overlapped orthogonal groups, then the LBP code can be computed separately for each group, and finally the histograms based on these separate LBP codes can be concatenated and used as the image description.

### 5.2.3 Comparison of OC-LBP and other popular LBP dimensionality reduction methods

We make a comparison between the proposed OC-LBP and other two popular dimensionality reduction methods for LBP both in terms of discriminative power and feature dimensionality. These two methods, namely "uniform patterns" [Ojala *et al.* 2002b] and CS-LBP [Heikkilä *et al.* 2009], are compared in this section with OC-LBP on operator level. The comparisons in the context of local region

Table 5.1: Comparison of the histogram dimensionality of different methods with $P$ neighboring pixels

| LBP | Uniform patterns | CS-LBP | OC-LBP |
|---|---|---|---|
| $2^P$ | $P \times (P-1) + 3$ | $2^{[P/2]}$ | $4 \times P$ |

descriptor will be presented in section 5.5.

In [Ojala *et al.* 2002b], the authors proposed the concept of "uniform patterns", which are certain parts of the original LBP, and are considered to be the fundamental properties of texture. These patterns are called "uniform" because they have one thing in common: no more than two spatial transitions (one-to-zero or zero-to-one) in the circular binary code. For $P$ neighboring pixels, they lead to a histogram of $P \times (P-1)+3$ dimensions. The "uniform patterns" have been proven to be an effective way for LBP dimensionality reduction [Huang *et al.* 2011]. In [Heikkilä *et al.* 2009], the authors proposed center-symmetric local binary pattern (CS-LBP) operator for dimensionality reduction. They modified the scheme of how to compare the pixels in the neighborhood. Instead of comparing each pixel with the central pixel, they compare center-symmetric pairs of pixels. This halves the number of comparisons compared to the original LBP.

Table 5.1 summarizes the dimensionality of the histograms produced by different methods with $P$ neighboring pixels.

As we can see, the most effective scheme in terms of histogram dimensionality reduction is the proposed OC-LBP, which is linear with $P$ — the number of neighboring pixels, compared to exponential dimension of the original LBP and CS-LBP, and quadratic dimension of "uniform patterns". Then, these methods are further compared in terms of their discriminative power.

Since the LBP operator is originally designed as a texture feature, a standard texture classification dataset [Ojala *et al.* 2002a] is chosen to carry out the comparisons. This dataset, namely Outex_TC_00014, contains images of 68 different textures, such as canvas, carpet, granite, tile, sandpaper, wood, and so on. Each kind of texture produces three images of size $746 \times 538$ pixels under three different illuminants: 2856K incandescent CIE A light source (Inca), 2300K horizon sunlight

(Horizon) and 4000K fluorescent TL84 (TL84). Then each image is equally divided into 20 non-overlapping sub-images of size $128 \times 128$ pixels, resulting in 1360 images for each illuminant. The training set is constituted by half of the images under the Inca illuminant, and the test set is constituted by half of the images under the two other illuminants (Horizon and TL84). Therefore, the total numbers of training and test images are 680 and 1360 respectively.

For texture classification, we follow the same process for all the features (the original LBP, "uniform patterns", CS-LBP and the proposed OC-LBP). For each image in the training / test set, each of the operators is applied on all the pixels of the image to get their binary pattern values, and the histogram computed throughout the image is then used as its texture feature. The Support Vector Machine (SVM) algorithm is applied for classification. We compute the $\chi^2$ distance as equation (2.36) to measure the similarity between each pair of the feature vectors. Then, the kernel based on this distance is computed as equation (2.37) for the SVM training and prediction. Finally, each test image is classified into texture category with the maximum SVM output decision value. We tune the parameters of the classifier on the training set via 5-fold cross-validation, and obtain the classification results on the test set.

The classification results and comparisons are presented in Table 5.2. It can be seen that the classification accuracy generally keeps improving when the number of neighboring pixels increases, suggesting that the consideration of more neighbors can be beneficial to the operator's performance. However, the increment speed of histogram size for the original LBP is devastating. For example, the LBP histogram size with 20 neighboring pixels is so enormous that it is impractical to be used directly. This shows the importance of dimensionality reduction for LBP. The CS-LBP operator reduces the LBP histogram size to its square root, but it also decreases the classification accuracy. One possible reason is that it discards the information of central pixel in comparison. The "uniform patterns" show good performances, because it significantly reduces the LBP histogram size, while still keeping high discriminative power. Actually, it performs even a little better than the original LBP, because it only keeps the most important part of LBP and removes the other

Table 5.2: Comparison of different LBP dimensionality reduction methods in terms of histogram size and classification accuracy on Outex_TC_00014 ($P, R$ — $P$ neighboring pixels equally located on a circle of radius $R$)

| P,R | LBP | | Uniform patterns | | CS-LBP | | OC-LBP | |
|---|---|---|---|---|---|---|---|---|
| | Bins | Result | Bins | Result | Bins | Result | Bins | Result |
| 4,1 | 16 | 58.5% | 15 | 58.8% | 4 | 27.8% | 16 | 58.5% |
| 8,1 | 256 | 61.4% | 59 | 66.1% | 16 | 50.2% | 32 | 65.4% |
| 12,2 | 4096 | 68.7% | 135 | 72.4% | 64 | 61.8% | 48 | 72.7% |
| 16,2 | 65536 | 67.6% | 243 | 73.4% | 256 | 54.7% | 64 | 73.2% |
| 20,3 | 1048576 | – | 383 | 74.0% | 1024 | 55.7% | 80 | 74.6% |

disturbances. Compared to these two methods, the proposed OC-LBP operator is more effective, because it outperforms CS-LBP and achieves almost the same high performance as the "uniform patterns" but with the smallest histogram size among them. Therefore, the proposed OC-LBP is very suitable for local image region description.

## 5.3 Local region description with OC-LBP

We construct a new local region descriptor based on the proposed OC-LBP operator by following the way similar to the SIFT [Lowe 2004] and CS-LBP [Heikkilä *et al.* 2009] descriptors. Figure 5.2 depicts the construction process. The input of the descriptor is a normalized local image region around the keypoint, which is either detected by certain interest point detector such as Harris-Laplace, or located on a dense sampling grid. The OC-LBP operator is then applied on all the pixels in the region to get their binary pattern values. In order to include coarse spatial information, the region is equally divided into several small cells, within which a histogram is built based on the binary pattern values of all the pixels. The final descriptor is constructed by concatenating all the histograms from the cells. We adopt the uniform strategy for pixel weighting, as the CS-LBP descriptor, and a SIFT-like approach for descriptor normalization. The descriptor is firstly normalized to unit length, each value is then restricted to be no larger than 0.2 (threshold) so that the influence of very large values is reduced, and finally the descriptor is renormalized

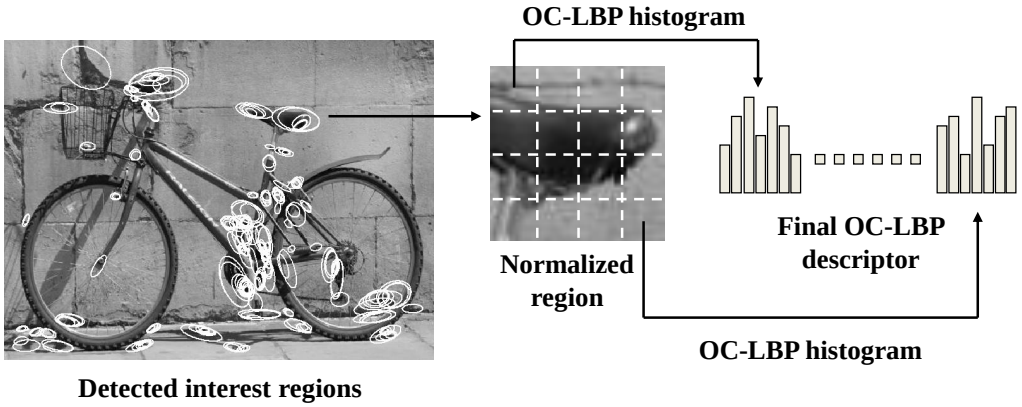



Figure 5.2: Construction of local image descriptor with OC-LBP

to unit length. We denote this new local image descriptor as OC-LBP descriptor.

## 5.4 Color OC-LBP descriptors

The classical LBP-related descriptors only use gray information. However, as we demonstrated in chapter 4, color information may significantly improve the discriminative power of a descriptor. Moreover, incorporating color information may enhance the photometric invariance properties when dealing with different kinds of illumination changes as described in section 4.2.

In order to incorporate color information, we further extend the OC-LBP descriptor to different color spaces and propose six color OC-LBP descriptors in this section. Following the similar way in chapter 4, the main idea is to calculate the original OC-LBP descriptor independently over different channels of a certain color space, and then concatenate them to get the final color OC-LBP descriptor, as shown in Figure 5.3.

Details of the proposed color OC-LBP descriptors and their properties are as follows:

**RGB-OC-LBP.** This color descriptor is obtained by computing the OC-LBP descriptor over all three channels of the *RGB* color space. It is invariant to monotonic light intensity change due to the property of the original OC-LBP descriptor.

**NRGB-OC-LBP.** This color descriptor is obtained by computing the OC-LBP

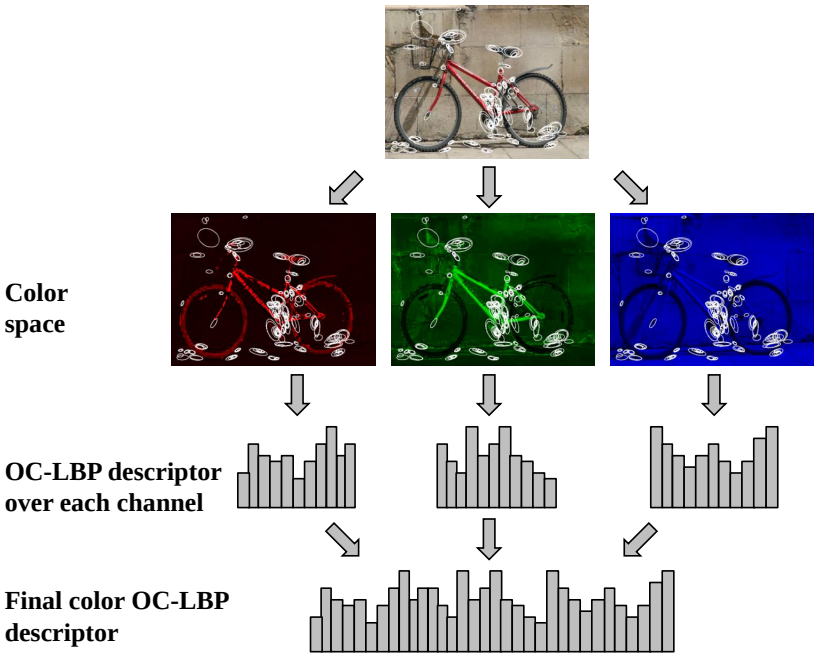



Figure 5.3: Calculation of color OC-LBP descriptor

descriptor over both $r$ and $g$ channels of the *normalized RGB* color space as equation (4.6) ($b$ channel is redundant because $r+g+b=1$). Due to the normalization, the change factors can be cancelled out if they are constant in all channels. Therefore, $r$ and $g$ channels are scale-invariant, which makes this descriptor invariant to light intensity change as equation (4.3).

**OPPONENT-OC-LBP.** This color descriptor is obtained by computing the OC-LBP descriptor over all three channels of the $OPPONENT$ color space as equation (4.8). Due to the subtraction in channel $O_1$ and $O_2$, the change offsets can be cancelled out if they are equal in all channels. Therefore, $O_1$ and $O_2$ channels are invariant to light intensity shift as equation (4.4). $O_3$ channel represents the intensity information, and has no invariance properties.

**NOPPONENT-OC-LBP.** This color descriptor is obtained by computing the OC-LBP descriptor over two channels of the *normalized* $OPPONENT$ color space as equation (4.10). Due to the normalization by intensity channel $O_3$, $O'_1$ and $O'_2$ channels are scale-invariant, which makes this descriptor invariant to light intensity change as equation (4.3).

**Hue-OC-LBP.** This color descriptor is obtained by computing the OC-LBP

descriptor over the $Hue$ channel of the $HSV$ color space as equation (4.11). Due to the subtraction and the division, $Hue$ channel is scale-invariant and shift-invariant, therefore this descriptor is invariant to light intensity change and shift as equation (4.5).

**TC-OC-LBP.** This color descriptor is obtained by computing the OC-LBP descriptor over all three channels of the $transformed$ color space as equation (4.12) ($\mu$ is the mean and $\sigma$ is the standard deviation of each channel). Due to the subtraction and the normalization, all three channels are scale-invariant and shift-invariant, which makes this descriptor invariant to light intensity change and shift as equation (4.5). Furthermore, because each channel is operated independently, this descriptor is also invariant to light color change and shift as equation (4.2).

It should be noticed that this descriptor has equal values to the RGB-OC-LBP descriptor. Because the LBP is computed by taking the subtraction of the neighboring pixels and the central one, the subtraction of the means in this color space is redundant, as this offset is already cancelled out when computing the LBP. And since the descriptor normalization for each channel is done separately, the division of the standard deviation is also redundant. Therefore, the RGB-OC-LBP descriptor is used in this chapter to represent both descriptors.

## 5.5 Experimental evaluation

We evaluated the proposed intensity-based and color OC-LBP descriptors in three different applications: (1)image matching, (2)object recognition and (3)scene classification. The proposed descriptors are compared with several state-of-the-art descriptors including SIFT [Lowe 2004], color SIFT [van de Sande *et al.* 2010], CS-LBP [Heikkilä *et al.* 2009], HOG [Dalal & Triggs 2005], SURF [Bay *et al.* 2008] and GIST [Oliva & Torralba 2001]. These descriptors have been chosen for their diversity in terms of local visual content characterization. While SIFT and color SIFT are the most popular and successful local descriptors in the literature, HOG is also a popular descriptor which captures local object appearance and shape through the distribution of intensity gradients. As such it is widely used for object detection

and recognition. GIST is a popular holistic feature which estimates the dominant spatial structure of a scene to capture a set of perceptual dimensions (naturalness, openness, roughness, expansion and ruggedness). As such it is widely applied for scene classification. SURF is a typical local descriptor using Haar wavelets as features. Finally, CS-LBP is also binary-pattern-based and provides a way for LBP dimensionality reduction, as introduced in section 5.2.

### 5.5.1 Parameter selection

There are three parameters to be fixed for the proposed OC-LBP descriptors, including the number of neighboring pixels for the OC-LBP operator ($P$), the radius of neighboring circle for the OC-LBP operator ($R$), and the number of cells for each region ($M \times M$). For simplicity, the parameters $P$ and $R$ are evaluated in pairs, such as (4,1), (8,1), (12,2), (16,2), (20,3), etc. Also, we select the parameters based on the gray OC-LBP descriptor, and apply the best settings on all color OC-LBP descriptors.

We adopt the standard Oxford image matching dataset [Visual Geometry Group ] for parameter selection. This dataset contains image pairs with different geometric and photometric transformations (image blur, viewpoint change, illumination change, etc.) and different scene types (structured and textured). The sample image pairs are shown in Figure 5.4. Here the image pair named "Graf" is used for parameter selection as in [Heikkilä *et al.* 2009]. To compute the descriptors, an interest region detector is required at first to detect interest regions in each image. We apply the Harris-Affine detector to detect the corner-like structures in images. It originally outputs the elliptic regions of varying scales, and all the regions are then normalized and mapped to a circular region with fixed radius to obtain scale and affine invariance. The normalized regions are also rotated to the direction of their dominant gradient orientations to obtain the rotation invariance. We use the software package available on the same website as the dataset for interest region detection and normalization. Each detected region is normalized to the size of $41 \times 41$ pixels. Then, all the regions from each image are described by the OC-LBP descriptor, and are matched by applying nearest

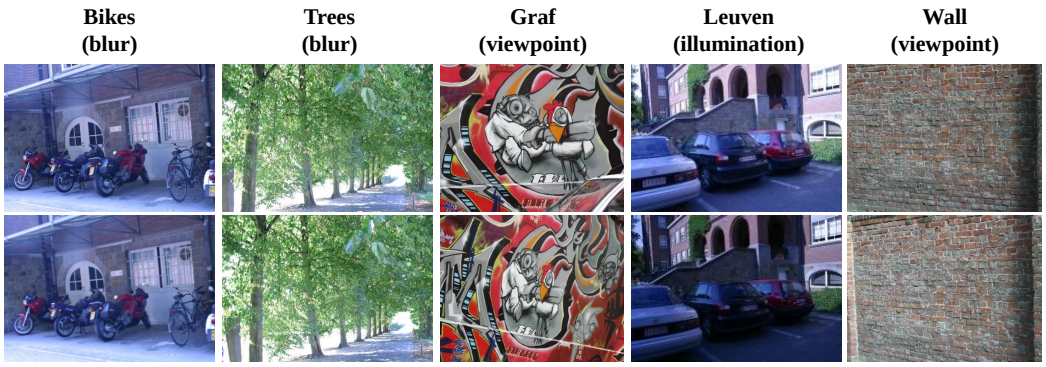



Figure 5.4: Sample image pairs of the Oxford dataset

Table 5.3: Parameter selection results (matching score %) for the OC-LBP descriptor

| P,R \ Cells | $1 \times 1$ | $2 \times 2$ | $3 \times 3$ | $4 \times 4$ | $5 \times 5$ |
|---|---|---|---|---|---|
| 4,1 | 2.84 | 19.11 | 25.43 | 25.77 | 25.48 |
| 8,1 | 8.76 | 26.79 | 34.07 | 32.88 | 31.23 |
| 12,2 | 13.77 | 33.56 | <u>39.31</u> | 36.75 | 34.64 |
| 16,2 | 11.43 | 32.48 | 38.74 | 35.67 | 33.56 |
| 20,3 | 13.03 | 34.47 | 38.91 | 37.26 | 34.41 |

neighbor strategy. A matching score is obtained by measuring the percentage of the correct matches.

From the results shown in Table 5.3, it can be seen that the best performance is obtained when the value of $(P, R)$ pair is set to $(12, 2)$ and the number of cells is set to $3 \times 3$. We apply this parameter setting on gray OC-LBP descriptor and all color OC-LBP descriptors in the following experiments.

### 5.5.2 Experiments on image matching

We adopt the same dataset introduced in section 5.5.1 to evaluate the proposed descriptors in the application of image matching. The performances of the descriptors are evaluated by the matching criterion, which is based on the number of correctly and falsely matched regions between a pair of images. Two image regions are considered to be matched if the Euclidean distance between their descriptors is below a threshold. The number of correct matches is determined by the "overlap error"

[Mikolajczyk & Schmid 2002]. A match is assumed to be correct if this error value is smaller than 0.5. The results are presented by recall versus 1-precision curve:

$$recall = \frac{\#correct\ matches}{\#correspondences} \tag{5.1}$$

$$1 - precision = \frac{\#false\ matches}{\#all\ matches} \tag{5.2}$$

where #correspondences is the ground truth number of matches between the images. By changing the distance threshold, we can obtain the recall versus 1-precision curve.

#### 5.5.2.1 Experimental setup

We use the software package mentioned in section 5.5.1 for interest region detection, region normalization, and SIFT computation. We implement the CS-LBP descriptor according to [Heikkilä *et al.* 2009], and apply the same parameter setting as the OC-LBP descriptor for fair comparison. To compute color SIFT descriptors, we use the "ColorDescriptor" software available online [Koen van de Sande ].

#### 5.5.2.2 Experimental results

The image matching results on the Oxford dataset are shown in Figure 5.5 and Figure 5.6. Figure 5.5 shows the comparisons of the proposed gray and color OC-LBP descriptors with the popular SIFT and CS-LBP descriptors. Figure 5.6 shows the comparisons of the best three color OC-LBP descriptors with the state-of-the-art color SIFT descriptors.

We can see from the results in Figure 5.5 that: (1) the OC-LBP descriptor performs better than the popular CS-LBP and SIFT descriptors; (2) the color OC-LBP descriptors outperform the intensity-based OC-LBP descriptor in most of the cases, proving the usefulness of incorporating color information and additional photometric invariance properties; (3) among the proposed color OC-LBP descriptors, Hue-OC-LBP, RGB-OC-LBP and NOPPONENT-OC-LBP descriptors have the best overall performance, consistent with their strong properties of illumination invariance.

We then compare the best three color OC-LBP descriptors with their counter-

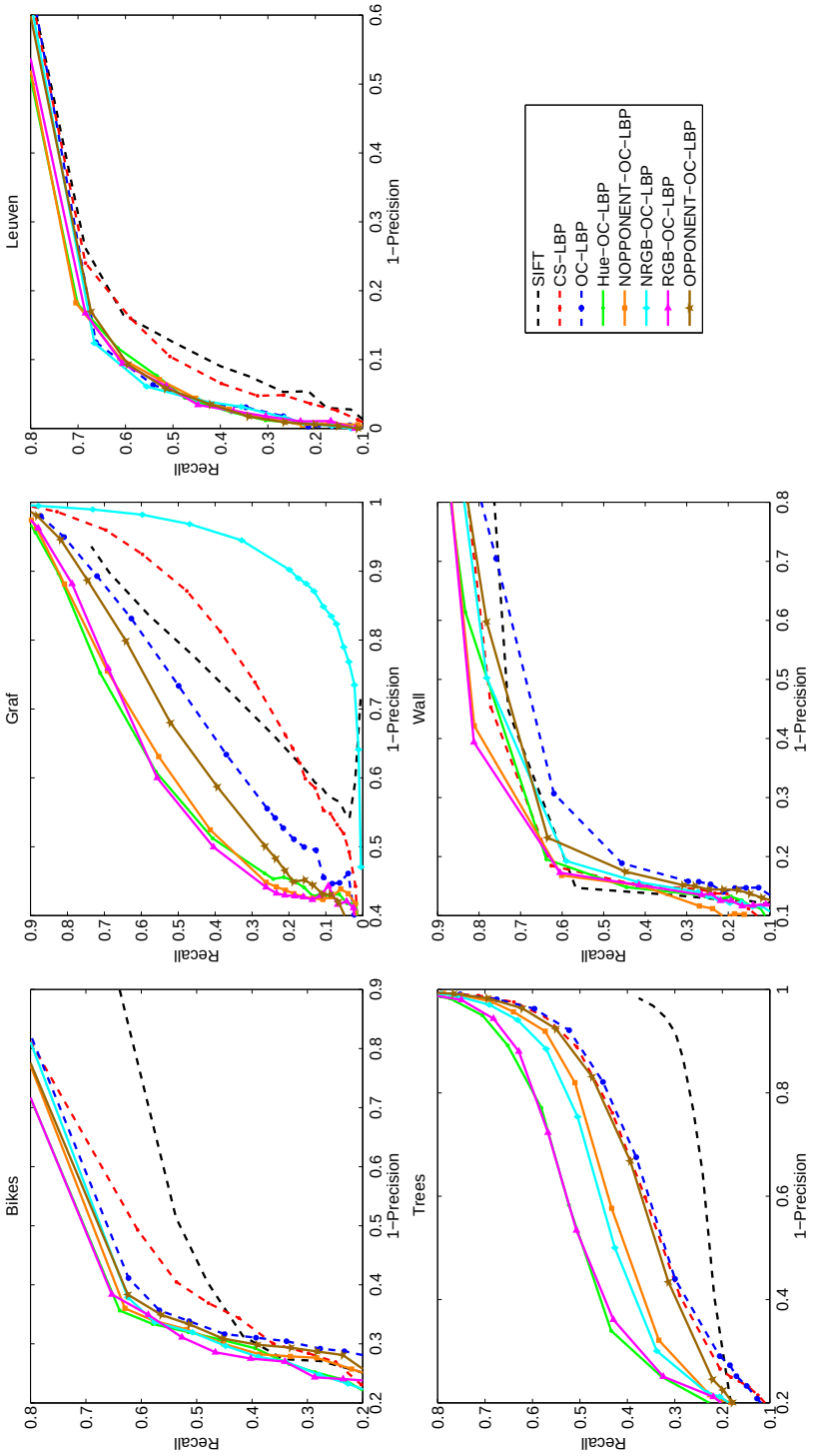



Figure 5.5: Image matching results on the Oxford dataset (comparisons of the proposed descriptors with the popular SIFT and CS-LBP descriptors)

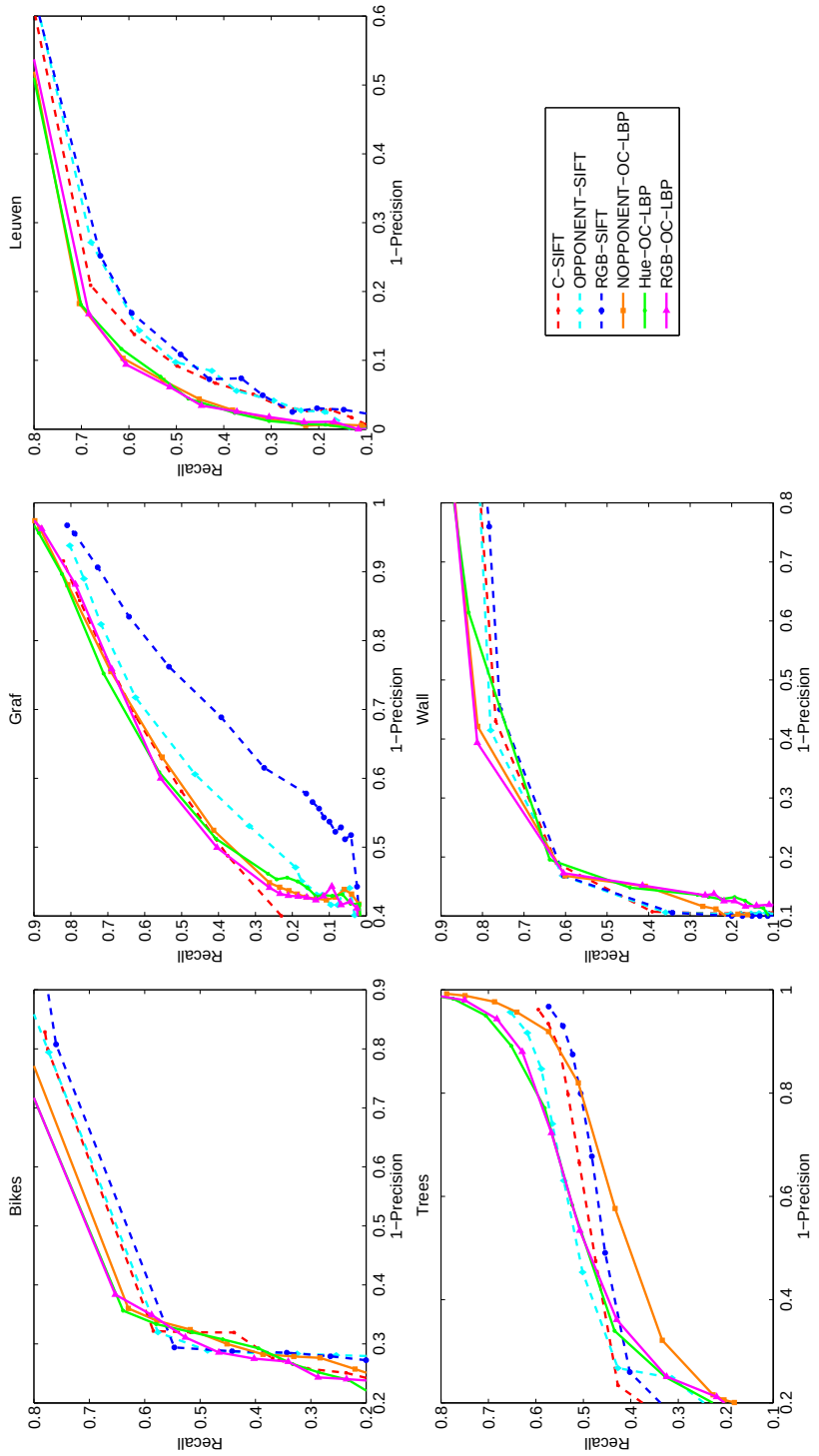



Figure 5.6: Image matching results on the Oxford dataset (comparisons of the best three color OC-LBP descriptors with the state-of-the-art color SIFT descriptors)

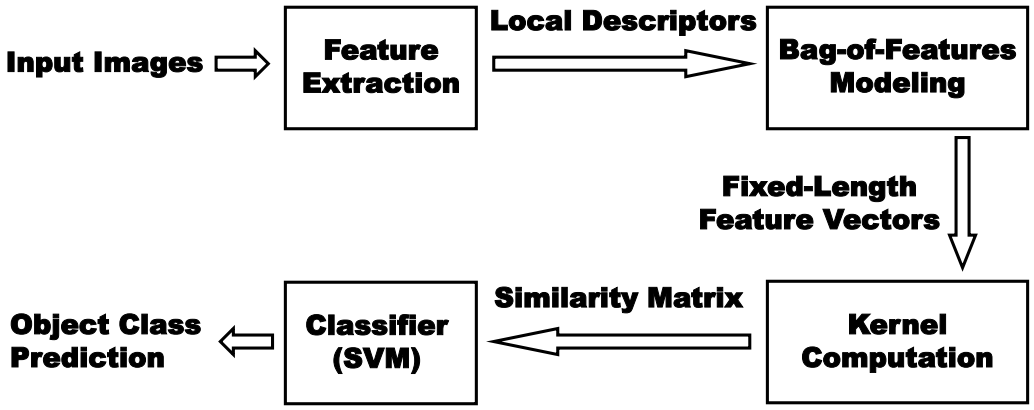



Figure 5.7: Flow chart of our approach for object recognition

parts, the state-of-the-art color SIFT descriptors. The best three color SIFT descriptors are chosen according to [van de Sande *et al.* 2010]. The results in Figure 5.6 show that the color OC-LBP descriptors also achieve slightly better performances than color SIFT.

### 5.5.3 Experiments on object recognition

In order to evaluate the proposed descriptors in the application of object recognition, two standard image datasets are used: the PASCAL VOC 2007 benchmark [Everingham *et al.* 2007] and the SIMPLIcity dataset [Wang *et al.* 2001]. A detailed introduction of both datasets can be found in chapter 3.

These two datasets have different characteristics. In the SIMPLIcity dataset, most images have little or no clutter. The objects tend to be centered in each image. Most objects are presented in a stereotypical pose. In the PASCAL VOC 2007 benchmark, all the images are taken from the real-world scenes, thus with background clutter, occlusions, and various variations in viewpoint, pose and lighting condition, which increase the difficulties of object recognition in this dataset.

#### 5.5.3.1 Our approach for object recognition

The block diagram of our approach for visual object recognition is depicted in Figure 5.7.

#### 5.5.3.2 Feature extraction

The interest points in images are firstly detected by applying the Harris-Laplace salient point detector, which uses a Harris corner detector and subsequently the Laplacian for scale selection. Then a set of local descriptors, including gray OC-LBP, three best color OC-LBP, CS-LBP, SURF, HOG, SIFT and three best color SIFT, are extracted from local region around each interest point. Unlike the settings in the application of image matching, the descriptors are not rotated to their dominant orientations, because this rotation invariance is useful for image matching, but decreases the accuracy for object recognition.

#### 5.5.3.3 Bag-of-Features modelling

After the step of feature extraction, each image is represented by a set of local descriptors. The number of local descriptors in each image varies because the number of the interest points (normally around thousands) changes from one image to another one. Thus, an efficient modeling method is required to transform this variable number of local descriptors into a more compact, informative and fixed length representation for further classification.

We apply the popular Bag-of-Features (BoF) method [Csurka *et al.* 2004] because of its great success in object recognition tasks. A detailed introduction of the BoF method can be found in section 2.2.2.3. Specifically, we build a vocabulary of 1000 "visual words" for the SIMPLIcity dataset and 4000 "visual words" for the PASCAL VOC 2007 benchmark for each kind of local descriptors respectively by applying the k-means clustering algorithm on a subset of the descriptors which are randomly selected from the training data.

#### 5.5.3.4 Classification

The Support Vector Machine (SVM) algorithm is applied for object classification. An introduction of SVM can be found in section 2.3.2.1. Here the LibSVM implementation [Chang & Lin 2001] is used. Once all the local descriptors are transformed to fixed-length feature vectors by the BoF method, the $\chi^2$ distance is comput-

Table 5.4: Object recognition results on the PASCAL VOC 2007 benchmark ("NOP-OC-LBP" is the abbreviation of "NOPPONENT-OC-LBP", "OP-SIFT" is the abbreviation of "OPPONENT-SIFT")

| AP (%) | OC-LBP | Hue-OC-LBP | NOP-OC-LBP | RGB-OC-LBP | CS-LBP | HOG | SURF | SIFT | OP-SIFT | C-SIFT | RGB-SIFT |
|---|---|---|---|---|---|---|---|---|---|---|---|
| airplane | 62.2 | 64.3 | 64.2 | 61.9 | 59.2 | 52.1 | 39.7 | 56.0 | 59.9 | 58.7 | 57.8 |
| bicycle | 38.6 | 35.4 | 39.1 | 42.0 | 44.8 | 26.9 | 45.9 | 44.9 | 43.8 | 38.9 | 44.6 |
| bird | 25.9 | 32.9 | 34.8 | 32.1 | 27.4 | 25.0 | 26.7 | 28.2 | 27.7 | 32.1 | 22.5 |
| boat | 56.4 | 56.0 | 60.8 | 59.5 | 53.0 | 40.6 | 21.0 | 45.7 | 49.1 | 51.8 | 46.6 |
| bottle | 15.0 | 20.4 | 20.0 | 20.3 | 19.5 | 12.8 | 10.2 | 19.6 | 21.2 | 21.4 | 21.0 |
| bus | 37.8 | 35.5 | 35.0 | 41.1 | 33.2 | 38.3 | 28.1 | 37.7 | 38.0 | 32.5 | 37.7 |
| car | 62.6 | 60.5 | 61.4 | 65.1 | 63.1 | 58.1 | 52.5 | 55.0 | 57.4 | 53.2 | 56.1 |
| cat | 38.9 | 39.3 | 39.7 | 42.9 | 40.2 | 27.5 | 24.3 | 36.5 | 37.7 | 34.1 | 37.3 |
| chair | 39.0 | 40.5 | 41.3 | 39.3 | 38.7 | 43.8 | 33.3 | 44.5 | 42.4 | 45.9 | 43.5 |
| cow | 20.6 | 21.5 | 14.6 | 24.9 | 18.3 | 19.8 | 20.8 | 25.9 | 17.0 | 16.6 | 27.8 |
| table | 35.0 | 36.1 | 37.0 | 32.0 | 33.1 | 33.6 | 25.7 | 29.6 | 36.7 | 38.7 | 29.1 |
| dog | 32.8 | 35.3 | 29.4 | 33.4 | 31.7 | 20.4 | 23.8 | 26.5 | 29.8 | 29.1 | 28.8 |
| horse | 57.6 | 64.6 | 63.6 | 58.3 | 55.2 | 59.3 | 50.7 | 57.0 | 59.1 | 61.9 | 54.8 |
| motor | 36.9 | 39.2 | 41.7 | 37.3 | 34.1 | 37.2 | 37.4 | 30.2 | 33.9 | 44.4 | 32.1 |
| person | 74.1 | 77.2 | 75.5 | 74.7 | 73.0 | 66.2 | 70.8 | 73.1 | 74.5 | 76.6 | 72.7 |
| plant | 21.3 | 22.7 | 26.7 | 20.1 | 17.5 | 10.4 | 13.8 | 11.5 | 19.9 | 27.1 | 11.5 |
| sheep | 12.3 | 23.5 | 26.0 | 19.9 | 16.9 | 18.4 | 9.4 | 27.4 | 31.2 | 30.9 | 19.4 |
| sofa | 25.8 | 27.8 | 27.5 | 25.0 | 19.0 | 26.3 | 19.3 | 23.6 | 22.9 | 23.2 | 24.6 |
| train | 56.1 | 44.2 | 51.7 | 55.5 | 56.8 | 52.7 | 42.9 | 53.4 | 54.5 | 58.5 | 51.1 |
| monitor | 25.6 | 29.2 | 27.9 | 31.8 | 31.7 | 32.3 | 25.7 | 33.7 | 35.0 | 27.3 | 35.6 |
| **Mean** | **38.7** | **40.3** | **40.9** | **40.9** | **38.3** | **35.1** | **31.1** | **38.0** | **39.6** | **40.1** | **37.7** |

ed as equation (2.36) to measure the similarity between each pair of feature vectors. Then, the kernel function based on this distance is computed as equation (2.37) for the SVM training and prediction.

For the SIMPLIcity dataset, each image is classified into the category with the maximum SVM output decision value. We tune the parameters of the classifier on the training set via 5-fold cross-validation, and obtain the results on the test set. For the PASCAL VOC 2007 benchmark, the precision-recall curve is plotted for each category according to the output decision values of the SVM classifier, and the AP (Average Precision) value is computed based on the proportion of the area under this curve. We train the classifier on the training set, then tune the parameters on the validation set, and obtain the classification results on the test set.

Table 5.5: Fusion results of color OC-LBP and color SIFT on the PASCAL VOC 2007 benchmark

| AP (%) | FUSION (3 Color OC-LBP) | FUSION (3 Color SIFT) | FUSION (3 Color OC-LBP +3 Color SIFT) |
|---|---|---|---|
| airplane | 67.0 | 61.8 | 67.8 |
| bicycle | 48.0 | 49.8 | 56.4 |
| bird | 36.7 | 35.0 | 43.4 |
| boat | 62.2 | 52.9 | 60.9 |
| bottle | 17.6 | 23.6 | 26.2 |
| bus | 46.4 | 44.4 | 51.3 |
| car | 67.8 | 61.7 | 68.6 |
| cat | 45.8 | 41.7 | 46.2 |
| chair | 43.6 | 48.2 | 48.6 |
| cow | 26.9 | 29.1 | 29.2 |
| table | 43.2 | 41.8 | 48.2 |
| dog | 35.8 | 32.9 | 39.3 |
| horse | 64.9 | 64.8 | 69.6 |
| motor | 46.1 | 48.3 | 53.3 |
| person | 77.8 | 77.3 | 79.2 |
| plant | 27.3 | 26.5 | 31.3 |
| sheep | 24.3 | 33.8 | 31.7 |
| sofa | 32.4 | 30.6 | 37.5 |
| train | 60.1 | 62.9 | 68.3 |
| monitor | 35.1 | 38.1 | 39.5 |
| **Mean** | **45.5** | **45.3** | **49.8** |

#### 5.5.3.5 Experimental results on PASCAL VOC 2007

The object recognition results on the PASCAL VOC 2007 benchmark are shown in Table 5.4. It can be seen that: (1) the proposed OC-LBP descriptor achieves the performance of 38.7% MAP, which is better than SURF and HOG, and comparable with CS-LBP and SIFT; (2) the best three color OC-LBP descriptors (Hue-OC-LBP, NOPPONENT-OC-LBP and RGB-OC-LBP) achieve 40.3%, 40.9% and 40.9% MAP respectively, which outperform the intensity-based OC-LBP by about 2% $\sim$ 3%, indicating that they truly benefit from additional color information and illumination invariance properties; (3) compared to the state-of-the-art color SIFT descriptors, the best three color OC-LBP descriptors achieve comparable or even better results.

After analyzing the detailed results in Table 5.4 by each object category, we

could observe that the LBP-based descriptors generally perform better on the non-rigid object categories such as bird, cat, dog, horse, person, plant and sofa, while the SIFT-based descriptors are generally better for the rigid object categories such as bicycle, bottle, chair, table, motor, train and monitor. Also, the color descriptors with different photometric invariance properties perform differently on the same object category. Therefore, we further combine different color OC-LBP descriptors, as well as color OC-LBP and color SIFT by average late fusion to check if they can provide complementary information to each other. The fusion results are shown in Table 5.5.

It can be observed that: (1) a great performance improvement (about 5%) can be obtained by fusing different color descriptors, both for OC-LBP and SIFT, proving that different color descriptors are not entirely redundant; (2) the color OC-LBP descriptors still achieve comparable or slightly better results than color SIFT after fusion; (3) the performance can be further improved (more than 4%) by fusing color OC-LBP and color SIFT, indicating that these two kinds of descriptors can provide complementary information to each other.

#### 5.5.3.6 Experimental results on SIMPLIcity

The object recognition results on the SIMPLIcity dataset are shown in Table 5.6 and Table 5.7. The similar observations to that on the PASCAL VOC benchmark can be noticed. The color OC-LBP descriptors outperform CS-LBP, SURF, HOG, SIFT as well as the intensity-based OC-LBP, and achieve comparable results with the color SIFT descriptors. Further improvement (nearly 5%) can be obtained by fusing three color OC-LBP and three color SIFT descriptors, since they provide complementary information to each other.

### 5.5.4 Experiments on scene classification

We also evaluated the proposed descriptors in the application of scene classification. The dataset from Oliva and Torralba [Oliva & Torralba 2001] is used, and denoted as OT scene dataset. Its detailed introduction can be found in section 3.6.

Table 5.6: Object recognition results on the SIMPLIcity dataset ("NOP-OC-LBP" is the abbreviation of "NOPPONENT-OC-LBP", "OP-SIFT" is the abbreviation of "OPPONENT-SIFT")

| Accuracy (%) | OC-LBP | Hue-OC-LBP | NOP-OC-LBP | RGB-OC-LBP | CS-LBP | HOG | SURF | SIFT | OP-SIFT | C-SIFT | RGB-SIFT |
|---|---|---|---|---|---|---|---|---|---|---|---|
| people | 70.0 | 84.0 | 80.0 | 78.0 | 70.0 | 58.0 | 72.0 | 76.0 | 76.0 | 84.0 | 74.0 |
| beach | 74.0 | 82.0 | 86.0 | 76.0 | 82.0 | 68.0 | 76.0 | 82.0 | 88.0 | 86.0 | 82.0 |
| building | 82.0 | 86.0 | 84.0 | 82.0 | 80.0 | 66.0 | 66.0 | 74.0 | 78.0 | 74.0 | 70.0 |
| bus | 98.0 | 96.0 | 96.0 | 98.0 | 88.0 | 90.0 | 92.0 | 94.0 | 96.0 | 90.0 | 96.0 |
| dinosaur | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| elephant | 74.0 | 70.0 | 72.0 | 72.0 | 80.0 | 70.0 | 78.0 | 88.0 | 84.0 | 74.0 | 94.0 |
| flower | 82.0 | 94.0 | 88.0 | 86.0 | 88.0 | 58.0 | 70.0 | 92.0 | 96.0 | 86.0 | 88.0 |
| horse | 98.0 | 98.0 | 98.0 | 96.0 | 96.0 | 92.0 | 82.0 | 96.0 | 98.0 | 100.0 | 94.0 |
| mountain | 68.0 | 68.0 | 74.0 | 68.0 | 64.0 | 64.0 | 50.0 | 62.0 | 70.0 | 72.0 | 70.0 |
| food | 88.0 | 92.0 | 100.0 | 96.0 | 80.0 | 72.0 | 78.0 | 86.0 | 88.0 | 94.0 | 90.0 |
| **Mean** | **83.4** | **87.0** | **87.8** | **85.2** | **82.8** | **73.8** | **76.4** | **85.0** | **87.4** | **86.0** | **85.8** |

Table 5.7: Fusion results of color OC-LBP and color SIFT on the SIMPLIcity dataset

| Accuracy (%) | FUSION (3 Color OC-LBP) | FUSION (3 Color SIFT) | FUSION (3 Color OC-LBP +3 Color SIFT) |
|---|---|---|---|
| people | 86.0 | 86.0 | 86.0 |
| beach | 86.0 | 88.0 | 86.0 |
| building | 86.0 | 78.0 | 86.0 |
| bus | 100.0 | 98.0 | 100.0 |
| dinosaur | 100.0 | 100.0 | 100.0 |
| elephant | 82.0 | 90.0 | 86.0 |
| flower | 98.0 | 100.0 | 98.0 |
| horse | 98.0 | 100.0 | 100.0 |
| mountain | 78.0 | 76.0 | 82.0 |
| food | 96.0 | 96.0 | 98.0 |
| **Mean** | **91.0** | **91.2** | **92.2** |

#### 5.5.4.1 Experimental setup

For this scene classification problem, our approach is the same as the one used for object recognition, as described in section 5.5.3.1, but with a different setting. Instead of detecting interest points in images using the Harris-Laplace detectors, we apply the dense sampling strategy to locate keypoints for local descriptor computation. This is because for scene classification, we prefer to focus on the content of the whole image, rather than on "object" part only. Specifically, the sampling spacing is set to 6 pixels, resulting in around 1700 keypoints per image. A visual vocabulary of 2000 "visual words" is constructed for each kind of local descriptor to build their Bag-of-Features (BoF) representations.

We randomly choose half of the images from each scene category for training, and the other half for test. The recognition accuracy is used as the evaluation criterion. We tune the parameters of the classifier on the training set via 5-fold cross-validation, and get the classification results on the test set.

#### 5.5.4.2 Experimental results

The classification results on the OT scene dataset are shown in Figure 5.8. It can be seen that the proposed OC-LBP descriptor performs better than SURF, and achieves comparable results with GIST, CS-LBP and SIFT. The proposed color OC-LBP descriptors further demonstrate their effectiveness as they display superior performances than all the intensity-based descriptors. They also show their ability of being complementary to the state-of-the-art color SIFT descriptors, since their fusion (fusion 3 in the figure) clearly improves the performance. It is worthy to notice that the NOPPONENT-OC-LBP descriptor does not perform well in this case, while its performance is quite good in the application of object recognition. We believe the main reason is that the OT scene dataset contains more varieties of illumination changes than the object recognition datasets, and the NOPPONENT-OC-LBP descriptor is deficient in power of dealing with these variations, because it is only invariant to light intensity change. This also explains why RGB-OC-LBP and RGB-SIFT perform the best among the color descriptors, since they possess

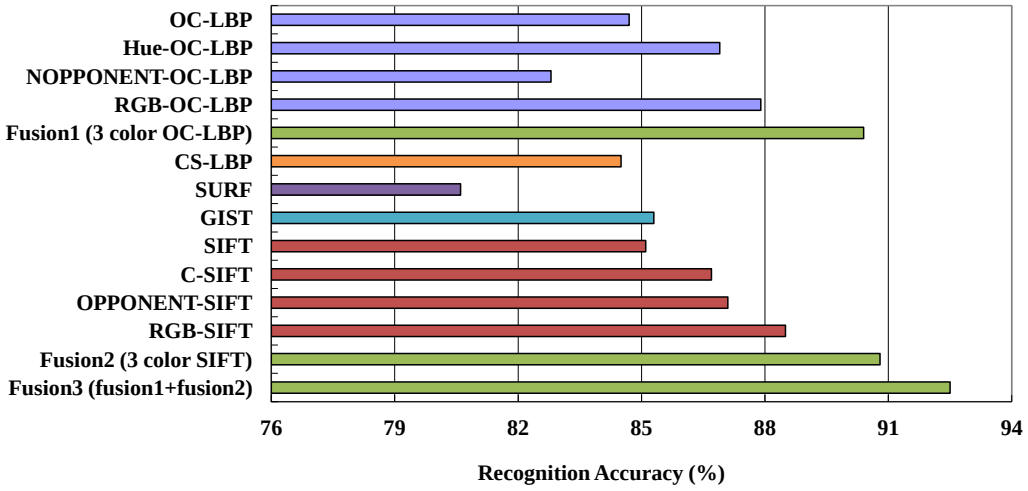



Figure 5.8: Classification results on the OT scene dataset

the strongest invariance properties (invariant to light color change and shift).

### 5.5.5 Computational cost comparison between descriptors

As we stated in the introduction, a good local descriptor should be both discriminative and computationally efficient. The discriminative power of the proposed gray and color OC-LBP descriptors has been demonstrated by the previous experiments and applications, and they achieve comparable or even better performances than the state-of-the-art descriptors. In this section, we show the computational efficiency of the proposed descriptors in comparison with the popular SIFT and color SIFT.

The comparisons are conducted on the 4 image datasets used in the previous experiments by utilizing a computer with Intel Core 2 Duo CPU @ 3.16 GHz and 3GB RAM. We implement the gray and color OC-LBP descriptors by a mixture of C and Matlab, and use the "ColorDescriptor" software [Koen van de Sande ] to compute the SIFT and color SIFT descriptors. We record in Table 5.8 the average computation time required per image for each descriptor respectively.

It can be seen that the OC-LBP descriptor is about 4 times faster to compute than SIFT. When incorporating color information, the computations of color descriptors are about 3 times slower than the intensity-based descriptors, mainly because

Table 5.8: Computational cost comparison between OC-LBP and SIFT descriptors

| Times (s) | Oxford (900×600) | SIMPLIcity (384 × 256) | PASCAL (500×375) | OT Scene (256×256) |
|---|---|---|---|---|
| OC-LBP | 0.273 | 0.062 | 0.101 | 0.042 |
| Hue-OC-LBP | 1.065 | 0.197 | 0.317 | 0.137 |
| NOPPONENT-OC-LBP | 0.889 | 0.181 | 0.296 | 0.117 |
| RGB-OC-LBP | 0.676 | 0.178 | 0.288 | 0.115 |
| SIFT | 1.064 | 0.328 | 0.432 | 0.161 |
| C-SIFT | 3.304 | 0.975 | 1.311 | 0.488 |
| OPPONENT-SIFT | 3.196 | 0.959 | 1.297 | 0.483 |
| RGB-SIFT | 3.147 | 0.955 | 1.282 | 0.477 |
| Total (3 color OC-LBP) | 2.630 | 0.556 | 0.901 | 0.369 |
| Total (3 color SIFT) | 9.647 | 2.889 | 3.890 | 1.448 |

of the increasing channels. However, the color OC-LBP descriptors are still about 4 times faster than color SIFT. Therefore, the proposed descriptors are much more computationally efficient, and thus are more suitable for large scale problems.

## 5.6 Conclusions

In this chapter, a new operator called the orthogonal combination of local binary patterns, denoted as OC-LBP, has firstly been proposed. It aims at reducing the dimensionality of the original LBP operator while keeping its discriminative power and computational efficiency.

We have also introduced several new local descriptors for image region description based on the proposed OC-LBP operator: the gray OC-LBP descriptor and six color OC-LBP descriptors, namely RGB-OC-LBP, NRGB-OC-LBP, OPPONENT-OC-LBP, NOPPONENT-OC-LBP, Hue-OC-LBP and TC-OC-LBP. The proposed descriptors incorporate color information to increase their discriminative power, and also to enhance their photometric invariance properties of dealing with different illumination changes.

The experiments in three different applications — image matching, object recognition and scene classification — show the effectiveness of the proposed descriptors. They outperform the popular SIFT, CS-LBP, HOG and SURF descriptors,

and achieve comparable or even better performances than the state-of-the-art color SIFT descriptors. Meanwhile, they provide complementary information to SIFT, since further improvement can be obtained by fusing them.

Moreover, the proposed gray and color OC-LBP descriptors are about 4 times faster to compute than the SIFT and color SIFT descriptors respectively. Therefore, they are very promising for large scale recognition problems.

Chapter 6

# Visual Object Recognition Using the DAISY Descriptor

Contents



## 6.1 Introduction

As we stated in chapter 5, visual content description is a key issue for the task of machine-based visual object recognition. A good visual descriptor should be both discriminative and computationally efficient, while possessing some properties of

robustness to changes in viewpoint, scale and lighting conditions. The recent literature has featured the gradient-distribution-based local descriptors, such as SIFT [Lowe 2004], GLOH [Mikolajczyk & Schmid 2005] and HOG [Dalal & Triggs 2005], as the main trend in object recognition tasks. Among them, SIFT is considered as the most powerful and successful one, and has been widely applied as the dominant feature in the state-of-the-art recognition/classification systems [Everingham *et al.* 2010]. The classic SIFT is a sparse descriptor computed on a set of points of interest (or keypoints) in images. However, several studies [Li & Perona 2005] [Furuya & Ohbuchi 2009] have shown that dense SIFT (SIFT computed on a dense grid) performs better than the original one for the task of object recognition.

There is now a trend in computer vision community that the scale of the benchmark datasets used for object recognition / image classification becomes larger year by year. However, it is well known that the downside of the state-of-the-art descriptors, including SIFT, GLOH, HOG, etc., is their relatively high computational cost, especially when the size of image or the scale of dataset significantly increases. Therefore, more computationally efficient and discriminative local descriptors are urgently demanded to deal with large scale datasets such as ImageNet [Deng *et al.* 2009] and TRECVID [Smeaton *et al.* 2006].

Usually, there are two ways to do this. One way is to replace the costly gradient information with other more efficient features, like LBP, as what we did in the case of the OC-LBP descriptor in chapter 5. The other way is to find more efficient methods to calculate the gradient information.

The DAISY descriptor [Tola *et al.* 2010], which was initially designed for wide-baseline stereo matching problem, is a newly introduced fast local descriptor based on gradient distribution, and has shown good robustness against many photometric and geometric transformations. It has never been used in the context of visual object recognition, while we believe that it is very suitable for this problem, and could well meet the mentioned demand. Therefore, in this chapter, we investigate the DAISY descriptor for the task of visual object recognition by evaluating and comparing it with the state-of-the-art SIFT both in terms of recognition accuracy and com-

putation complexity on two standard image datasets: Caltech 101 [Li *et al.* 2007] and PASCAL VOC 2007 [Everingham *et al.* 2007]. DAISY provides a fast way to calculate the gradient information and proves very promising for the task of visual object recognition.

## 6.2 The DAISY descriptor

Similar to SIFT, the DAISY descriptor is a 3D histogram of gradient locations and orientations. The differences between them lie in two aspects. One is that DAISY replaces the weighted sums of gradient norms used in SIFT by convolutions of gradients in specific directions with several Gaussian filters. This is for computing descriptor efficiently at every pixel location, because the histograms only need to be computed once per region and could be reused for all neighboring pixels. The other is that DAISY uses a circular neighborhood configuration instead of the rectangular one used in SIFT, as the comparison shown in Figure 6.1.

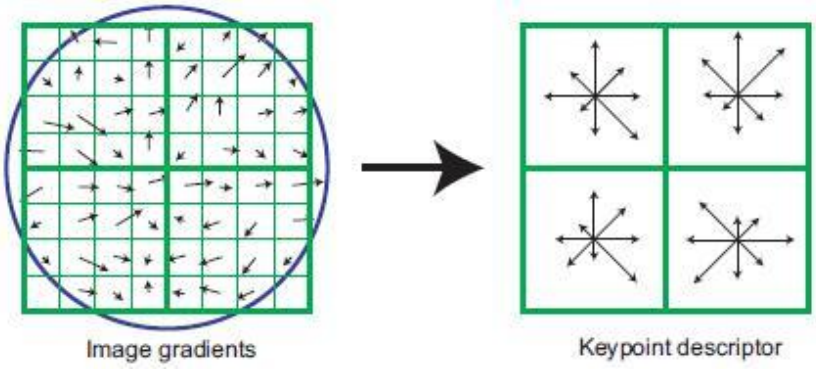
Given an input image $I$, a certain number of orientation maps $G_o$, one for each quantized direction $o$, are first computed. They are formally defined as:

$$G_o = \left(\frac{\partial I}{\partial o}\right)^+ \tag{6.1}$$

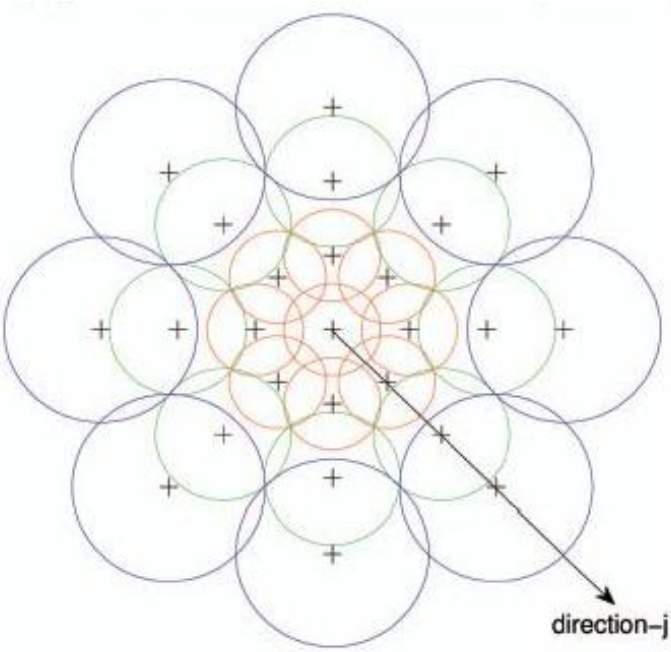
The + sign means that only positive values are kept to preserve the polarity of the intensity changes.

Each orientation map, which represents the image gradient norms for that direction at all pixel locations, is then convolved several times with Gaussian kernels of different standard deviation values to obtain the convolved orientation maps. The efficiency of DAISY descriptor comes right here, because Gaussian filters are separable and thus the convolutions can be implemented very efficiently. This means the convolutions with large Gaussian kernel can be obtained from several consecutive convolutions with smaller kernels. The computational amount is thus reduced.

At each pixel location, its neighborhood is divided into circles of different size located on a series of concentric rings, as shown in Figure 6.1(b). The radius of




(a)




(b)

Figure 6.1: Comparison of SIFT and DAISY shapes. (a) SIFT uses a rectangular grid [Lowe 2004]. (b) DAISY considers a circular configuration [Tola *et al.* 2010], where the radius of each circle is proportional to its distance from the center.

each circle is proportional to its distance from the central pixel, and the standard deviation of Gaussian kernel is proportional to the size of the circle. A vector is then made within each circle by gathering the values of all the convolved orientation maps with corresponding Gaussian smoothing. The final DAISY descriptor is made by concatenating all the vectors from the circles, after they are normalized to unit norm.

There are mainly four parameters to determine the shape of the DAISY descriptor: neighborhood area radius (R); number of quantized orientations (o); number of convolved orientation rings (r); and number of circles on each ring (c). The influence of different parameters will be analyzed experimentally in section 6.4.

## 6.3 Approach for visual object recognition

The approach applied in this chapter for visual object recognition is similar to the one introduced in section 5.5.3. The block diagram of the approach is depicted in Figure 5.7.

### 6.3.1 Feature extraction

We extract the DAISY and SIFT descriptors from input images as their features. The original DAISY descriptor introduced in section 6.2 is designed for wide-baseline stereo matching, so it is computed at every pixel location, leading to a very high dimensional descriptor. For example, a $500 \times 350$ image will yield a DAISY descriptor with the size of $175000 \times 200$ by default. Such high dimension is impractical for the task of object recognition because of the huge computation and storage requirements, especially for large images and datasets.

Therefore, we extract the DAISY descriptor on a dense grid for our purpose. Instead of at every pixel location, it is only computed on a dense sampling grid, which is the same as how the dense SIFT descriptor is computed. The sampling spacing is the parameter to control the number of sampling points. By this way, the dimension of the DAISY descriptor is reduced significantly, making it suitable to visual object recognition tasks.

### 6.3.2 Bag-of-Features modelling

To transform the extracted local descriptors (DAISY or SIFT) into a more compact, informative and fixed-length representation for further classification, we apply the popular Bag-of-Features (BoF) method [Csurka *et al.* 2004] because of its great success in object recognition tasks. A detailed introduction of the BoF method is given in section 2.2.2.3.

Since the BoF method ignores all spatial information of local descriptors, we also apply the spatial pyramid [Lazebnik *et al.* 2006] technique (see section 2.2.2.3 for a detailed introduction) to take into account coarse spatial relationship between them.

### 6.3.3 Classification

The Support Vector Machine (SVM) algorithm is applied for object classification. An introduction of SVM can be found in section 2.3.2.1. Once all local descriptors are transformed to fixed-length feature vectors by the BoF method, the $\chi^2$ distance is computed as equation (2.36) to measure the similarity between each pair of the feature vectors. Then, the kernel function based on this distance is computed as equation (2.37) for the SVM training and prediction. Finally, for each test image, the output probabilities of the SVM classifier are used to predict the object categories.

## 6.4 Experimental evaluation

In order to evaluate the performance of the DAISY descriptor, as well as to compare it with the state-of-the-art SIFT descriptor in the context of object recognition, we use two standard image datasets: Caltech 101 [Li *et al.* 2007] and PASCAL VOC 2007 [Everingham *et al.* 2007]. A detailed introduction of both datasets can be found in chapter 3.

These two datasets have different characteristics. In Caltech 101, most images have little or no clutter. The objects tend to be centered in each image. Most objects are presented in a stereotypical pose. In PASCAL VOC 2007, all the images are taken from the real-world scenes, thus with background clutter, occlusions, and

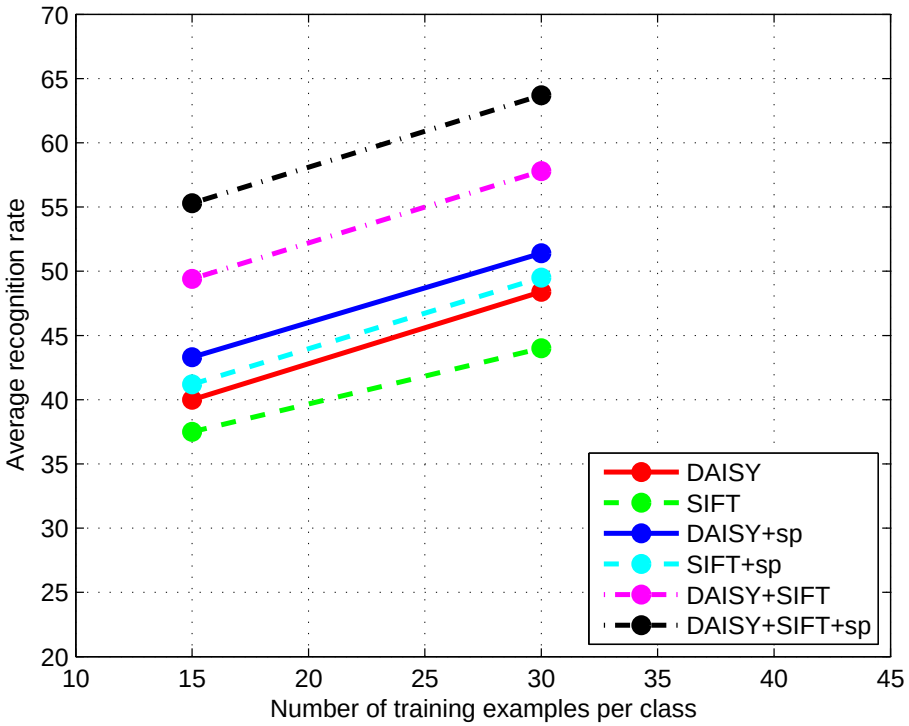



Figure 6.2: Experimental results on the Caltech 101 dataset ("sp" is the abbreviation for "spatial pyramid")

various variations in viewpoint, pose and lighting condition, which increase the difficulties of object recognition in this dataset.

### 6.4.1 Experimental setup

We follow the approach described in section 6.3 for both datasets. The DAISY and SIFT descriptors are extracted on the same dense grid for fair comparison. The sampling spacing is set to 6 pixels, resulting in around 2000 and 5000 descriptors per image for Caltech 101 and PASCAL VOC 2007 respectively. The parameter setting of 15R8o3r4c is applied for the DAISY descriptor (see section 6.4.4 for reasons), resulting in a 104-dimensional descriptor. A visual vocabulary with 1000 (for Caltech 101) or 4000 (for PASCAL VOC 2007) "visual words" is then constructed by applying k-means clustering algorithm to 600 000 randomly selected descriptors from the training set. Each image is finally represented by a fixed-length BoF histogram. A

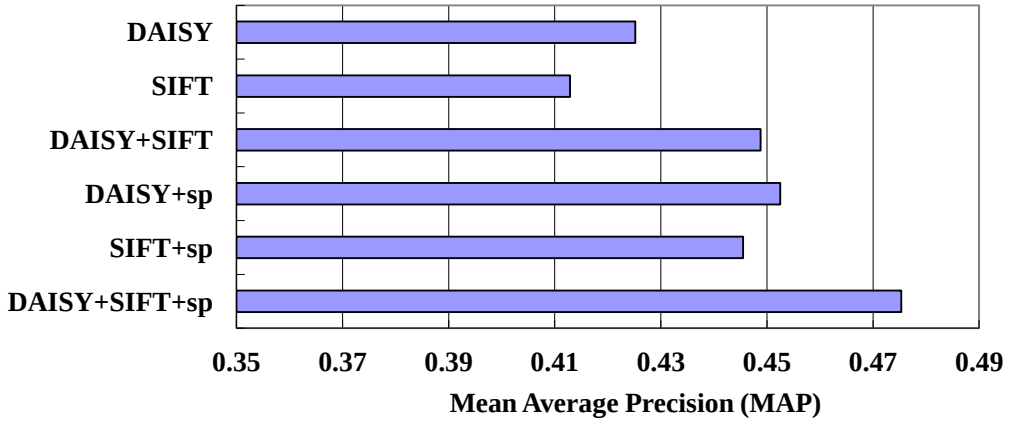



Figure 6.3: Experimental results on the PASCAL VOC 2007 dataset ("sp" is the abbreviation for "spatial pyramid")

$1 \times 1$ (whole image) + $2 \times 2$ (four equal quarters) + $3 \times 1$ (three equal horizontal bars) combination is applied for spatial pyramid. The LibSVM implementation [Chang & Lin 2001] of the SVM algorithm is used to perform the classification.

### 6.4.2 Results on Caltech 101

For the Caltech 101 dataset, we follow the common training and testing settings. Two training sets are constructed respectively by randomly selecting 15 or 30 images per category. Another 15 images are randomly selected per category for test (except for categories including less than 45 images). Each test image is classified into the category with the maximum SVM output decision value. We tune the parameters of the classifier on the training set via 5-fold cross-validation, and obtain the classification results on the test set. The experiments are repeated three times with different training and test sets, and average recognition accuracy is reported. The results are shown in Figure 6.2.

As we can see from the results, the recognition accuracy is improved for 2.5% (15 training) and 4.5% (30 training) respectively by using DAISY instead of SIFT. When spatial pyramid information is taken into account, the performances of DAISY and SIFT are both improved. But still, DAISY outperforms SIFT by 2.1% on average. Furthermore, when we combine DAISY and SIFT together by multiple kernel learn-

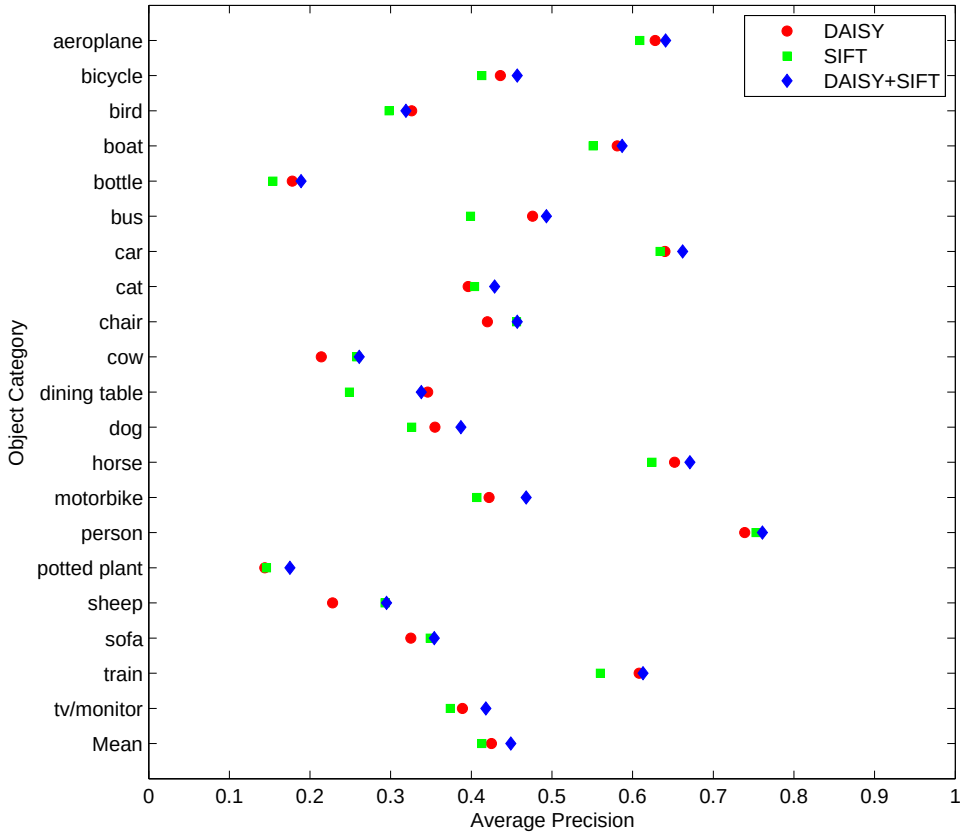



Figure 6.4: Performance comparison of DAISY and SIFT on the PASCAL VOC 2007 dataset split out per category

ing (MKL) [Rakotomamonjy *et al.* 2008] algorithm introduced in section 2.3.2.2, the recognition accuracy is improved significantly for 9.5% (15 training) and 12.1% (30 training), indicating that both descriptors can provide complementary information to each other.

### 6.4.3 Results on PASCAL VOC 2007

For the PASCAL VOC 2007 dataset, the precision-recall curve is plotted for each category according to the output decision values of the classifier, and the AP (Average Precision) value is computed based on the proportion of the area under this curve. We train the classifier on the training set, then tune the parameters on the

validation set, and obtain the classification results on the test set. The results are shown in Figure 6.3.

As we can see, similar to the results on Caltech 101, the performance of DAISY is better than that of SIFT, although the lead drops a little because the PASCAL VOC 2007 dataset is more challenging. Figure 6.4 shows the performance comparison of both descriptors split out per category. It can be seen that DAISY is better for some classes like plane, bike, bus, table, train, etc, while SIFT is better for other classes like chair, cow, person, plant, sheep, sofa, etc. This proves the complementarities of both descriptors, and explains why the performance can be improved by fusing them.

### 6.4.4 Influence of parameters in DAISY

As described in section 6.2, there are mainly 4 parameters to control the DAISY descriptor: neighborhood area radius (R); number of quantized orientations (o); number of convolved orientation rings (r); and number of circles on each ring (c). The influences of different parameters are evaluated experimentally on the Caltech 101 dataset. To do this, we obtain a series of line graphs of recognition accuracy by alternately changing one parameter while fixing the others. To keep the scales of different orientation rings, we set R as 5 for 1 ring, R as 10 for 2 rings, and R as 15 for 3 rings. The results are shown in Figure 6.5, Figure 6.6 and Figure 6.7.

The following conclusions can be made: 8 orientations perform clearly better than 4, while 12 show no superiority to 8, indicating that 8 orientations are sufficient; the performance keeps improving as the number of rings increases, showing that more rings are better, since more neighboring information is included; 4, 8 and 12 circles have very similar performances, implying that large number of circles on each ring is unnecessary, due to overlapping of adjacent regions. Therefore, 8o3r4c is a good choice of parameters for DAISY, and is applied in our experiments.

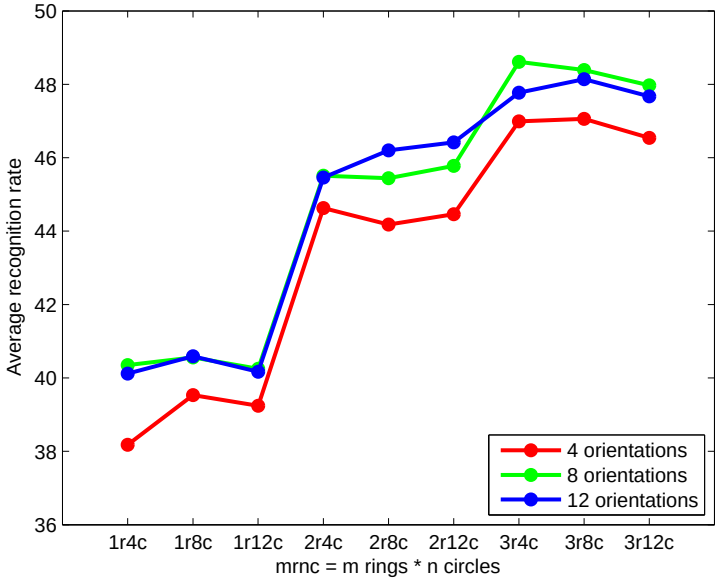



Figure 6.5: Performance comparison for different number of quantized orientations used in DAISY

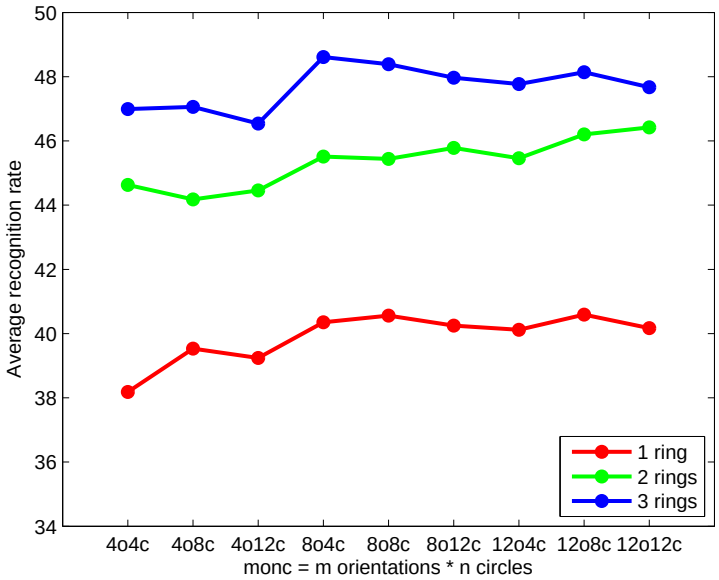



Figure 6.6: Performance comparison for different number of convolved orientation rings used in DAISY

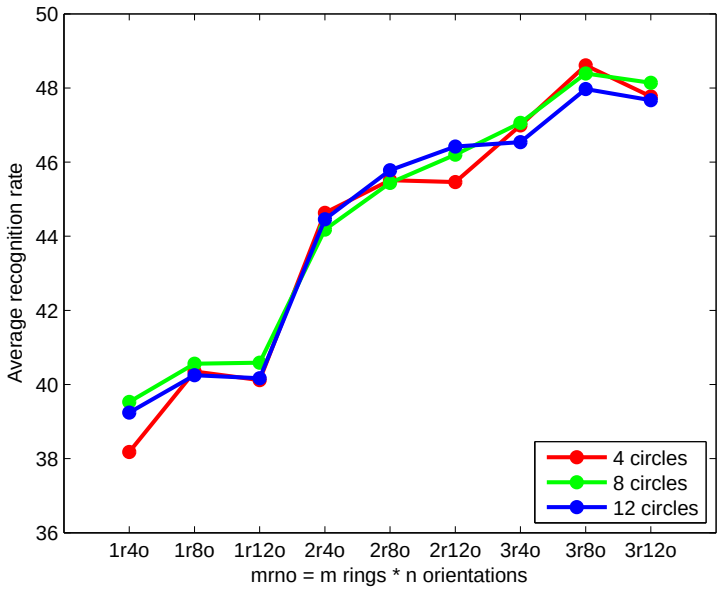



Figure 6.7: Performance comparison for different number of circles used on each ring in DAISY

### 6.4.5 Computational cost

In order to validate the computational efficiency of DAISY, we compare it with SIFT in Table 6.1. The comparisons are conducted on the Caltech 101 dataset with 30 training settings, and on an Intel Core 2 Duo CPU @ 3.16 GHz with 3GB RAM. The last column of the table means the average time required for descriptor extraction per image (about size of $300\times200$)[1]. It can be seen that the best DAISY (15R8o3r4c) is 3 times faster than SIFT, with more than 4% superiority on performance. Even a simpler DAISY (15R4o1r4c) can obtain comparable performance to SIFT, with only 1/6 descriptor length and 12 times faster computation.

## 6.5 Conclusions

In this chapter, we investigated DAISY, an efficient local descriptor, for the task of visual object recognition. We carefully evaluated its performances with different parameter settings on two standard image datasets, namely Caltech 101 and PAS-

[1]We use the MATLAB implementations available online for computing both descriptors. For DAISY, http://cvlab.epfl.ch/~tola/daisy.html. For SIFT, http://www.vlfeat.org/.

Table 6.1: Performance comparison of DAISY and SIFT

| Caltech 101 (30 train) | Recognition accuracy | Descriptor length | Computation time |
|---|---|---|---|
| DAISY (15R8o3r4c) | 48.61% | 104 | 0.218s |
| DAISY (15R4o2r8c) | 46.36% | 68 | 0.126s |
| DAISY (15R4o1r4c) | 44.17% | 20 | 0.054s |
| SIFT | 44.06% | 128 | 0.666s |

CAL VOC 2007, and compared it with the state-of-the-art SIFT descriptor. The experimental results showed that DAISY outperforms SIFT with a shorter descriptor length, and can operate 12 times faster than SIFT when displaying similar recognition accuracy. All these make DAISY a very competitive local descriptor for the task of visual object recognition.

Chapter 7

# Histograms of the Second Order Gradients (HSOG) for Object Recognition

Contents

## 7.1 Introduction

As we introduced in section 2.2.2.2, many local image descriptors [Lowe 2004] [Dalal & Triggs 2005] [Bay *et al.* 2008] [Tola *et al.* 2010] [Belongie *et al.* 2002] [Heikkilä *et al.* 2009] calculated based on interest regions have been proposed and proven competent compared with the global ones, and these local features are highly distinctive to identify specific objects, partially invariant to illumination variations, robust to occlusions, and insensitive to local image distortions.

Since long ago, it has been admitted that human visual processing could not be explained only by the first order mechanisms which capture the spatio-temporal variations in luminance, and the second order based ones capture complementary information such as difference of texture and spatial frequency [Smith & Scott-Samuel 2001]. Despite the great variety in design principle and implementation, the overwhelming majority of the existing local image descriptors share one common ground that they make use of the information of the first order gradients, e.g. locations, orientations and magnitudes. In contrast, quite limited efforts are made on the second order gradients. In [Brown *et al.* 2011], the authors proposed an unified framework for local descriptor design, and pointed out high order gradients (2nd and 4th) are helpful in the application of multi-view stereo matching. However, to the best of our knowledge, local image descriptors based on the second order gradients are seldom investigated in the literature for the purpose of object recognition. Intuitively, the second order gradient information should not only possess certain discriminative power to distinguish different object classes, but also tend to be complementary to the information provided by the first order gradients. This intuition could also be characterized by an analogy of object motion which requires not only the velocity but also the acceleration for a comprehensive description. According to this analogy, within a pre-defined distance between two pixels, the first order gradients imitate the velocity of the gray value variation, while the second order gradients simulate its corresponding acceleration. Therefore, in order to address the confusion caused by intra-class variations as well as inter-class similarities, and ameliorate the quality of visual content representation, both the

first and second order gradient information is necessary.

Therefore, in this chapter, we propose a novel and powerful local image descriptor, namely Histograms of the Second Order Gradients (HSOG), for object recognition. As its name implies, HSOG encodes the second order gradient information to represent local image variations. Specifically, for a certain image region, HSOG begins with computing its first order Oriented Gradient Maps (OGMs), each of which is for a quantized direction, and the histograms of the second order gradients are then extracted on the OGMs. The histograms of all OGMs are further concatenated, and after PCA-based dimensionality reduction, a compact local image representation is finally achieved. Additionally, we embed spatial information by introducing the multi-scale strategy to improve the categorization accuracy. The experiments are carried out on the Caltech 101 dataset [Li *et al.* 2007], and the results clearly demonstrate the effectiveness of the proposed HSOG descriptor and show that they are also complementary to the first order gradient based ones.

## 7.2 HSOG descriptor construction

In this section, we present the Histograms of the Second Order Gradient (HSOG) descriptor in detail. Its construction is composed of four main steps: (1) computation of the first order Oriented Gradient Maps (OGMs); (2) computation of the second order gradients based on these computed OGMs; (3) spatial pooling; and (4) dimensionality reduction. The entire process is illustrated in Figure 7.1.

### 7.2.1 Computation of the first order Oriented Gradient Maps (OGMs)

The input of the proposed HSOG descriptor is an image region around the keypoint, which is either detected by interest point detectors, e.g. Harris-Laplace, or located on a dense sampling grid. For each pixel $(x, y)$ within the given region $I$, a certain number of gradient maps $G_1$, $G_2$, ..., $G_N$, one for each quantized direction $o$, are

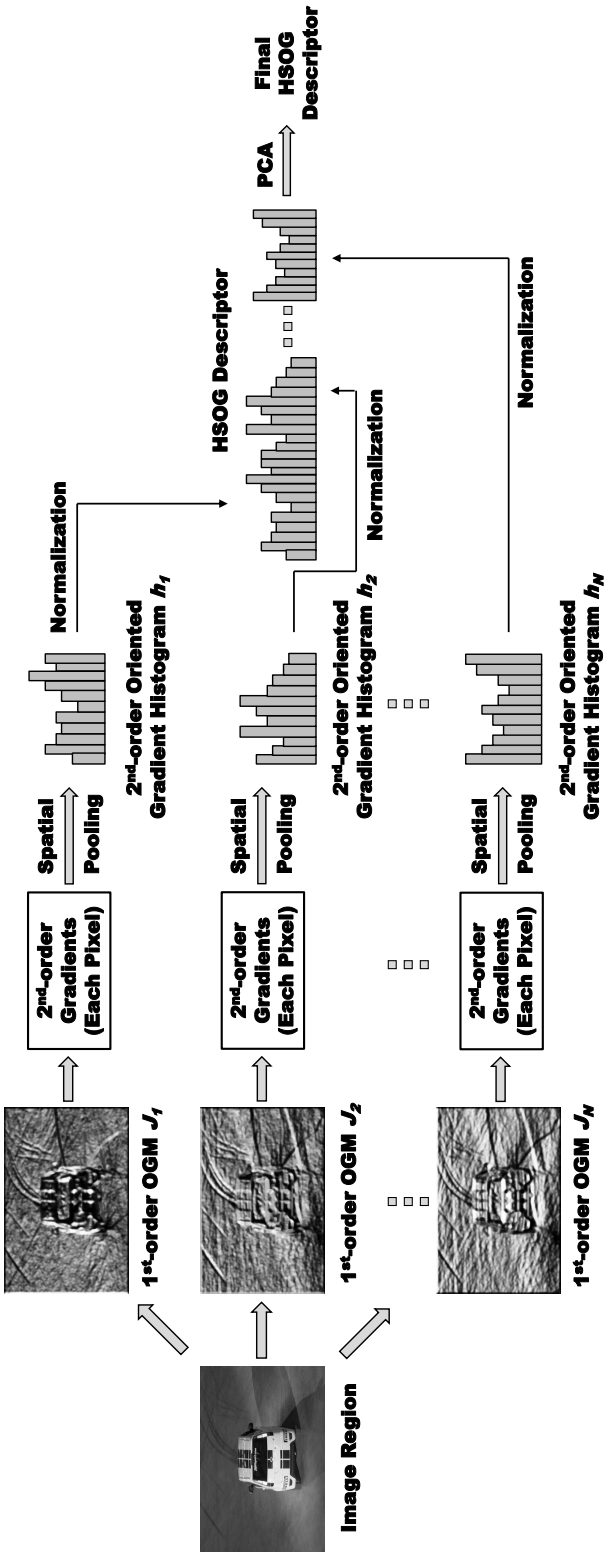



Figure 7.1: Construction process of the proposed HSOG descriptor

first computed. They are formally defined as:

$$G_o = \left(\frac{\partial I}{\partial o}\right)^+; \quad o = 1, 2, \ldots, N. \tag{7.1}$$

where the '+' sign means that only positive values are kept to preserve the polarity of the intensity changes, while the negative ones are set to zero.

Each gradient map describes gradient norms of the input image region in a direction $o$ at every pixel location. We then convolve its gradient maps with a Gaussian kernel $G$. The standard deviation of the Gaussian kernel $G$ is proportional to the radius of the given neighborhood, $R$, as equation (7.2):

$$\rho_o^R = G_R * G_o \tag{7.2}$$

The purpose of the convolution with Gaussian kernels is to allow the gradients to shift within a neighborhood without abrupt changes.

At a given pixel location $(x, y)$, we collect all the values of these convolved gradient maps at that location and build the vector $\rho^R(x, y)$ as:

$$\rho^R(x, y) = \left[\rho_1^R(x, y), \cdots, \rho_N^R(x, y)\right]^T \tag{7.3}$$

This vector, $\rho^R(x, y)$, is further normalized to unit norm vector, which is called in the subsequent entire orientation vector and denoted by $\underline{\rho^R}$, and the image region can be thus represented by entire orientation vectors. Specifically, given an image region $I$, we generate an Oriented Gradient Map (OGM) $J_o$ for each orientation $o$ defined as:

$$J_o(x, y) = \underline{\rho}_o^R(x, y) \tag{7.4}$$

Figure 7.2 illustrates such a process. Thanks to the computation of gradient maps as well as the following normalization step, OGMs possess the property of being invariant to affine lighting transformations, which can be inherited by the whole HSOG descriptor.

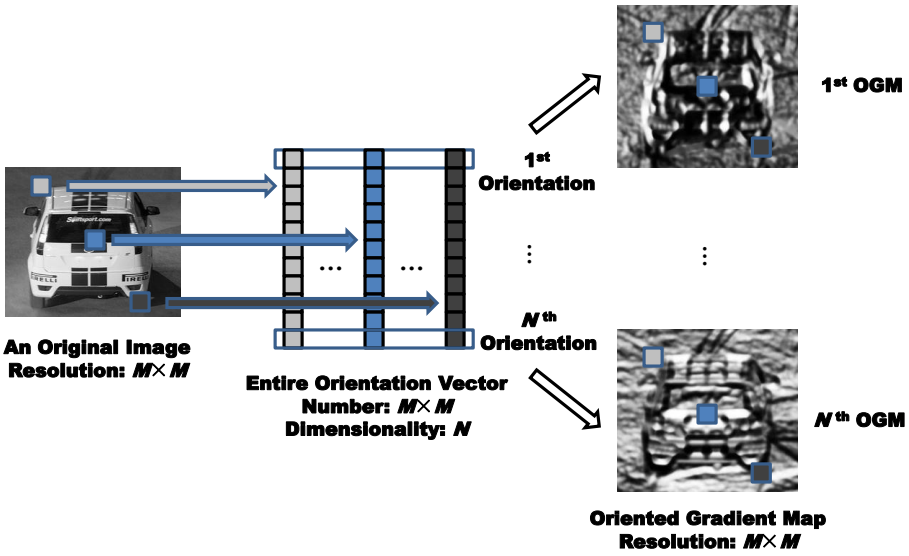



Figure 7.2: An illustration of the oriented gradient maps for each of the quantized orientations $o$

### 7.2.2 Computation of the second order gradients

Once the first order OGMs of all quantized directions are generated, they are employed as the input for computing the second order gradients in the same image region. Precisely, for each first order OGM, $J_o(x, y)$, $o = 1, 2, \ldots, N$, we consider it as a regular image, and calculate the gradient magnitude $mag_o$ and orientation $\theta_o$ at every pixel location as equation (7.5) and (7.6):

$$mag_o(x, y) = \sqrt{\left(\frac{\partial J_o(x, y)}{\partial x}\right)^2 + \left(\frac{\partial J_o(x, y)}{\partial y}\right)^2} \tag{7.5}$$

$$\theta_o(x, y) = \arctan\left(\frac{\partial J_o(x, y)}{\partial y} \Big/ \frac{\partial J_o(x, y)}{\partial x}\right) \tag{7.6}$$

where $o = 1, 2, \ldots, N$;

$$\frac{\partial J_o(x, y)}{\partial x} = J_o(x + 1, y) - J_o(x - 1, y) \tag{7.7}$$

$$\frac{\partial J_o(x, y)}{\partial y} = J_o(x, y + 1) - J_o(x, y - 1) \tag{7.8}$$

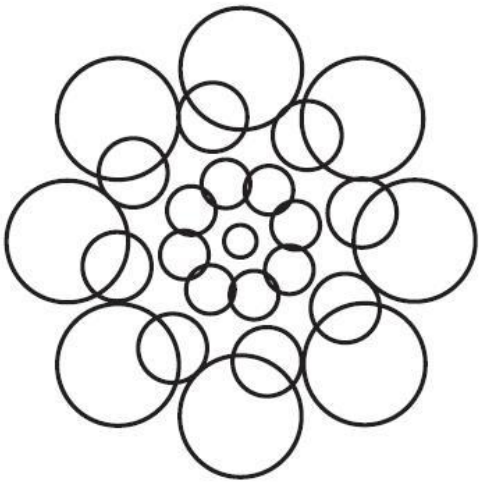


Figure 7.3: Spatial pooling arrangement (DAISY-style in [Brown *et al.* 2011]) of the proposed HSOG descriptor

Then, each orientation $\theta_o$ is mapped from $[-\pi/2, \pi/2]$ to $[0, 2\pi]$, and quantized into $N$ dominant orientations, which keeps consistent with the number of the first order OGMs. After quantization, the entry $n_o$ of each direction $\theta_o$ is calculated as equation (7.9):

$$n_o(x,y) = \mathrm{mod}\left(\left\lfloor \frac{\theta_o(x,y)}{2\pi/N} + \frac{1}{2} \right\rfloor, N\right), o = 1, 2, \ldots, N \tag{7.9}$$

### 7.2.3 Spatial pooling

Spatial pooling is an effective way for local descriptors to encode coarse spatial information of image pixels. It divides the input image region into sub-regions and accumulates a histogram of certain property (gradients, edge points, binary patterns, etc.) within each sub-region. All these histograms are then concatenated to construct the final descriptor. Brown et al. [Brown *et al.* 2011] analyzed different spatial pooling schemes and compared their performances, indicating that the best performance was achieved by the DAISY-style arrangement, as illustrated in Figure 7.3. Therefore, we follow this way for spatial pooling of the HSOG descriptor.

The input image region is divided into circles of different size located on a series of concentric rings. The radius of each circle is proportional to its distance from the central pixel. As a result, there are four parameters that determine the spatial

arrangement of the HSOG descriptor: the radius of the region area ($R$); the number of quantized orientations ($N$); the number of concentric rings ($CR$); the number of circles on each ring ($C$). The influence of different parameters will be analyzed experimentally in section 7.4.2.

The total number of the divided circles can be calculated as $T = CR \times C + 1$. Within each circle $CIR_j$, $j = 1, 2, \ldots, T$, and for each first order OGM $J_o$, $o = 1, 2, \ldots, N$, a second order oriented gradient histogram, $h_{oj}$, is built as equation (7.10) by accumulating the gradient magnitudes $mag_o$ of all the pixels with the same quantized orientation entry $n_o$.

$$h_{oj}(i) = \sum_{(x,y) \in CIR_j} f(n_o(x,y) == i) * mag_o(x,y) \tag{7.10}$$

where $i = 0, 1, \ldots, N-1$; $o = 1, 2, \ldots, N$, $j = 1, 2, \ldots, T$,

$$f(x) = \begin{cases} 1, \text{if } x \text{ is true} \\ 0, \text{otherwise} \end{cases} \tag{7.11}$$

Then, for each first order OGM $J_o$, its second order oriented gradient histogram $h_o$ is generated by concatenating all the histograms from $T$ circles as equation (7.12):

$$h_o = [h_{o1}, h_{o2}, h_{o3}, \cdots, h_{oT}]^{\mathrm{T}} \tag{7.12}$$

where $o = 1, 2, \ldots, N$. The HSOG descriptor is obtained by concatenating all $N$ histograms of the second order oriented gradient as equation (7.13). Each histogram $h_o$ is normalized to an unit norm vector $\hat{h}_o$ before the concatenation.

$$\mathrm{HSOG} = [\hat{h}_1, \hat{h}_2, \hat{h}_3, \cdots, \hat{h}_N]^{\mathrm{T}} \tag{7.13}$$

### 7.2.4 Dimensionality reduction

The dimension of the achieved HSOG descriptor is $T \times N^2$, which is relatively high (from hundreds up to more than one thousand) for the following steps. In order to reduce the dimensionality and increase the discriminative power, we further apply

the well known Principal Component Analysis (PCA) technique [Jolliffe 2002], since it has been successfully applied in the PCA-SIFT and GLOH cases for the same objective.

To build the eigenspace, we located 76,000 local image patches by applying the Harris-Laplace interest point detector [Mikolajczyk & Schmid 2004] on a diverse collection of images which is out of the dataset for validation. Each of these patches was adopted to compute its HSOG descriptor, and PCA was applied on the covariance matrix of these descriptors. The matrix consisting of the top $n$ eigenvectors was stored and utilized as the projection matrix.

For a certain local image region, its HSOG descriptor is firstly computed and then projected into a low-dimensional feature space by multiplying the pre-trained projection matrix. The dimension of the final HSOG descriptor is hence reduced to $n$. We experimentally determined the best values for $n$, and set $n = 128$ in the following experiments. The discussion about the choice of the value $n$ will be presented in section 7.4.3.

## 7.3 Attribute comparison with main local descriptors

As we presented in section 2.2.2.2, the attributes of the most popular local descriptors applied to the domain of object recognition are summarized in Table 2.3, including representation type (sparse or dense), encoded information, spatial pooling scheme (neighborhood grid), computation method (comp.), and dimensionality (dim.). The comparisons can now be updated as in Table 7.1 after we introduced the DAISY descriptor in chapter 6 and proposed the HSOG descriptor in this chapter.

## 7.4 Experimental evaluation

We evaluate the proposed HSOG descriptor in the context of visual object recognition on the standard Caltech 101 dataset [Li *et al.* 2007]. Its detailed introduction can be found in section 3.2.

Table 7.1: Attribute summary of main local image descriptors applied to object recognition

| Descriptor | Type | Information | Grid | Comp. | Dim. |
|---|---|---|---|---|---|
| SIFT | Sparse | Gradient (1st) | Rect. | Distr. | 128 |
| PCA-SIFT | Sparse | Gradient (1st) | Rect. | Distr. | 36 |
| Color SIFT | Sparse | Gradient (1st) | Rect. | Distr. | 384 |
| GLOH | Sparse | Gradient (1st) | Polar | Distr. | 128 |
| HOG | Dense | Gradient (1st) | Rect. & Polar | Distr. | 36 |
| SURF | Sparse | Wavelet response | Rect. | Filter | 64 |
| Shape Context | Sparse | Edge points | Polar | Distr. | 60 |
| CS-LBP | Sparse | Binary patterns | Rect. | Distr. | 256 |
| DAISY | Dense | Gradient (1st) | Polar | Filter | 200 |
| **HSOG** | Sparse | **Gradient(2nd)** | Polar | Distr. | 128 |

### 7.4.1 Experimental setup

We follow the same approach as introduced in section 5.5.3 for object recognition. The block diagram of the approach is depicted in Figure 5.7.

For each image in the dataset, the Harris-Laplace detector is firstly applied to detect interest points, and a local region around each interest point is then selected to extract the HSOG descriptor. For the purpose of comparison, several state-of-the-art descriptors are also extracted from these regions, including SIFT [Lowe 2004], DAISY [Tola *et al.* 2010] and CS-LBP [Heikkilä *et al.* 2009]. Specifically, we implement the CS-LBP descriptor according to [Heikkilä *et al.* 2009], and use the source codes available online [1] for computing SIFT and DAISY.

We apply the popular Bag-of-Features (BoF) modelling method [Csurka *et al.* 2004] introduced in section 2.2.2.3 due to its great success in object recognition tasks. In our case, a vocabulary of 4000 "visual words" is constructed for each kind of local descriptors respectively by applying the k-means clustering algorithm on a subset of the descriptors randomly selected from the training data as in [van de Sande *et al.* 2010].

The Support Vector Machine (SVM) algorithm introduced in section 2.3.2.1 is applied for classification. When all the local descriptors are transformed to fixed-

[1] Code for SIFT: `http://www.vlfeat.org/`
Code for DAISY: `http://cvlab.epfl.ch/~tola/daisy.html/`

length feature vectors by the BoF method, the $\chi^2$ distance is computed as equation (2.36) to measure the similarity between each pair of the feature vectors. Then, the kernel function based on this distance is utilized as equation (2.37) for the SVM training and prediction. Finally, each test image is classified into object class with the maximum SVM output decision value. We tune the parameters of the classifier on the training set via 5-fold cross-validation, and obtain the recognition accuracy on the test set.

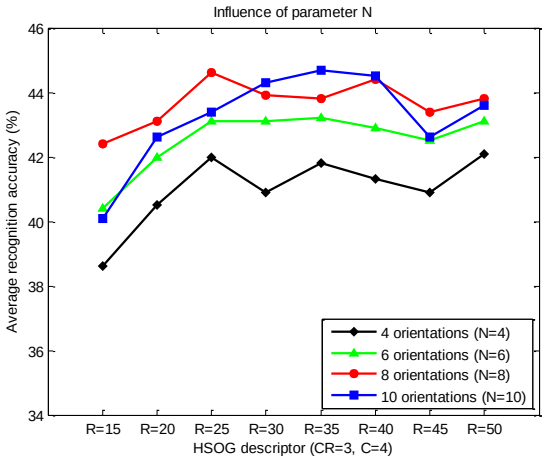
To carry out the experiments on the Caltech 101 dataset, we follow the common training and test settings as used in [Varma & Ray 2007] [Zhang *et al.* 2006]. For each object category, 30 images are randomly selected, while 15 are for training and the other 15 for test, resulting in totally 1530 images for training and 1530 images for test respectively. The experiments are repeated three times with different training and test selections, and the average recognition accuracy is reported.

### 7.4.2 Parameter selection

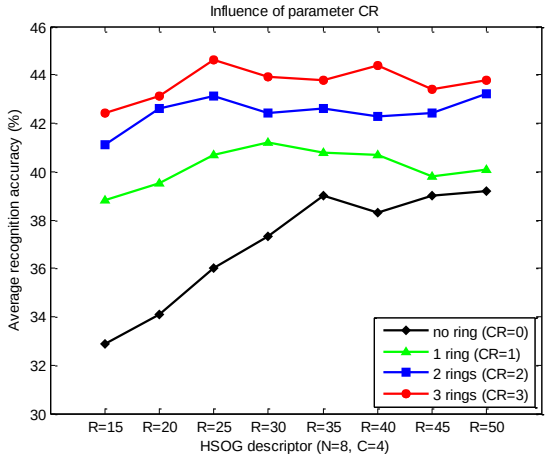
Recall that the HSOG descriptor has four parameters: the radius of the region area ($R$); the number of quantized orientations ($N$); the number of concentric rings ($CR$); as well as the number of circles on each ring ($C$). To evaluate their impacts on the performance of the descriptor, we draw a series of line graphs of the recognition accuracy on different $R$ by alternately changing one parameter while fixing the others for $N$, $CR$ and $C$. The results are shown in Figure 7.4.

It can be observed from Figure 7.4 (a) that the descriptors with 8 orientations perform clearly better than that with 4 and 6; while the one with 10 orientations shows no superiority to that with 8, indicating that 8 orientations are sufficient to describe local image variations. From Figure 7.4 (b), we can see that the performance keeps improving when the number of concentric rings increases, showing that the descriptor based on more rings is better, because more neighboring information is included. Figure 7.4 (c) shows that raising the number of the circles on each ring does not improve the performance, implying that large number of circles on each ring is unnecessary, due to overlapping of adjacent regions.

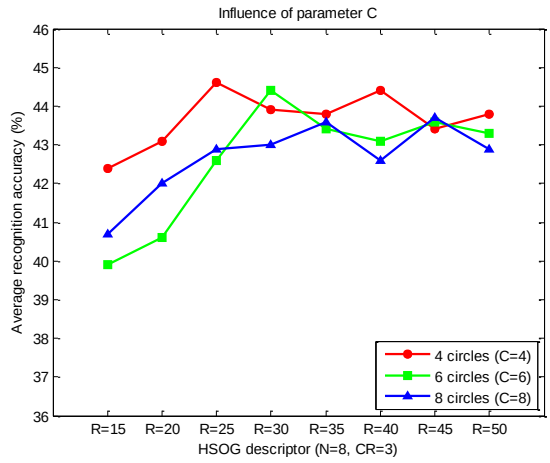
Another phenomenon from these three figures is that the performance rises con-




(a)




(b)




(c)

Figure 7.4: Influence of different parameters in HSOG. (a) the number of quantized orientations $N$; (b) the number of concentric rings $CR$; (c) the number of circles on each ring $C$.

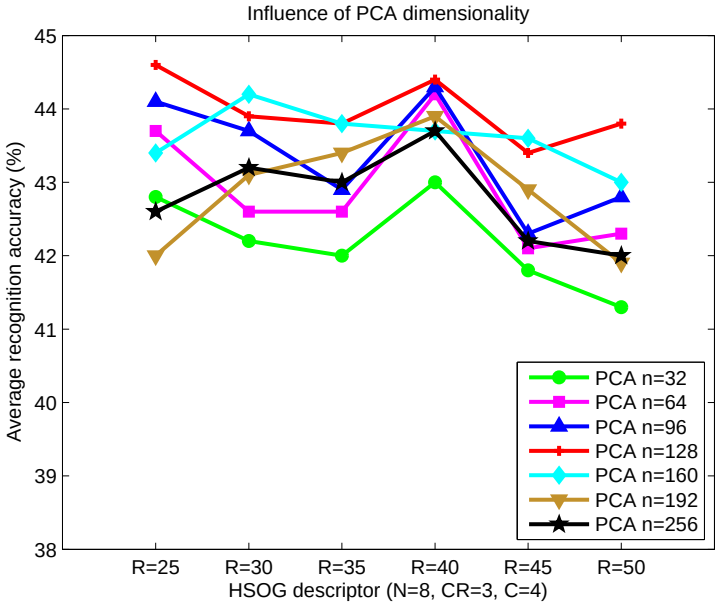



Figure 7.5: Influence of the PCA-based dimensionality reduction for the proposed HSOG descriptor

tinuously with the size of region area $R$ when it is small. After $R$ reaches a certain point (about 25 pixels), the performance improvement is not obvious if $R$ continues increasing. Therefore, we choose the best parameter setting for the proposed HSOG descriptor as follows: $R = 25$, $N = 8$, $CR = 3$, $C = 4$.

### 7.4.3 Influence of PCA-based dimensionality reduction

We also discussed the impact of the PCA-based dimensionality reduction on the HSOG performance. A series of curves of the recognition accuracy based on different region sizes are generated by varying the dimensionality $n$ calculated by PCA from 32 to 256, as shown in Figure 7.5.

We calculated the values of means and deviations of the descriptors with individual fixed dimensions, and found that the performance of the 128-dimensional descriptor ($44.00 \pm 0.44$) was better than those of the others, such as 32 dimension ($42.18 \pm 0.63$); 64 dimension ($42.92 \pm 0.84$); 96 dimension ($43.35 \pm 0.80$); 160 dimension ($43.62 \pm 0.40$); 192 dimension ($42.87 \pm 0.79$); and 256 dimension ($42.78 \pm 0.64$). Therefore, 128 is chosen as the dimensionality of the HSOG descriptor.

### 7.4.4 Multi-scale extension

In order to compute the HSOG descriptor, a local image region around keypoints should be fixed. The optimal size of this region is often selected based on the scale of the keypoint given by detectors or chosen manually. In section 7.4.2, we experimentally evaluated the impacts of different region sizes, and selected a good one. However, a single size of region is probably not enough to characterize the neighborhood of a keypoint. More spatial information could be embedded if the regions with multiple sizes are considered. Therefore, we adopt the multi-scale strategy to further improve the discriminative power of the HSOG descriptor.

We make use of the multiple kernel learning (MKL) algorithm [Rakotomamonjy *et al.* 2008] (see section 2.3.2.2 for a detailed introduction) to combine different HSOG descriptors from multi-scale regions, since this strategy does not increase the dimensionality of the features, and the similarity scores based on different parameters can be calculated individually, leading to a realistic implementation of parallel computing, e.g. GPU programming, without increasing the time cost. Specifically, for each keypoint $p$, we choose a certain number of concentric regions around $p$ with increasing sizes. The HSOG descriptor is then extracted from each region and applied for object recognition independently by following the approach described in section 7.4.1. The kernel matrices of different descriptors are combined using MKL to achieve the final recognition results.

From the experimental results shown in Table 7.2, we can see that the performance of the HSOG descriptor is significantly improved from 44.64% (the best single scale region) to 52.55% (4-region fusion) and 54.25% (8-region fusion). This nearly 10% improvement clearly proves the effectiveness of the multi-scale fusion. Furthermore, 8-region fusion performs better than 4-region fusion, indicating that the performance could benefit from more regions.

### 7.4.5 Performance evaluation and comparison

We evaluate the proposed HSOG descriptor with the best parameter setting on the Caltech 101 dataset. As introduced in section 7.4.1, we compare it with other state-

Table 7.2: Performance comparison of the HSOG descriptors (multi-scale regions vs. single scale regions) on the Caltech 101 dataset

| Type | Recognition Accuracy (%) | | | |
|---|---|---|---|---|
| Single-scale | $R = 15$ | 42.35 | $R = 20$ | 43.07 |
| | $R = 25$ | 44.64 | $R = 30$ | 43.92 |
| | $R = 35$ | 43.79 | $R = 40$ | 44.44 |
| | $R = 45$ | 43.40 | $R = 50$ | 43.79 |
| Multi-scale | $R = 25$ to 40 | | 52.55 | |
| | $R = 15$ to 50 | | 54.25 | |

of-the-art descriptors including SIFT, DAISY and CS-LBP as well. The parameter setting of HSOG is $N = 8$; $CR = 3$; $C = 4$, with the dimensionality of 128. SIFT uses the standard configuration as in [Lowe 2004], thus with 128-dimension. DAISY applies the same parameter setting as HSOG, and its dimension is 104. The parameters of CS-LBP are set according to [Heikkilä *et al.* 2009], i.e. the $4 \times 4$ grid with CS-LBP$_{2,8,0.01}$, resulting in a 256-dimensional descriptor.

We can see from Table 7.3 that the single-scale HSOG outperforms the first order gradient based descriptors, i.e. CS-LBP, DAISY and SIFT, and the categorization result achieved by multi-scale HSOG which combines the ones of four different regions is significantly increased by over 10%, clearly demonstrating the effectiveness of the HSOG descriptor. On the other hand, the fusion of the single scale (Ss) HSOG or multi-scale (Ms) HSOG with SIFT, DAISY or CS-LBP improves the categorization accuracy again, indicating that HSOG provides complementary information to that given by the existing local image descriptors, and their joint use is a promising way for visual content representation.

Also, we calculated the average computation time required for each input image (about size of $300 \times 250$) of these local descriptors using an Intel Core 2 Duo CPU @ 3.16 GHz with 3GB RAM, and it can be seen that the current version of HSOG is 3 times slower than SIFT. Nevertheless, it should be noted that because each first order OGM and its corresponding second order gradients can be computed individually, the current implementation of HSOG can be accelerated by GPU programming as we mentioned in section 7.4.4, which makes HSOG run approximately $N$ times faster ($N$ is the number of OGMs, e.g. 8 in our case), leading to a consumed time

Table 7.3: Performance and consumed time comparison between the HSOG descriptor and other state-of-the-art descriptors on the Caltech 101 dataset

| Descriptor | Recognition Accuracy (%) | Time (s) |
|---|---|---|
| **SIFT** | 40.92 | 0.316 |
| **DAISY** | 42.48 | 0.108 |
| **CS-LBP** | 35.62 | 0.087 |
| **HSOG (Ss)** | **44.64** | 0.985 |
| **HSOG (Ms)** | **52.55** | – |
| **HSOG (Ss) + SIFT** | **52.81** | – |
| **HSOG (Ss) + DAISY** | 51.70 | – |
| **HSOG (Ss) + CS-LBP** | 50.92 | – |
| **HSOG (Ms) + SIFT** | **56.27** | – |
| **HSOG (Ms) + DAISY** | 54.58 | – |
| **HSOG (Ms) + CS-LBP** | 54.64 | – |

comparable to the existing descriptors.

## 7.5 Conclusions

In this chapter, we presented a novel local image descriptor for object recognition, making use of histograms of the second order gradients, denoted as HSOG. The proposed HSOG descriptor intends to capture the acceleration information on pixel gray value changes, while the existing descriptors in the literature, such as SIFT, HOG, DAISY, etc., are based on the first order gradient information. The recognition results achieved on the Caltech 101 dataset clearly demonstrate that the proposed HSOG descriptor possesses a good discriminative power to distinguish different object categories, especially embedded with more spatial information provided by the multi-scale strategy. Furthermore, the information given by HSOG proves complementary to that based on the existing ones which exploit the first order gradient information.

Chapter 8

# Conclusions and Future Work

Contents



## 8.1 Conclusions

In this thesis, we focus on the problem of machine-based visual object recognition, which is a very active and important research topic during recent years, and still remains one of the most challenging problems in computer vision community. We follow the popular feature & classifier based approaches. As the very first step, visual content description is considered as one of the key issues for this problem. A good visual descriptor, which is both discriminative and computationally efficient while possessing some invariance properties against changes in viewpoint, scale and illumination, could greatly improve the classification performance. In such context, we propose, in this thesis, some innovative contributions to the task of visual object recognition, in particular by presenting several new visual features / descriptors to effectively and efficiently represent the visual content of images. Our contributions are summarized as follows.

Our first contribution is presented in chapter 4. We propose six multi-scale color local binary pattern (LBP) features to incorporate color information into the original LBP operator, which is a computationally efficient yet powerful texture feature that has been successfully applied to many applications as diverse as texture classification, texture segmentation, face recognition and facial expression recognition.

However, it has two main shortcomings. On one hand, the original LBP ignores all color information because its calculation is based on gray images, while color is an important clue for distinguishing objects, especially in natural scenes. On the other hand, the original LBP is only invariant to gray-level monotonic illumination changes, and thus is deficient in power to deal with various lighting condition changes in real-world scenes, which further complicate the recognition task. Therefore, the aim of the proposed features is to incorporate color information, as well as to enhance the discriminative power and the photometric invariance property of the original LBP. In addition, in order to encode spatial information of texture structures, a coarse-to-fine image division strategy is applied for calculating the proposed features within image blocks, and the performances are further improved. The experimental results on the PASCAL VOC 2007 benchmark prove that the proposed features can gain significant improvement on recognition accuracy, and thus are promising for real-world object recognition tasks.

Our second contribution lies in a new type of local image descriptor based on LBP. In chapter 5, we propose several new local descriptors based on the orthogonal combination of local binary patterns (denoted as OC-LBP) to deal with the downside of the state-of-the-art descriptors such as SIFT and its extensions or refinements, their relatively high computational cost. With the trend of significant increase of the dataset scale, it is highly desirable that local descriptors offer both high discriminative power and computational efficiency. The LBP operator is a good candidate to be used to construct a local descriptor, because of its computational simplicity and strong descriptive power for texture structures. However, the barrier lies in the high dimensional feature vectors that it produces, especially when the number of considered neighboring pixels increases. Therefore, we first propose a new dimensionality reduction method for LBP, namely the orthogonal combination of local binary patterns (the OC-LBP operator). It proves much more effective than other popular methods such as "uniform patterns" and CS-LBP operator by the experiments on a standard texture classification dataset. Then, we adopt the OC-LBP operator to construct a distribution-based local descriptor, denoted as the OC-LBP descriptor, by following a way similar to SIFT. Our aim is to

build a more efficient local descriptor by replacing the costly gradient information with local texture patterns in the SIFT scheme. Moreover, as the extension of our first contribution, we also propose six color OC-LBP descriptors by extending the intensity-based OC-LBP descriptor to different color spaces in order to enhance its discriminative power and photometric invariance property. The experimental results in three different applications — image matching, object recognition and scene classification — show the effectiveness of the proposed descriptors. They outperform the popular SIFT and CS-LBP descriptors, and achieve comparable or even better performances than the state-of-the-art color SIFT descriptors. Meanwhile, they provide complementary information to SIFT, since further improvement can be obtained by fusing these two kinds of descriptors. Moreover, the proposed gray and color OC-LBP descriptors are about 4 times faster to compute than the SIFT and color SIFT descriptors respectively. Therefore, they are very promising for large scale recognition problems.

Our third contribution is presented in chapter 6. We introduce the DAISY descriptor for the task of visual object recognition. There is now a trend in computer vision community that the scale of the benchmark datasets used for object recognition / image classification becomes larger year by year. However, it is well known that the gradient-distribution-based local descriptors such as SIFT, GLOH and HOG obtain the state-of-the-art performances, while the main drawback of them is their relatively high computational cost. Thus, more computationally efficient local descriptors are urgently demanded to deal with large scale datasets such as ImageNet and TRECVID. Usually, there are two ways to do this: one is to replace the costly gradient information with other more efficient features, just as what we did in the case of the OC-LBP descriptor; the other is to find more efficient methods to calculate the gradient information. The DAISY descriptor, which was initially designed for wide-baseline stereo matching problem, is a newly introduced fast local descriptor based on gradient distribution, and has shown good robustness against many photometric and geometric transformations. It has never been used in the task of visual object recognition, while we believe that it is very suitable for this problem. Therefore, we investigate the DAISY descriptor in the context of visual

object recognition by evaluating and comparing it with the popular SIFT both in terms of recognition accuracy and computation complexity on two standard image benchmarks. The experimental results on Caltech 101 and PASCAL VOC 2007 show that DAISY outperforms SIFT with a shorter descriptor length, and can operate 12 times faster than SIFT when displaying similar recognition accuracies. DAISY thus provides a fast and more efficient way to calculate the gradient information for the task of visual object recognition.

Our fourth contribution is presented in chapter 7. We propose a novel local image descriptor called histograms of the second order gradients (denoted as HSOG) for visual object recognition. In the literature, the most effective feature for characterizing an object's appearance or the content of an image is the first order gradient information, based on which many successful and state-of-the-art descriptors, such as SIFT, GLOH, HOG and DAISY, are constructed. Intuitively, the second order gradient information, which, to the best of our knowledge, is seldom investigated in the literature for object recognition, should not only possess certain discriminative power to distinguish different objects, but also tends to be complementary to the description provided by the first order gradients. Indeed, since long ago, it has been admitted that human visual processing could not be explained only by the first order mechanisms which capture the spatio-temporal variations in luminance. The second order mechanisms could capture complementary information such as difference of texture and spatial frequency. This intuition could also be characterized by an analogy of object motion which requires not only the velocity but also the acceleration for a comprehensive description. According to this analogy, within a pre-defined distance between two pixels, the first order gradient imitates the velocity of the gray value variation, while the second order gradient simulates its corresponding acceleration. In order to ameliorate the quality of visual content representation, both the first and second order gradient information is necessary. The experimental results achieved on the Caltech 101 dataset show that the proposed HSOG descriptor outperforms the first order gradient based descriptors, e.g. SIFT, CS-LBP and DAISY, by more than 10%, indicating that HSOG possesses a good discriminative power to distinguish different object categories, especially embedded

with more spatial information provided by the multi-scale strategy. Furthermore, the fusion of HSOG with SIFT, CS-LBP or DAISY improves the recognition accuracy again, demonstrating the complementarity of information provided by both the first and second order gradient based descriptors.

## 8.2 Perspectives for future work

We present in this section some perspectives for future research directions.

For the OC-LBP descriptor, we now use 4-orthogonal-neighbor as the basic unit to divide the neighboring pixels of the original LBP operator into non-overlapping groups. Other types of the basic unit could also be considered. For example, we could use the basic unit of 3-equilateral-triangular-neighbor, which would further reduce the dimensionality of the original LBP. Therefore, the performance of the descriptor using different basic units remains to be evaluated through comprehensive experiments in future.

For the HSOG descriptor, other ways for gradient computation could also be adopted. According to [Dalal & Triggs 2005], the descriptor performance is sensitive to the way in which gradients are computed. Therefore, future work could be done by evaluating the performance of the HSOG descriptor with different ways to compute gradients, such as uncentred 1D mask $[-1, 1]$, cubic-corrected 1D mask $[1, -8, 0, 8, -1]$, $3 \times 3$ Sobel masks, and $2 \times 2$ diagonal masks $\left[\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} -1 & 0 \\ 0 & 1 \end{smallmatrix}\right]$, as in [Dalal & Triggs 2005]. Moreover, since the first and second order gradients are computed separately in the HSOG construction, they could adopt different ways for computation as well. In addition, the performance of the HSOG descriptor may be improved by applying the linear discriminant analysis (LDA), which is a discriminative technique, to replace the principal component analysis (PCA), which is a non-discriminative technique, for its dimension reduction.

The DAISY and the HSOG descriptors could also be incorporated with color information to enhance their discriminative power and photometric invariance properties, as what we did in the cases of LBP and OC-LBP.

For the extraction of the proposed features / descriptors, different parts of an

image are now equally treated. In other words, the features extracted from the different parts of an image are considered to have equal importance, regardless of their locations in the image. However, intuitively, they should have different importance. For example, the features extracted from the object area should have greater importance than those from the background area, especially in the case of datasets with big background clutter. This point has been confirmed in [Zhang *et al.* 2007]. Therefore, future work could be done to first locate the interest areas (usually the objects) in images by some detection or segmentation techniques, and then assign different weights to the features during extraction according to their locations in images.

For the classification, we now apply the standard SVM algorithm, which considers each training sample equally while training the classifier. However, due to intra-class variations and inter-class correlations, it is difficult for SVM to deal with the complexity of data distribution when the samples within the same category exhibit diversities and the samples from different categories display similarities in terms of visual attributes. Therefore, future work could be done to introduce different weights for different samples during the SVM training process. How to decide the values of weights for different samples also remains a problem, while [Malisiewicz & Efros 2008], [Lin *et al.* 2007] and [Yang *et al.* 2009b] provide some ideas.

Appendix A

# Participation in the Popular Challenges

We present here a brief introduction of our participation, during this thesis, in two popular challenges in computer vision community: the PASCAL VOC challenge [1] in image domain and the TRECVID challenge [2] in video domain, partly based on the work of this thesis.

## A.1 Participation in the PASCAL VOC challenge

The PASCAL Visual Object Classes (VOC) challenge is a popular benchmark for visual object recognition and detection in image domain. A detailed introduction of the PASCAL VOC can be found in section 3.1.

We participated in this challenge in 2009, 2010 and 2011 for the classification task. Its aim is to predict, for each test image, the presence or the absence of each of the twenty predefined classes.

In 2009, we participated in this challenge for the first time. The dataset includes 3473 images for training, 3581 images for validation, and 6650 images for test. As our baseline recognition system, we extracted from each image the dense SIFT descriptor and a set of global features, including Color Histogram, Color Moments, Color Coherence Vectors, Gray Level Co-occurrence Matrix, Local Binary Patterns, Edge Histogram, and Line Segment (see chapter 2 for their detailed introduction), to describe the visual content of images. A vocabulary of 4000 visual words was

[1] http://pascallin.ecs.soton.ac.uk/challenges/VOC/
[2] http://trecvid.nist.gov/

created for the Bag-of-Features model of SIFT, and hard assignment was adapted to build the histogram. The SVM classifier was used for classification, and the Chi-square distance was computed as the kernel of SVM for all kinds of features. The predicted probabilities of different features were fused according to their EER (Equal Error Rate) to decide the final classification results. For each object class, we trained the classifier on the "train" set, and tuned the parameters on the "val" set.

As a result, we achieved MAP (Mean Average Precision) of 45.0%, and ranked 13/20 by teams and 30/48 by submissions. The results by teams from the organizers are shown in Figure A.1.

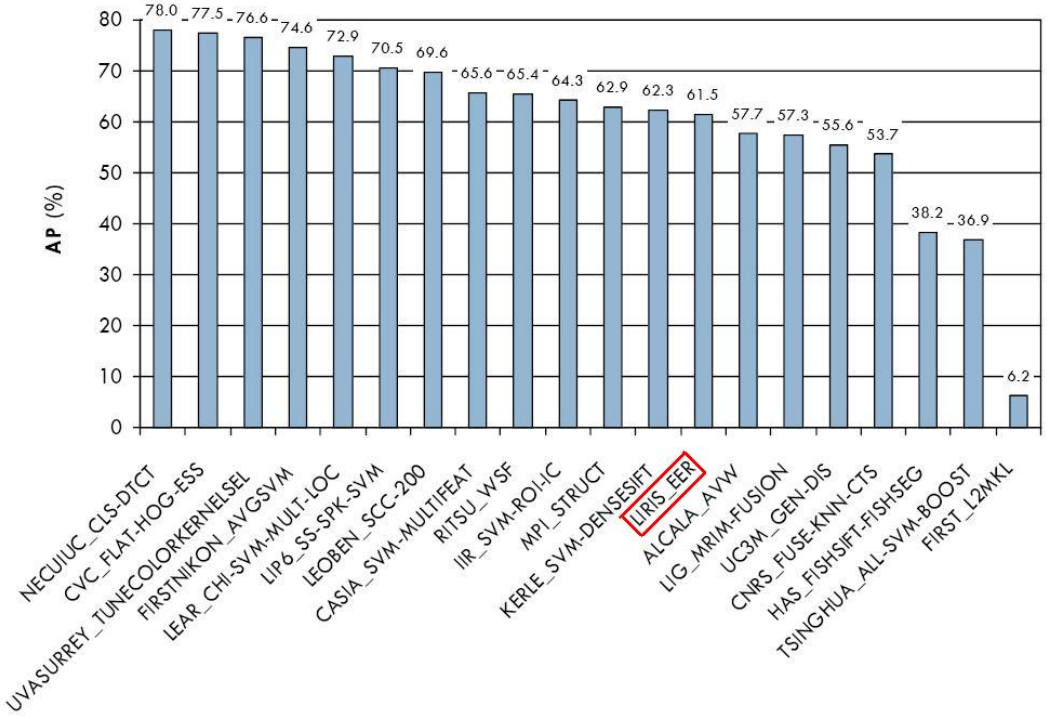



Figure A.1: PASCAL VOC challenge 2009 results by teams from the organizers

In 2010, the dataset was enlarged to include 4998 images for training, 5105 images for validation, and 9637 images for test. To improve the performance of our recognition system, we added our color LBP features presented in chapter 4, and considered more local descriptors including HOG and color SIFT (dense sampling + interest points). A vocabulary of 4000 visual words was created for the Bag-

of-Features model of each kind of local descriptors. Spatial pyramid information was also taken into account. The MKL (Multiple Kernel Learning) algorithm was applied to combine different features and perform the classification. The Chi-square distance was computed as the kernel for MKL. For each object class, we trained the classifier on the "train + val" set, and tuned the parameters via cross-validation.

As a result, we achieved MAP (Mean Average Precision) of 60.0%, and ranked 9/22 by teams and 15/32 by submissions, which was a great improvement compared to the year of 2009. The results by submissions from the organizers are shown in Figure A.2.

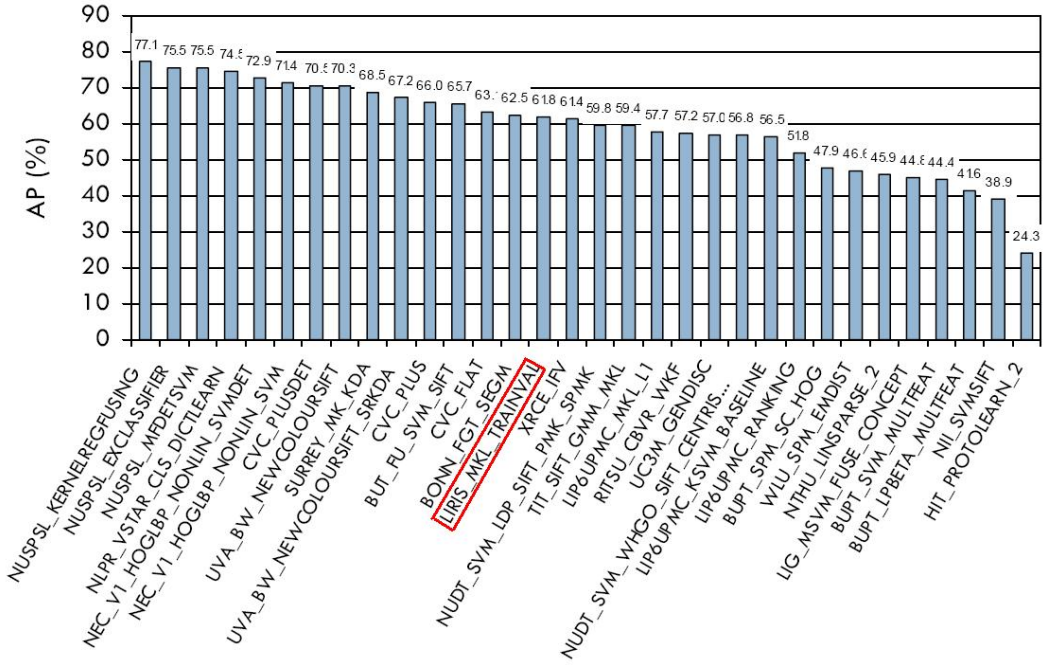



Figure A.2: PASCAL VOC challenge 2010 results by submissions from the organizers

In 2011, the dataset was enlarged again to include 5717 images for training, 5823 images for validation, and 10994 images for test. We made two submissions this year. For the submission LIRIS_CLS, we followed the same approach applied in 2010, but added two new kinds of features to further improve the recognition performance: color OC-LBP descriptors presented in chapter 5, and the DAISY descriptor presented in chapter 6. For the submission LIRIS_CLSDET, we improved the performance of the submission LIRIS_CLS by combining it with object

detection results. For object detection, we applied the HOG feature to train deformable part models [Felzenszwalb *et al.* 2010], and used the models together with sliding window approach to detect objects. Finally, we combined the outputs of classification and detection by late fusion.

As a result, our best submission (LIRIS_CLSDET) achieved MAP (Mean Average Precision) of 66.8%, and ranked 5/13 by teams and 7/20 by submissions, which was another improvement compared to the year of 2010. The results by submissions from the organizers are shown in Figure A.3.

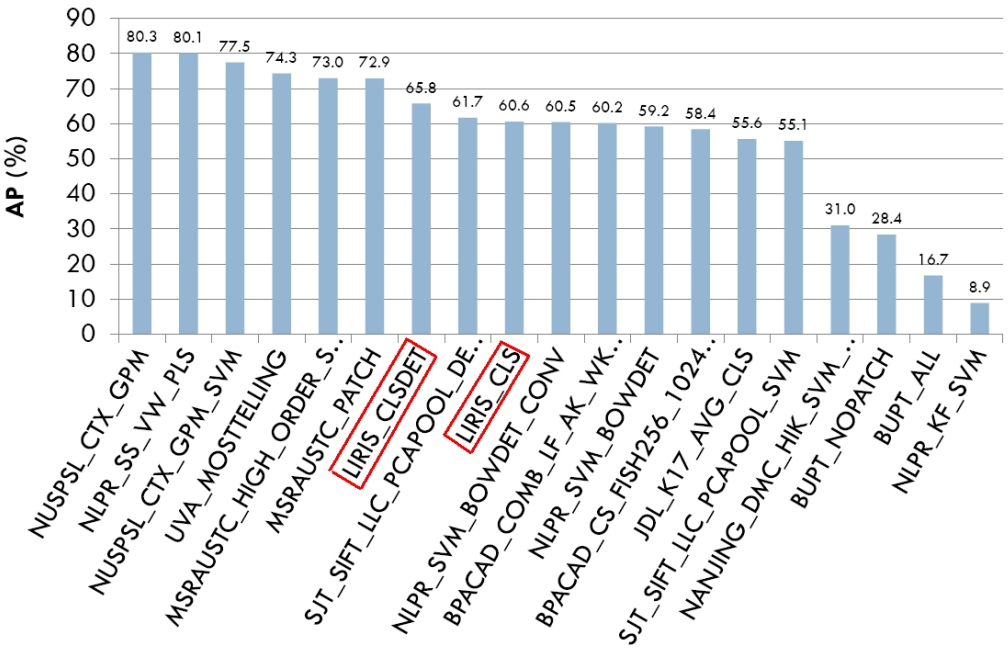



Figure A.3: PASCAL VOC challenge 2011 results by submissions from the organizers

## A.2 Participation in the TRECVID challenge

The TREC Video Retrieval Evaluation (TRECVID) challenge is a popular benchmark in video domain for content-based video analysis and retrieval. A detailed introduction of the TRECVID can be found in section 3.7.

We participated in the TRECVID challenge in 2011 for the first time, and focus on the semantic indexing task. Its aim is to automatically analyze the meaning con-

veyed by videos and tag video segments (shots) with semantic concept labels. More precisely, given the test collection, master shot reference, and concept definitions, participants are required to return for each concept a list of at most 2000 shot IDs from the test collection ranked according to the possibility of detecting the concept. In 2011, there are totally 346 concepts. The test set includes 200-hour video data with durations between 10 seconds and 3.5 minutes, while the development set contains 400-hour video data with durations just longer than 3.5 minutes. There are two types of runs for participants:

- Full run: including results for all 346 concepts
- Lite run: including results for 50 concepts, which is a subset of all 346 concepts selected by the organizers

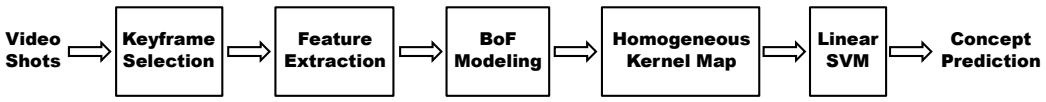



Figure A.4: Flowchart of our approach for participating in the semantic indexing task of the TRECVID challenge 2011

The flowchart of our approach is shown in Figure A.4. For keyframe selection, we decoded video data and kept single keyframe for each video shot. For feature extraction, we chose 4 visual features, including dense SIFT, color SIFT, OC-LBP and DAISY, together with 1 audio feature consisting of MFCC with delta and acceleration. Then we applied the Bag-of-Features method to transform all the visual descriptors into the fixed-length histograms to represent the visual content of the keyframes. For classification, since the popular non-linear SVM classifier is impractical for this problem due to the huge scale of video data, we adopted the solution of using non-linear kernel mapping together with fast linear SVM classifier. We applied the Homogeneous Kernel Map method proposed by Vedaldi and Zisserman [Vedaldi & Zisserman 2012] for non-linear kernel mapping. Its basic idea is to transform the data into a compact linear representation which reproduces the desired non-linear kernel to a very good level of approximation. Finally, we adopt-

ed a late-fusion strategy which directly averages the output probabilities of all the classifiers.

The results are presented in Figure A.5 and A.6. Our best submission (visual + audio) achieved the rank of 45/102 for lite run and 37/68 for full run. Considering that this is our first time to participate in this challenge, and we only used basic features and single keyframe representation due to the limited time, further improvement could be made by applying more powerful features and using multi-frame representation in the future work.

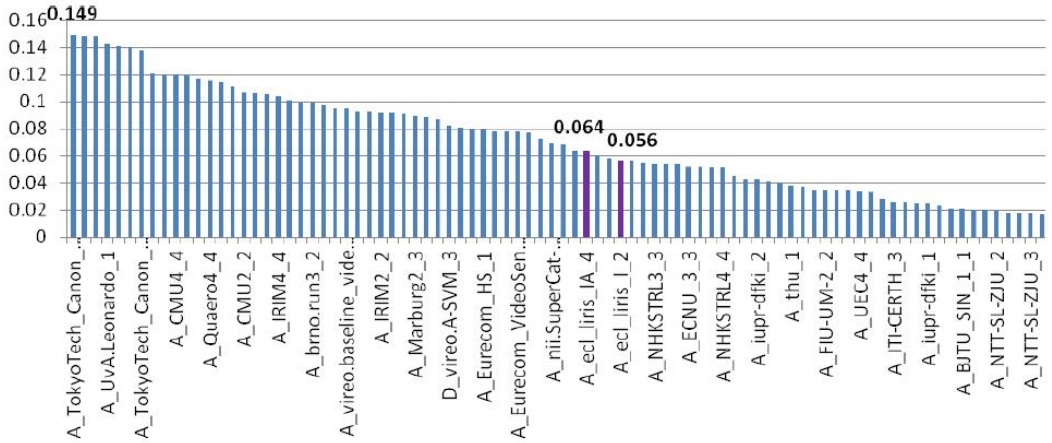



Figure A.5: Lite run results of TRECVID challenge 2011

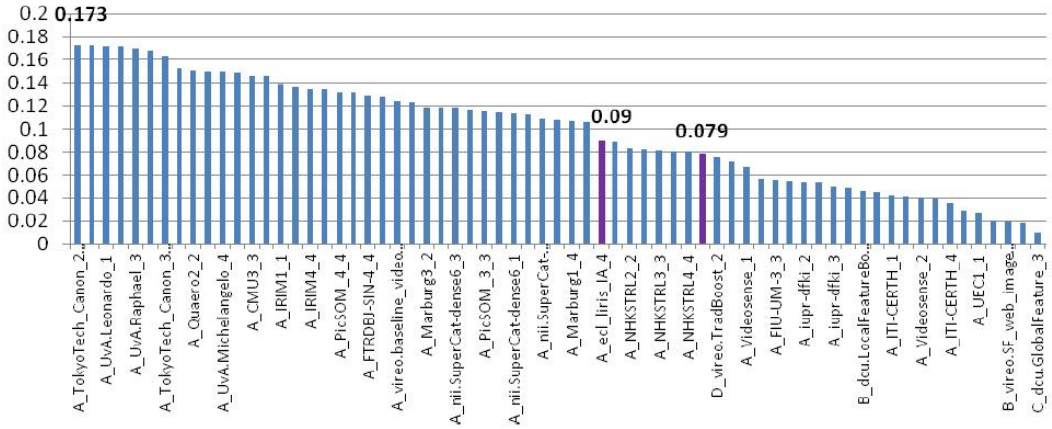



Figure A.6: Full run results of TRECVID challenge 2011

Appendix B

# Comparison of the Popular Features for Object Recognition

---

In section 2.2.1, we introduce several popular global features proposed in the literature, including Color Histogram (CH), Color Moments (CM), Color Coherence Vectors (CCV), Color Auto-Correlogram (CAC), Gray Level Co-occurrence Matrix (GLCM), Texture Auto-Correlation (TAC), Gabor, Edge Histogram (EH), and Line Segments (LS). In section 2.2.2.2, a set of popular local features are presented. We evaluate and compare these features here in the context of visual object recognition by carrying out the experiments on the PASCAL VOC 2007 benchmark (see section 3.1 for an introduction).

Regarding the implementation of the global features, the *RGB* color space is adopted for computing all the color features. For CH, each color channel is quantized into 11 bins, resulting in a 1331-dimensional histogram. For CM, three orders of color moments are computed respectively in each color channel with a $5 \times 5$ image division, leading to a 225-dimensional vector. For CCV, each color channel is quantized into 4 bins, so that the final vector is of 128-dimension. For CAC, each color channel is quantized into 4 bins, and the maximal distance between two pixels is set to 8, resulting in a 512-dimensional vector. For GLCM, 4 directions (horizontal, vertical and two diagonals) with 1 offset between two pixels are considered. For TAC, (0,2,4,6,8) are applied as position difference in both $x$ and $y$ directions. For Gabor, 5 scales and 8 orientations are used. For EH, 5 types of edge (horizontal, vertical, 45-degree diagonal, 135-degree diagonal and non-directional) are extracted. For LS, 6 orientation bins and 4 length bins are selected for the detected line segments.

For the local features, we select the SIFT, three color SIFT (C-SIFT, Oppo-

## Appendix B. Comparison of the Popular Features for Object Recognition

Table B.1: Comparison of popular global features in the context of object recognition on the PASCAL VOC 2007 benchmark

| AP (%) | CH | CM | CCV | CAC | GLCM | TAC | Gabor | EH | LS |
|---|---|---|---|---|---|---|---|---|---|
| airplane | 45.3 | 52.5 | 45.7 | 43.9 | 44.2 | 25.5 | 39.3 | 33.8 | 36.4 |
| bicycle | 21.7 | 21.1 | 10.3 | 16.7 | 11.4 | 16.3 | 17.5 | 12.8 | 18.7 |
| bird | 24.0 | 15.2 | 19.6 | 22.7 | 18.1 | 19.7 | 15.3 | 18.3 | 15.9 |
| boat | 30.3 | 30.7 | 29.0 | 22.8 | 9.0 | 15.6 | 12.3 | 13.5 | 35.7 |
| bottle | 19.1 | 12.8 | 10.9 | 8.6 | 8.0 | 7.5 | 7.8 | 6.1 | 12.7 |
| bus | 17.6 | 18.7 | 20.4 | 15.5 | 18.4 | 13.3 | 11.6 | 9.6 | 24.8 |
| car | 40.7 | 44.1 | 36.3 | 30.6 | 41.5 | 38.9 | 33.5 | 30.1 | 38.9 |
| cat | 22.8 | 19.2 | 22.3 | 15.8 | 18.9 | 13.7 | 15.8 | 13.6 | 23.6 |
| chair | 23.1 | 26.4 | 25.6 | 22.3 | 29.5 | 19.4 | 19.0 | 13.5 | 32.3 |
| cow | 9.2 | 9.9 | 15.6 | 14.2 | 6.9 | 9.1 | 8.1 | 12.9 | 13.8 |
| table | 25.2 | 21.4 | 27.5 | 23.9 | 19.5 | 7.1 | 12.2 | 5.8 | 17.5 |
| dog | 24.0 | 25.2 | 24.0 | 15.2 | 23.2 | 14.9 | 18.5 | 13.1 | 26.6 |
| horse | 57.2 | 55.9 | 44.7 | 45.6 | 31.8 | 12.4 | 31.6 | 27.8 | 21.1 |
| motor | 31.3 | 31.1 | 18.6 | 14.6 | 19.2 | 10.5 | 11.9 | 16.7 | 16.0 |
| person | 71.0 | 61.5 | 65.4 | 62.1 | 53.5 | 56.9 | 56.5 | 53.6 | 65.9 |
| plant | 22.6 | 11.0 | 20.5 | 19.4 | 9.4 | 7.2 | 8.7 | 6.1 | 8.6 |
| sheep | 22.9 | 15.3 | 20.6 | 20.5 | 13.3 | 9.6 | 10.7 | 12.9 | 17.8 |
| sofa | 11.7 | 22.4 | 14.1 | 12.6 | 11.0 | 8.8 | 11.8 | 10.2 | 13.2 |
| train | 33.4 | 38.8 | 33.8 | 26.1 | 24.0 | 16.3 | 19.0 | 21.9 | 23.7 |
| monitor | 13.5 | 18.7 | 18.8 | 14.3 | 16.2 | 6.7 | 15.4 | 10.1 | 22.8 |
| **Mean** | **28.3** | **27.6** | **26.2** | **23.4** | **21.4** | **16.5** | **18.8** | **17.1** | **24.3** |

nentSIFT and RGB-SIFT), and HOG descriptors for evaluation. For their extraction, we use the source codes available online [1] with the default parameter setting.

For classification, the Support Vector Machine (SVM) algorithm (see section 2.3.2.1 for an introduction) is applied. Once all the features are extracted from the dataset, and are transformed into fixed-length histograms by the Bag-of-Features modelling method (required for local features, 4000 visual words, see section 2.2.2.3 for an introduction), the Chi-square ($\chi^2$) kernel is computed as equation (2.36) and (2.37) for the SVM training and prediction. Finally, the precision-recall curve is plotted according to the output decision values of the SVM classifier, and the Average Precision (AP) value is computed based on the proportion of the area under this curve. For each category in the dataset, we train the classifier on the training

[1] Code for SIFT and color SIFT: http://www.colordescriptors.com/
Code for HOG: http://www.robots.ox.ac.uk/~vgg/research/caltech/phog.html/

## Appendix B. Comparison of the Popular Features for Object Recognition

Table B.2: Comparison of popular local features in the context of object recognition on the PASCAL VOC 2007 benchmark ("OP-SIFT" is the abbreviation of "OpponentSIFT", "HL" stands for "Harris-Laplace Interest Points", "DS" stands for "Dense Sampling")

| AP (%) | SIFT (HL) | SIFT (DS) | OP-SIFT (HL) | OP-SIFT (DS) | C-SIFT (HL) | C-SIFT (DS) | RGB-SIFT (HL) | RGB-SIFT (DS) | HOG |
|---|---|---|---|---|---|---|---|---|---|
| airplane | 56.0 | 60.9 | 59.9 | 64.3 | 58.7 | 63.2 | 57.8 | 65.2 | 52.1 |
| bicycle | 44.9 | 41.3 | 43.8 | 41.5 | 38.9 | 40.2 | 44.6 | 40.6 | 26.9 |
| bird | 28.2 | 29.8 | 27.7 | 38.9 | 32.1 | 42.5 | 22.5 | 30.4 | 25.0 |
| boat | 45.7 | 55.1 | 49.1 | 54.9 | 51.8 | 56.1 | 46.6 | 54.9 | 40.6 |
| bottle | 19.6 | 15.4 | 21.2 | 22.5 | 21.4 | 22.5 | 21.0 | 17.7 | 12.8 |
| bus | 37.7 | 39.9 | 38.0 | 40.2 | 32.5 | 36.8 | 37.7 | 42.4 | 38.3 |
| car | 55.0 | 63.4 | 57.4 | 62.2 | 53.2 | 60.1 | 56.1 | 64.7 | 58.1 |
| cat | 36.5 | 40.4 | 37.7 | 38.6 | 34.1 | 35.5 | 37.3 | 42.3 | 27.5 |
| chair | 44.5 | 45.6 | 42.4 | 43.5 | 45.9 | 44.3 | 43.5 | 43.4 | 43.8 |
| cow | 25.9 | 25.8 | 17.0 | 24.4 | 16.6 | 21.6 | 27.8 | 25.8 | 19.8 |
| table | 29.6 | 24.9 | 36.7 | 33.2 | 38.7 | 26.9 | 29.1 | 29.4 | 33.6 |
| dog | 26.5 | 32.6 | 29.8 | 34.3 | 29.1 | 30.5 | 28.8 | 37.0 | 20.4 |
| horse | 57.0 | 62.4 | 59.1 | 63.4 | 61.9 | 69.9 | 54.8 | 61.3 | 59.3 |
| motor | 30.2 | 40.7 | 33.9 | 44.7 | 44.4 | 42.3 | 32.1 | 40.7 | 37.2 |
| person | 73.1 | 75.3 | 74.5 | 76.4 | 76.6 | 76.5 | 72.7 | 75.8 | 66.2 |
| plant | 11.5 | 14.6 | 19.9 | 14.5 | 27.1 | 26.2 | 11.5 | 14.6 | 10.4 |
| sheep | 27.4 | 29.3 | 31.2 | 35.0 | 30.9 | 33.1 | 19.4 | 29.5 | 18.4 |
| sofa | 23.6 | 34.9 | 22.9 | 29.3 | 23.2 | 31.8 | 24.6 | 31.5 | 26.3 |
| train | 53.4 | 56.0 | 54.5 | 57.8 | 58.5 | 60.2 | 51.1 | 57.5 | 52.7 |
| monitor | 33.7 | 37.4 | 35.0 | 38.0 | 27.3 | 36.6 | 35.6 | 37.8 | 32.3 |
| **Mean** | **38.0** | **41.3** | **39.6** | **42.9** | **40.1** | **42.8** | **37.7** | **42.1** | **35.1** |

set, then tune the parameters on the validation set, and obtain the classification results on the test set. The detailed results are presented in Table B.1 and B.2.

# Publications

During this thesis, 5 papers have been published, including 1 paper in an international journal and 4 papers in international conferences. In addition, 3 papers have been submitted for review, including 2 papers to international journals and 1 paper to an international conference.

**Accepted Paper in International Journal:**

1. C. Zhu, H. Fu, C.E. Bichot, E. Dellandréa, and L. Chen: "Visual Object Recognition Using Multi-scale Local Binary Patterns and Line Segment Feature", International Journal of Signal and Imaging Systems Engineering (IJSISE), to appear, 2011.

**Accepted Papers in International Conferences:**

1. C. Zhu, C.E. Bichot, and L. Chen: "Visual Object Recognition Using DAISY Descriptor", in Proc. of IEEE International Conference on Multimedia and Expo (ICME), pp.1-6, Barcelona, Spain, 11-15 July 2011.
2. C. Zhu, C.E. Bichot, and L. Chen: "Multi-scale Color Local Binary Patterns for Visual Object Classes Recognition", in Proc. of 20th International Conference on Pattern Recognition (ICPR), pp.3065-3068, Istanbul, Turkey, 23-26 Aug. 2010.
3. C. Zhu, H. Fu, C.E. Bichot, E. Dellandréa, and L. Chen: "Visual Object Recognition Using Local Binary Patterns and Segment-based Feature", in Proc. of International Conference on Image Processing Theory, Tools and Applications (IPTA), pp.426-431, Paris, France, 7-10 July 2010.
4. H. Fu, C. Zhu, E. Dellandréa, C.E. Bichot, and L. Chen: "Visual Object Categorization via Sparse Representation", in Proc. of International Conference on Image and Graphics (ICIG), pp.943-948, Xi'an, China, 20-23 Sept. 2009.

**Submitted Papers in International Journals:**

1. C. Zhu, C.E. Bichot, and L. Chen: "Image Region Description Using Orthogonal Combination of Local Binary Patterns Enhanced with Color Information", submitted to Pattern Recognition (PR), 2011.
2. N. Liu, C. Zhu, Y. Zhang, E. Dellandréa, C.E. Bichot, S. Bres, B. Tellez, and L. Chen: "Multimodal Recognition of Visual Concepts Using Histograms of Textual Concepts and Selective Weighted Late Fusion Scheme", submitted to Computer Vision and Image Understanding (CVIU), 2011.

**Submitted Paper in International Conference:**

1. C. Zhu, D. Huang, C.E. Bichot, Y. Wang, and L. Chen: "HSOG: A Novel Local Image Descriptor based on Histograms of Second Order Gradients for Object Recognition", submitted to European Conference on Computer Vision (ECCV), 2012.

**Other Papers:**

1. C. Zhu, C.E. Bichot, and L. Chen: "Color Orthogonal Local Binary Patterns Combination for Image Region Description", Technical Report, LIRIS UMR5205 CNRS, Ecole Centrale de Lyon, 2011.

2. C. Zhu, B. Gao, N. Liu, Y. Zhang, C.E. Bichot, E. Dellandréa, and L. Chen: "ECL-LIRIS at TRECVID 2011: Semantic Indexing", TRECVID Workshop Notebook Paper, 2011.

3. N. Liu, E. Dellandréa, C. Zhu, Y. Zhang, C.E. Bichot, S. Bres, B. Tellez, and L. Chen: "LIRIS-Imagine at ImageCLEF 2011 Photo Annotation task", ImageCLEF Workshop Paper, 2011.

# Bibliography


[Abdel-Hakim & Farag 2006] Alaa E. Abdel-Hakim and Aly A. Farag. *CSIFT: A SIFT Descriptor with Color Invariant Characteristics.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1978–1983, 2006. 92

[Agarwal & Roth 2002] Shivani Agarwal and Dan Roth. *Learning a Sparse Representation for Object Detection.* In Proc. European Conference on Computer Vision (ECCV), pages 113–130, 2002. 18

[Agarwal & Triggs 2006] Ankur Agarwal and Bill Triggs. *Hyperfeatures - Multilevel Local Coding for Visual Recognition.* In Proc. European Conference on Computer Vision (ECCV), pages 30–43, 2006. 33

[Agin 1972] G.J. Agin. *Representation and Description of Curved Objects.* In PhD Thesis, Stanford University, 1972. ix, 16, 17

[Ahonen *et al.* 2004] Timo Ahonen, Abdenour Hadid and Matti Pietikäinen. *Face Recognition with Local Binary Patterns.* In Proc. European Conference on Computer Vision (ECCV), pages 469–481, 2004. 7, 78

[Ahonen *et al.* 2006] Timo Ahonen, Abdenour Hadid and Matti Pietikäinen. *Face Description with Local Binary Patterns: Application to Face Recognition.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 28, no. 12, pages 2037–2041, 2006. 7, 78

[Ayache *et al.* 2007] Stéphane Ayache, Georges Quénot and Jérôme Gensel. *Classifier Fusion for SVM-Based Multimedia Semantic Indexing.* In Proc. European Conference on Advances in Information Retrieval (ECIR), pages 494–504, 2007. 61

[Bach *et al.* 2004] Francis R. Bach, Gert R. G. Lanckriet and Michael I. Jordan. *Multiple Kernel Learning, Conic Duality, and the SMO Algorithm.* In Proc. International Conference on Machine Learning (ICML), 2004. 54

[Bay *et al.* 2006] Herbert Bay, Tinne Tuytelaars and Luc J. Van Gool. *SURF: Speeded Up Robust Features.* In Proc. European Conference on Computer Vision (ECCV), pages 404–417, 2006. 35

[Bay *et al.* 2008] Herbert Bay, Andreas Ess, Tinne Tuytelaars and Luc J. Van Gool. *SURF: Speeded-Up Robust Features.* Computer Vision and Image Understanding (CVIU), vol. 110, no. 3, pages 346–359, 2008. 35, 102, 134

[Beaudet 1978] P.R. Beaudet. *Rotationally Invariant Image Operators.* In Proc. International Joint Conference on Pattern Recognition, pages 579–583, 1978. 32

[Belhumeur *et al.* 1997] Peter N. Belhumeur, João P. Hespanha and David J. Kriegman. *Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 19, no. 7, pages 711–720, 1997. 18

[Belongie *et al.* 2002] Serge Belongie, Jitendra Malik and Jan Puzicha. *Shape Matching and Object Recognition Using Shape Contexts.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 24, no. 4, pages 509–522, 2002. 6, 36, 134

[Binford 1971] T.O. Binford. *Visual Perception by Computer.* In Proc. IEEE Conference on Systems and Control, 1971. 16

[Bishop 1995] Christopher M. Bishop. Neural networks for pattern recognition. Clarendon Press, Oxford, UK, 1995. 6

[Boiman *et al.* 2008] Oren Boiman, Eli Shechtman and Michal Irani. *In Defense of Nearest-Neighbor Based Image Classification.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1–8, 2008. 42, 68

[Bosch *et al.* 2008] Anna Bosch, Andrew Zisserman and Xavier Muñoz. *Scene Classification Using a Hybrid Generative / Discriminative Approach.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 30, no. 4, pages 712–727, 2008. 92

[Bouchard & Triggs 2005] Guillaume Bouchard and Bill Triggs. *Hierarchical Part-Based Visual Object Categorization.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 710–715, 2005. ix, 20, 22

[Boughorbel *et al.* 2004] Sabri Boughorbel, Jean-Philippe Tarel and Francois Fleuret. *Non-Mercer Kernels for SVM Object Recognition.* In Proc. British Machine Vision Conference (BMVC), pages 137–146, 2004. 57

[Brown *et al.* 2011] Matthew Brown, Gang Hua and Simon A. J. Winder. *Discriminative Learning of Local Image Descriptors.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 33, no. 1, pages 43–57, 2011. xi, 38, 134, 139

[Burghouts & Geusebroek 2009] Gertjan J. Burghouts and Jan-Mark Geusebroek. *Performance Evaluation of Local Color Invariants.* Computer Vision and Image Understanding (CVIU), vol. 113, no. 1, pages 48–62, 2009. 92

[Chang & Lin 2001] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: A Library for Support Vector Machines.* `http://www.csie.ntu.edu.tw/~cjlin/libsvm`, 2001. 54, 86, 109, 126

[Chatfield *et al.* 2011] Ken Chatfield, Victor Lemtexpitsky, Andrea Vedaldi and Andrew Zisserman. *The Devil is in the Details: An Evaluation of Recent Feature Encoding Methods.* In Proc. British Machine Vision Conference (BMVC), 2011. vii, 44, 67

[Chevalier *et al.* 2007] Fanny Chevalier, Jean-Philippe Domenger, Jenny Benois-Pineau and Maylis Delest. *Retrieval of Objects in Video by Similarity Based on Graph Matching.* Pattern Recognition Letters, vol. 28, no. 8, pages 939–949, 2007. 3

[Cortes & Vapnik 1995] Corinna Cortes and Vladimir Vapnik. *Support-Vector Networks.* Machine Learning, vol. 20, no. 3, pages 273–297, 1995. 6, 23, 50

[Cover & Hart 1967] Thomas M. Cover and Peter E. Hart. *Nearest Neighbor Pattern Classification.* IEEE Trans. on Information Theory, vol. 13, no. 1, pages 21–27, 1967. 6, 56

[Csurka *et al.* 2004] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski and Cédric Bray. *Visual Categorization with Bags of Keypoints.* In Proc. Workshop on Statistical Learning in Computer Vision, ECCV, pages 1–22, 2004. 3, 21, 39, 109, 124, 142

[Dalal & Triggs 2005] Navneet Dalal and Bill Triggs. *Histograms of Oriented Gradients for Human Detection.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 886–893, 2005. 6, 35, 102, 120, 134, 153

[Daugman 1988] J.G. Daugman. *Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression.* IEEE Trans. on Acoustics, Speech and Signal Processing, vol. 36, no. 7, pages 1169–1179, 1988. 28

[Deng *et al.* 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li and Fei-Fei Li. *ImageNet: A Large-Scale Hierarchical Image Database.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 248–255, 2009. 9, 68, 120

[Everingham *et al.* 2007] 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. 33, 66, 79, 86, 108, 121, 124

[Everingham *et al.* 2010] Mark Everingham, Luc J. Van Gool, Christopher K. I. Williams, John M. Winn and Andrew Zisserman. *The Pascal Visual Object Classes (VOC) Challenge.* International Journal of Computer Vision (IJCV), vol. 88, no. 2, pages 303–338, 2010. 3, 21, 44, 65, 92, 120

[Farquhar *et al.* 2005] J. D. H. Farquhar, Sandor Szedmak, Hongying Meng and John Shawe-Taylor. *Improving "Bag-of-keypoints" Image Categorization: Generative Models and PDF-Kernels.* Technical Report, University of Southampton, 2005. 40, 42, 47

[Fellbaum 1998] Christiane Fellbaum. Wordnet: An electronic lexical database. MIT Press, Cambridge, MA, 1998. 68

[Felzenszwalb & Huttenlocher 2005] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. *Pictorial Structures for Object Recognition.* International Journal of Computer Vision (IJCV), vol. 61, no. 1, pages 55–79, 2005. ix, 20, 21, 22

[Felzenszwalb *et al.* 2008] Pedro F. Felzenszwalb, David A. McAllester and Deva Ramanan. *A Discriminatively Trained, Multiscale, Deformable Part Model.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008. 21

[Felzenszwalb *et al.* 2010] Pedro F. Felzenszwalb, Ross B. Girshick, David A. McAllester and Deva Ramanan. *Object Detection with Discriminatively Trained Part-Based Models.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 32, no. 9, pages 1627–1645, 2010. 21, 158

[Fergus *et al.* 2003] Robert Fergus, Pietro Perona and Andrew Zisserman. *Object Class Recognition by Unsupervised Scale-Invariant Learning.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 264–271, 2003. ix, 20, 22

[Fischler & Elschlager 1973] M.A. Fischler and R.A. Elschlager. *The Representation and Matching of Pictorial Structures.* IEEE Trans. on Computers, vol. 22, no. 1, pages 67–92, 1973. ix, 20, 21, 22

[Freeman & Adelson 1991] William T. Freeman and Edward H. Adelson. *The Design and Use of Steerable Filters.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 13, no. 9, pages 891–906, 1991. 38

[Freund & Schapire 1997] Yoav Freund and Robert E. Schapire. *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting.* Journal of Computer and System Sciences, vol. 55, no. 1, pages 119–139, 1997. 6, 56

[Fulkerson *et al.* 2008] Brian Fulkerson, Andrea Vedaldi and Stefano Soatto. *Localizing Objects with Smart Dictionaries.* In Proc. European Conference on Computer Vision (ECCV), pages 179–192, 2008. 41

[Furuya & Ohbuchi 2009] Takahiko Furuya and Ryutarou Ohbuchi. *Dense Sampling and Fast Encoding for 3D Model Retrieval Using Bag-of-Visual Features.* In Proc. International Conference on Image and Video Retrieval (CIVR), 2009. 33, 120

[Gehler & Nowozin 2009] Peter V. Gehler and Sebastian Nowozin. *On Feature Combination for Multiclass Object Classification.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 221–228, 2009. 68

[Gokalp & Aksoy 2007] Demir Gokalp and Selim Aksoy. *Scene Classification Using Bag-of-Regions Representations.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007. 46

[Goldberger *et al.* 2003] Jacob Goldberger, Shiri Gordon and Hayit Greenspan. *An Efficient Image Similarity Measure Based on Approximations of KL-Divergence Between Two Gaussian Mixtures.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 487–493, 2003. 47

[Gool *et al.* 1996] Luc J. Van Gool, Theo Moons and Dorin Ungureanu. *Affine / Photometric Invariants for Planar Intensity Patterns.* In Proc. European Conference on Computer Vision (ECCV), pages 642–651, 1996. 38

[Gorisse *et al.* 2010] David Gorisse, Matthieu Cord and Frédéric Precioso. *Scalable Active Learning Strategy for Object Category Retrieval.* In Proc. International Conference on Image Processing (ICIP), pages 1013–1016, 2010. 3

[Grauman & Darrell 2005a] Kristen Grauman and Trevor Darrell. *Efficient Image Matching with Distributions of Local Invariant Features.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 627–634, 2005. 58

[Grauman & Darrell 2005b] Kristen Grauman and Trevor Darrell. *The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 1458–1465, 2005. 44, 59

[Griffin *et al.* 2007] Gregory Griffin, Alex Holub and Pietro Perona. *Caltech-256 Object Category Dataset.* Technical Report, California Institute of Technology, 2007. 68

[Gu *et al.* 2009] Chunhui Gu, Joseph J. Lim, Pablo Arbelaez and Jitendra Malik. *Recognition Using Regions.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1030–1037, 2009. x, 46, 47

[Guillaumin *et al.* 2010] Matthieu Guillaumin, Jakob J. Verbeek and Cordelia Schmid. *Multimodal Semi-supervised Learning for Image Classification.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 902–909, 2010. 3

[Guzman 1971] A. Guzman. *Analysis of Curved Line Drawings Using Context and Global Information.* In B. Meltzer and D. Mitchie, editeurs, Machine Intelligence, pages 325–376. Edinburgh University Press, 1971. ix, 16, 17

[Harris & Stephens 1988] Chris Harris and Mike Stephens. *A Combined Corner and Edge Detection.* In Proc. Alvey Vision Conference (AVC), pages 147–151, 1988. 31

[Harzallah *et al.* 2009] Hedi Harzallah, Frédéric Jurie and Cordelia Schmid. *Combining Efficient Object Localization and Image Classification.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 237–244, 2009. vii, 3, 67

[Haussler 1999] David Haussler. *Convolution Kernels on Discrete Structures.* Technical Report, 1999. 57

[Hegerath *et al.* 2006] Andre Hegerath, Thomas Deselaers and Hermann Ney. *Patch-based Object Recognition Using Discriminatively Trained Gaussian Mixtures.* In Proc. British Machine Vision Conference (BMVC), pages 519–528, 2006. 3

[Heikkilä *et al.* 2009] Marko Heikkilä, Matti Pietikäinen and Cordelia Schmid. *Description of Interest Regions with Local Binary Patterns.* Pattern Recognition, vol. 42, no. 3, pages 425–436, 2009. 8, 36, 93, 96, 97, 99, 102, 103, 105, 134, 142, 147

[Hofmann 1999] Thomas Hofmann. *Probabilistic Latent Semantic Indexing.* In Proc. International Conference on Research and Development in Information Retrieval, pages 50–57, 1999. 55

[Hu *et al.* 2011] Rui Hu, Tinghuai Wang and John P. Collomosse. *A Bag-of-Regions Approach to Sketch-based Image Retrieval.* In Proc. International Conference on Image Processing (ICIP), pages 3661–3664, 2011. 46

[Huang *et al.* 1997] Jing Huang, Ravi Kumar, Mandar Mitra, Wei-Jing Zhu and Ramin Zabih. *Image Indexing Using Color Correlograms.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 762–768, 1997. 26

[Huang *et al.* 2011] Di Huang, Caifeng Shan, Mohsen Ardabilian, Yunhong Wang and Liming Chen. *Local Binary Patterns and Its Application to Facial Image Analysis: A Survey.* IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews (TSMCC), vol. 41, no. 4, pages 1–17, 2011. 97

[Huttenlocher & Ullman 1987] Daniel P. Huttenlocher and Shimon Ullman. *Object Recognition Using Alignment.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 102–111, 1987. 16

[Indyk & Thaper 2003] Piotr Indyk and Nitin Thaper. *Fast Image Retrieval via Embeddings.* In International Workshop on Statistical and Computational Theories of Vision, 2003. 58

[Jain *et al.* 2008] Prateek Jain, Brian Kulis and Kristen Grauman. *Fast Image Search for Learned Metrics.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008. 68

[Jebara & Kondor 2003] Tony Jebara and Risi Imre Kondor. *Bhattacharyya Expected Likelihood Kernels.* In Proc. Computational Learning Theory and Kernel Machines (COLT), pages 57–71, 2003. 60

[Jebara *et al.* 2004] Tony Jebara, Risi Imre Kondor and Andrew Howard. *Probability Product Kernels.* Journal of Machine Learning Research (JMLR), vol. 5, pages 819–844, 2004. 60

[Jolliffe 2002] I.T. Jolliffe. Principal component analysis. Springer, second édition, 2002. 10, 34, 141

[Jurie & Triggs 2005] Frédéric Jurie and Bill Triggs. *Creating Efficient Codebooks for Visual Recognition.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 604–610, 2005. 40

[Kadir & Brady 2001] Timor Kadir and Michael Brady. *Saliency, Scale and Image Description.* International Journal of Computer Vision (IJCV), vol. 45, no. 2, pages 83–105, 2001. 32

[Ke & Sukthankar 2004] Yan Ke and Rahul Sukthankar. *PCA-SIFT: A More Distinctive Representation for Local Image Descriptors.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 506–513, 2004. 34

[Khan *et al.* 2009] Fahad Shahbaz Khan, Joost van de Weijer and Maria Vanrell. *Top-Down Color Attention for Object Recognition.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 979–986, 2009. vii, 67

[Koen van de Sande ] University of Amsterdam Koen van de Sande. *ColorDescriptor Software.* `http://www.colordescriptors.com`. 105, 115

[Koenderink & van Doorn 1987] J. Koenderink and A. van Doorn. *Representation of Local Geometry in the Visual System.* Biological Cybernetics, vol. 55, pages 367–375, 1987. 38

[Kullback 1968] Solomon Kullback. Information theory and statistics. Dover Publications, 1968. 60

[Lanckriet *et al.* 2004] Gert R. G. Lanckriet, Tijl De Bie, Nello Cristianini, Michael I. Jordan and William Stafford Noble. *A Statistical Framework for Genomic Data Fusion.* Bioinformatics, vol. 20, no. 16, pages 2626–2635, 2004. 54

[Lazebnik & Raginsky 2009] Svetlana Lazebnik and Maxim Raginsky. *Supervised Learning of Quantizer Codebooks by Information Loss Minimization.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 31, no. 7, pages 1294–1309, 2009. 41

[Lazebnik *et al.* 2006] Svetlana Lazebnik, Cordelia Schmid and Jean Ponce. *Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2169–2178, 2006. x, 3, 44, 45, 68, 124

[Li & Allinson 2008] Jing Li and Nigel M. Allinson. *A Comprehensive Review of Current Local Features for Computer Vision.* Neurocomputing, vol. 71, no. 10-12, pages 1771–1787, 2008. 38, 92

[Li & Perona 2005] Fei-Fei Li and Pietro Perona. *A Bayesian Hierarchical Model for Learning Natural Scene Categories.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 524–531, 2005. 33, 120

[Li *et al.* 2007] Fei-Fei Li, Robert Fergus and Pietro Perona. *Learning Generative Visual Models From Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories.* Computer Vision and Image Understanding (CVIU), vol. 106, no. 1, pages 59–70, 2007. 67, 121, 124, 135, 141

[Lin *et al.* 2007] Yen-Yu Lin, Tyng-Luh Liu and Chiou-Shann Fuh. *Local Ensemble Kernel Learning for Object Category Recognition.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007. 154

[Lindeberg 1998] Tony Lindeberg. *Feature Detection with Automatic Scale Selection.* International Journal of Computer Vision (IJCV), vol. 30, no. 2, pages 79–116, 1998. 32

[Liu *et al.* 2009] Jingen Liu, Yang Yang and Mubarak Shah. *Learning Semantic Visual Vocabularies Using Diffusion Distance.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 461–468, 2009. 41

[Lowe 1999] David G. Lowe. *Object Recognition from Local Scale-Invariant Features.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 1150–1157, 1999. 32, 34

[Lowe 2004] David G. Lowe. *Distinctive Image Features From Scale-Invariant Keypoints.* International Journal of Computer Vision (IJCV), vol. 60, no. 2, pages 91–110, 2004. xi, 3, 6, 21, 32, 34, 92, 99, 102, 120, 122, 134, 142, 147

[Lyu 2005] Siwei Lyu. *Mercer Kernels for Object Recognition with Local Features.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 223–229, 2005. 57

[MacQueen 1967] J.B. MacQueen. *Some Methods for Classification and Analysis of Multivariate Observations.* In Proc. the fifth Berkeley Symposium on Mathematical Statistics and Probability, pages 281–297, 1967. 39, 55

[Mäenpää *et al.* 2000a] Topi Mäenpää, Timo Ojala, Matti Pietikäinen and Maricor Soriano. *Robust Texture Classification by Subsets of Local Binary Patterns.* In Proc. International Conference on Pattern Recognition (ICPR), pages 3947–3950, 2000. 7, 78

[Mäenpää *et al.* 2000b] Topi Mäenpää, Matti Pietikäinen and Timo Ojala. *Texture Classification by Multi-Predicate Local Binary Pattern Operators.* In Proc. International Conference on Pattern Recognition (ICPR), pages 3951–3954, 2000. 7, 78

[Malisiewicz & Efros 2008] Tomasz Malisiewicz and Alexei A. Efros. *Recognition by Association via Learning Per-exemplar Distances.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008. 154

[Manjunath & Ma 1996] B. S. Manjunath and Wei-Ying Ma. *Texture Features for Browsing and Retrieval of Image Data.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 18, no. 8, pages 837–842, 1996. 28

[Manjunath *et al.* 2001] B. S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan and Akio Yamada. *Color and Texture Descriptors.* IEEE Trans. on Circuits and Systems for Video Technology, vol. 11, no. 6, pages 703–715, 2001. 26, 28

[Marée *et al.* 2005] Raphaël Marée, Pierre Geurts, Justus H. Piater and Louis Wehenkel. *Random Subwindows for Robust Image Classification.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 34–40, 2005. 33

[Marszalek & Schmid 2006] Marcin Marszalek and Cordelia Schmid. *Spatial Weighting for Bag-of-Features.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2118–2125, 2006. 3, 44

[Marszalek & Schmid 2007] Marcin Marszalek and Cordelia Schmid. *Semantic Hierarchies for Visual Object Recognition.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007. 3

[Marszalek *et al.* 2007] Marcin Marszalek, Cordelia Schmid, Hedi Harzallah and Joost Van De Weijer. *Learning Object Representations for Visual Object Class Recognition.* In Proc. Visual Recognition Challange Workshop, in conjunction with ICCV, 2007. vii, 67

[Matas *et al.* 2002] Jiri Matas, Ondrej Chum, Martin Urban and Tomás Pajdla. *Robust Wide Baseline Stereo from Maximally Stable Extremal Regions.* In Proc. British Machine Vision Conference (BMVC), pages 384–393, 2002. 32

[McCallum & Nigam 1998] Andrew McCallum and Kamal Nigam. *A Comparison of Event Models for Naive Bayes Text Classification.* In Proc. AAAI Workshop on Learning for Text Categorization, pages 41–48, 1998. 39

[Mikolajczyk & Schmid 2001] Krystian Mikolajczyk and Cordelia Schmid. *Indexing Based on Scale Invariant Interest Points.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 525–531, 2001. 32

[Mikolajczyk & Schmid 2002] Krystian Mikolajczyk and Cordelia Schmid. *An Affine Invariant Interest Point Detector.* In Proc. European Conference on Computer Vision (ECCV), pages 128–142, 2002. 32, 105

[Mikolajczyk & Schmid 2004] Krystian Mikolajczyk and Cordelia Schmid. *Scale & Affine Invariant Interest Point Detectors.* International Journal of Computer Vision (IJCV), vol. 60, no. 1, pages 63–86, 2004. 32, 141

[Mikolajczyk & Schmid 2005] Krystian Mikolajczyk and Cordelia Schmid. *A Performance Evaluation of Local Descriptors.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 27, no. 10, pages 1615–1630, 2005. 6, 35, 38, 92, 120

[Mikolajczyk *et al.* 2005] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir and Luc J. Van Gool. *A Comparison of Affine Region Detectors.* International Journal of Computer Vision (IJCV), vol. 65, no. 1-2, pages 43–72, 2005. 33

[Moosmann *et al.* 2006] Frank Moosmann, Bill Triggs and Frédéric Jurie. *Fast Discriminative Visual Codebooks using Randomized Clustering Forests.* In Proc. Annual Conference on Neural Information Processing Systems (NIPS), pages 985–992, 2006. 41

[Moreno *et al.* 2003] Pedro J. Moreno, Purdy Ho and Nuno Vasconcelos. *A Kullback-Leibler Divergence Based Kernel for SVM Classification in Multimedia Applications.* In Proc. Annual Conference on Neural Information Processing Systems (NIPS), 2003. 47, 61

[Mundy & Zisserman 1992] Joseph L. Mundy and Andrew Zisserman. Geometric invariance in computer vision. MIT Press, 1992. 16

[Mundy 2006] Joseph L. Mundy. *Object Recognition in the Geometric Era: A Retrospective.* In Toward Category-Level Object Recognition, pages 3–28, 2006. 16, 17

[Murase & Nayar 1995] Hiroshi Murase and Shree K. Nayar. *Visual Learning and Recognition of 3-d Objects from Appearance.* International Journal of Computer Vision (IJCV), vol. 14, no. 1, pages 5–24, 1995. ix, 17, 18, 19

[Nevatia & Binford 1977] Ramakant Nevatia and Thomas O. Binford. *Description and Recognition of Curved Objects.* Artificial Intelligence, vol. 8, no. 1, pages 77–98, 1977. 16

[Nowak *et al.* 2006] Eric Nowak, Frédéric Jurie and Bill Triggs. *Sampling Strategies for Bag-of-Features Image Classification.* In Proc. European Conference on Computer Vision (ECCV), pages 490–503, 2006. 33

[Ojala & Pietikäinen 1999] Timo Ojala and Matti Pietikäinen. *Unsupervised Texture Segmentation Using Feature Distributions.* Pattern Recognition, vol. 32, no. 3, pages 477–486, 1999. 7, 78

[Ojala *et al.* 1996] Timo Ojala, Matti Pietikäinen and David Harwood. *A Comparative Study of Texture Measures with Classification Based on Featured Distributions.* Pattern Recognition, vol. 29, no. 1, pages 51–59, 1996. 28, 77, 94

[Ojala *et al.* 2002a] Timo Ojala, Topi Mäenpää, Matti Pietikäinen, Jaakko Viertola, Juha Kyllönen and Sami Huovinen. *Outex - New Framework for Empirical Evaluation of Texture Analysis Algorithms.* In Proc. International Conference on Pattern Recognition (ICPR), pages 701–706, 2002. 97

[Ojala *et al.* 2002b] Timo Ojala, Matti Pietikäinen and Topi Mäenpää. *Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 24, no. 7, pages 971–987, 2002. 7, 8, 28, 77, 78, 84, 92, 93, 96, 97

[Oliva & Torralba 2001] Aude Oliva and Antonio Torralba. *Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope.* International Journal of Computer Vision (IJCV), vol. 42, no. 3, pages 145–175, 2001. 70, 102, 112

[Papageorgiou & Poggio 2000] Constantine Papageorgiou and Tomaso Poggio. *A Trainable System for Object Detection.* International Journal of Computer Vision (IJCV), vol. 38, no. 1, pages 15–33, 2000. 18

[Parikh & Zitnick 2010] Devi Parikh and C. Lawrence Zitnick. *The Role of Features, Algorithms and Data in Visual Recognition.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2328–2335, 2010. 63

[Park *et al.* 2000] Dong Kwon Park, Yoon Seok Jeon and Chee Sun Won. *Efficient Use of Local Edge Histogram Descriptor.* In Proc. ACM International Workshops on Multimedia, pages 51–54, 2000. 29

[Pass *et al.* 1996] Greg Pass, Ramin Zabih and Justin Miller. *Comparing Images Using Color Coherence Vectors.* In Proc. ACM International Conference on Multimedia, pages 65–73, 1996. 25

[Perronnin *et al.* 2006] Florent Perronnin, Christopher R. Dance, Gabriela Csurka and Marco Bressan. *Adapted Vocabularies for Generic Visual Categorization.* In Proc. European Conference on Computer Vision (ECCV), pages 464–475, 2006. 41, 42

[Perronnin *et al.* 2010] Florent Perronnin, Jorge Sánchez and Thomas Mensink. *Improving the Fisher Kernel for Large-Scale Image Classification.* In Proc. European Conference on Computer Vision (ECCV), pages 143–156, 2010. vii, 43, 67

[Pujol & Chen 2007] Alain Pujol and Liming Chen. *Line Segment based Edge Feature Using Hough Transform.* In Proc. International Conference on Visualization, Imaging and Image Processing (VIIP), pages 201–206, 2007. 29

[Quinlan 1986] J. Ross Quinlan. *Induction of Decision Trees.* Machine Learning, vol. 1, pages 81–106, 1986. 55, 56

[Quinlan 1993] J. Ross Quinlan. C4.5: Programs for machine learning. Morgan Kaufmann, 1993. 6, 55, 56

[Rakotomamonjy *et al.* 2008] Alain Rakotomamonjy, Francis Bach, Stephane Canu and Yves Grandvalet. *SimpleMKL.* Journal of Machine Learning Research (JMLR), vol. 9, pages 2491–2521, 2008. 54, 127, 146

[Roberts 1963] Lawrence G. Roberts. Machine perception of three-dimensional solids. Outstanding Dissertations in the Computer Sciences. Garland Publishing, New York, 1963. ix, 16, 17

[Rosenblatt 1962] Frank Rosenblatt. Principles of neurodynamics: Perceptrons and the theory of brain mechanisms. Spartan Books, 1962. 55

[Rubner *et al.* 2000] Yossi Rubner, Carlo Tomasi and Leonidas J. Guibas. *The Earth Mover's Distance as a Metric for Image Retrieval.* International Journal of Computer Vision (IJCV), vol. 40, no. 2, pages 99–121, 2000. 57

[Salton & McGill 1983] Gerard Salton and Michael McGill. Introduction to modern information retrieval. McGraw-Hill Book Company, 1983. 39

[Schaffalitzky & Zisserman 2002] Frederik Schaffalitzky and Andrew Zisserman. *Multi-view Matching for Unordered Image Sets.* In Proc. European Conference on Computer Vision (ECCV), pages 414–431, 2002. 38

[Schmid *et al.* 2000] Cordelia Schmid, Roger Mohr and Christian Bauckhage. *Evaluation of Interest Point Detectors.* International Journal of Computer Vision (IJCV), vol. 37, no. 2, pages 151–172, 2000. 33

[Schneiderman & Kanade 2004] Henry Schneiderman and Takeo Kanade. *Object Detection Using the Statistics of Parts.* International Journal of Computer Vision (IJCV), vol. 56, no. 3, pages 151–177, 2004. 18

[Shan *et al.* 2009] Caifeng Shan, Shaogang Gong and Peter W. McOwan. *Facial Expression Recognition Based on Local Binary Patterns: A Comprehensive Study.* Image and Vision Computing (IVC), vol. 27, no. 6, pages 803–816, 2009. 7, 78

[Sivic & Zisserman 2003] Josef Sivic and Andrew Zisserman. *Video Google: A Text Retrieval Approach to Object Matching in Videos.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 1470–1477, 2003. 3, 39

[Smeaton *et al.* 2006] Alan F. Smeaton, Paul Over and Wessel Kraaij. *Evaluation Campaigns and TRECVid.* In Proc. ACM International Workshop on Multimedia Information Retrieval (MIR), pages 321–330, 2006. 9, 71, 120

[Smeaton *et al.* 2009] Alan F. Smeaton, Paul Over and Wessel Kraaij. *High-Level Feature Detection from Video in TRECVid: A 5-Year Retrospective of Achievements.* In Multimedia Content Analysis, Theory and Applications, pages 151–174. Springer Verlag, 2009. 3

[Smith & Scott-Samuel 2001] Andrew T. Smith and Nicholas E. Scott-Samuel. *First-order and Second-order Signals Combine to Improve Perceptual Accuracy.* Journal of Optical Society America A, vol. 18, no. 9, pages 2267–2272, 2001. 134

[Snoek *et al.* 2005] Cees Snoek, Marcel Worring and Arnold W. M. Smeulders. *Early Versus Late Fusion in Semantic Video Analysis.* In Proc. ACM International Conference on Multimedia, pages 399–402, 2005. 61

[Stricker & Orengo 1995] Markus A. Stricker and Markus Orengo. *Similarity of Color Images.* In Proc. Storage and Retrieval for Image and Video Databases (SPIE), pages 381–392, 1995. 25

[Swain & Ballard 1991] Michael J. Swain and Dana H. Ballard. *Color Indexing.* International Journal of Computer Vision (IJCV), vol. 7, no. 1, pages 11–32, 1991. 24

[Tola *et al.* 2010] Engin Tola, Vincent Lepetit and Pascal Fua. *DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 32, no. 5, pages 815–830, 2010. xi, 9, 120, 122, 134, 142

[Tuceryan & Jain 1998] Mihran Tuceryan and Anil K. Jain. *Texture Analysis.* In Handbook of Pattern Recognition and Computer Vision, 2nd Edition, pages 207–248. World Scientific Publishing Co., River Edge, NJ, USA, 1998. 26, 27, 86

[Turk & Pentland 1991a] Matthew Turk and Alex Pentland. *Eigenfaces for Recognition.* Journal of Cognitive Neuroscience, vol. 3, pages 71–86, 1991. 17

[Turk & Pentland 1991b] Matthew Turk and Alex Pentland. *Face Recognition Using Eigenfaces.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 586–591, 1991. 18

[Tuytelaars & Gool 2000] Tinne Tuytelaars and Luc J. Van Gool. *Wide Baseline Stereo Matching Based on Local, Affinely Invariant Regions.* In Proc. British Machine Vision Conference (BMVC), 2000. 32

[Tuytelaars & Gool 2004] Tinne Tuytelaars and Luc J. Van Gool. *Matching Widely Separated Views Based on Affine Invariant Regions.* International Journal of Computer Vision (IJCV), vol. 59, no. 1, pages 61–85, 2004. 32

[Ullman *et al.* 2001] Shimon Ullman, Erez Sali and Michel Vidal-Naquet. *A Fragment-Based Approach to Object Representation and Classification.* In Proc. International Workshop on Visual Form, pages 85–102, 2001. 20

[van de Sande *et al.* 2008] Koen E. A. van de Sande, Theo Gevers and Cees G. M. Snoek. *Evaluation of Color Descriptors for Object and Scene Recognition.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1–8, 2008. 34

[van de Sande *et al.* 2010] Koen E. A. van de Sande, Theo Gevers and Cees G. M. Snoek. *Evaluating Color Descriptors for Object and Scene Recognition.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 32, no. 9, pages 1582–1596, 2010. ix, x, 3, 23, 33, 34, 38, 44, 45, 79, 92, 102, 108, 142

[van de Weijer *et al.* 2006] Joost van de Weijer, Theo Gevers and Andrew D. Bagdanov. *Boosting Color Saliency in Image Feature Detection.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 28, no. 1, pages 150–156, 2006. 92

[van Gemert *et al.* 2008] Jan van Gemert, Jan-Mark Geusebroek, Cor J. Veenman and Arnold W. M. Smeulders. *Kernel Codebooks for Scene Categorization.* In Proc. European Conference on Computer Vision (ECCV), pages 696–709, 2008. x, 42, 43

[van Gemert *et al.* 2010] Jan van Gemert, Cor J. Veenman, Arnold W. M. Smeulders and Jan-Mark Geusebroek. *Visual Word Ambiguity.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 32, no. 7, pages 1271–1283, 2010. 3, 42

[Vapnik 1995] Vladimir N. Vapnik. The nature of statistical learning theory. Springer-Verlag New York, Inc., New York, NY, USA, 1995. 50

[Varma & Ray 2007] Manik Varma and Debajyoti Ray. *Learning The Discriminative Power-Invariance Trade-Off.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 1–8, 2007. 143

[Vasconcelos *et al.* 2004] Nuno Vasconcelos, Purdy Ho and Pedro J. Moreno. *The Kullback-Leibler Kernel as a Framework for Discriminant and Localized Representations for Visual Recognition.* In Proc. European Conference on Computer Vision (ECCV), pages 430–441, 2004. 47

[Vasconcelos 2004] Nuno Vasconcelos. *On the Efficient Evaluation of Probabilistic Similarity Functions for Image Retrieval.* IEEE Trans. on Information Theory, vol. 50, no. 7, pages 1482–1496, 2004. 47

[Vedaldi & Zisserman 2012] Andrea Vedaldi and Andrew Zisserman. *Efficient Additive Kernels via Explicit Feature Maps.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 34, no. 3, pages 480–492, 2012. 159

[Vedaldi *et al.* 2009] Andrea Vedaldi, Varun Gulshan, Manik Varma and Andrew Zisserman. *Multiple Kernels for Object Detection.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 606–613, 2009. 54

[Vieux *et al.* 2012] Rémi Vieux, Jenny Benois-Pineau and Jean-Philippe Domenger. *Content Based Image Retrieval Using Bag-Of-Regions.* In Proc. International Conference on Advances in Multimedia Modeling (MMM), pages 507–517, 2012. 46

[Viola & Jones 2001] Paul Viola and Michael Jones. *Robust Real-time Object Detection.* International Journal of Computer Vision (IJCV), vol. 57, no. 2, pages 137–154, 2001. 18

[Visual Geometry Group ] University of Oxford Visual Geometry Group. *Comparison of Region Descriptors.* http://www.robots.ox.ac.uk/~vgg/research/affine/desc_evaluation.html. 103

[Vogel & Schiele 2004] Julia Vogel and Bernt Schiele. *Natural Scene Retrieval Based on a Semantic Modeling Step.* In Proc. International Conference on Image and Video Retrieval (CIVR), pages 207–215, 2004. 41

[Wallraven *et al.* 2003] Christian Wallraven, Barbara Caputo and Arnulf B. A. Graf. *Recognition with Local Features: the Kernel Recipe.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 257–264, 2003. 57

[Wang *et al.* 2001] James Ze Wang, Jia Li and Gio Wiederhold. *SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 23, no. 9, pages 947–963, 2001. 70, 108

[Wang *et al.* 2009a] Gang Wang, Derek Hoiem and David A. Forsyth. *Building Text Features for Object Image Classification.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1367–1374, 2009. 3

[Wang *et al.* 2009b] Gang Wang, Derek Hoiem and David A. Forsyth. *Learning Image Similarity from Flickr Groups Using Stochastic Intersection Kernel Machines.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 428–435, 2009. vii, 67

[Wang *et al.* 2010] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas S. Huang and Yihong Gong. *Locality-Constrained Linear Coding for Image Classification.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3360–3367, 2010. vii, 43, 67, 68

[Weber *et al.* 2000] Markus Weber, Max Welling and Pietro Perona. *Unsupervised Learning of Models for Recognition.* In Proc. European Conference on Computer Vision (ECCV), pages 18–32, 2000. 20

[Winn *et al.* 2005] John M. Winn, Antonio Criminisi and Thomas P. Minka. *Object Categorization by Learned Universal Visual Dictionary.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 1800–1807, 2005. 33, 41, 42

[Wu & Rehg 2009] Jianxin Wu and James M. Rehg. *Beyond the Euclidean Distance: Creating Effective Visual Codebooks Using the Histogram Intersection Kernel.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 630–637, 2009. 40

[Yang *et al.* 2007] Jun Yang, Yu-Gang Jiang, Alexander G. Hauptmann and Chong-Wah Ngo. *Evaluating Bag-of-Visual-Words Representations in Scene Classification.* In Proc. ACM International Workshop on Multimedia Information Retrieval (MIR), pages 197–206, 2007. ix, 40

[Yang *et al.* 2008] Liu Yang, Rong Jin, Rahul Sukthankar and Frédéric Jurie. *Unifying Discriminative Visual Codebook Generation with Classifier Training for Object Category Recognition.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1–8, 2008. 41

[Yang *et al.* 2009a] Jianchao Yang, Kai Yu, Yihong Gong and Thomas S. Huang. *Linear Spatial Pyramid Matching Using Sparse Coding for Image Classification.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1794–1801, 2009. 68

[Yang *et al.* 2009b] Jingjing Yang, Yuanning Li, YongHong Tian, Lingyu Duan and Wen Gao. *Group-sensitive Multiple Kernel Learning for Object Categorization.* In Proc. IEEE International Conference on Computer Vision (ICCV), pages 436–443, 2009. vii, 3, 54, 55, 67, 68, 154

[Yilmaz *et al.* 2008] Emine Yilmaz, Evangelos Kanoulas and Javed A. Aslam. *A Simple and Efficient Sampling Method for Estimating AP and NDCG.* In Proc. ACM International Conference on Research and Development in Information Retrieval (SIGIR), pages 603–610, 2008. 71

[Zhang *et al.* 2000] Dengsheng Zhang, Aylwin Wong, Maria Indrawan and Guojun Lu. *Content-Based Image Retrieval Using Gabor Texture Features.* In Proc. IEEE Pacific-Rim Conference on Multimedia (PCM), pages 392–395, 2000. 28, 86

[Zhang *et al.* 2006] Hao Zhang, Alexander C. Berg, Michael Maire and Jitendra Malik. *SVM-KNN: Discriminative Nearest Neighbor Classification for Visual Category Recognition.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2126–2136, 2006. 3, 68, 143

[Zhang *et al.* 2007] Jianguo Zhang, Marcin Marszalek, Svetlana Lazebnik and Cordelia Schmid. *Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study.* International Journal of Computer Vision (IJCV), vol. 73, no. 2, pages 213–238, 2007. 3, 38, 40, 87, 92, 154

[Zhao & Pietikäinen 2007] Guoying Zhao and Matti Pietikäinen. *Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions.* IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 29, no. 6, pages 915–928, 2007. 7, 78

[Zhou *et al.* 2010] Xi Zhou, Kai Yu, Tong Zhang and Thomas S. Huang. *Image Classification Using Super-Vector Coding of Local Image Descriptors.* In Proc. European Conference on Computer Vision (ECCV), pages 141–154, 2010. vii, 43, 67

[Zhu 2004] Mu Zhu. *Reall, Precision and Average Precision.* Technical Report, University of Waterloo, 2004. 67