Contributions to generic visual object categorization

Huanzhang Fu

THÈSE

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

présentée et soutenue publiquement par

Huanzhang FU

le 14 décembre 2010

Contributions to Generic Visual Object Categorization

École Doctorale InfoMaths

Directeur de thèse: Liming CHEN
Co-directeur de thèse: Emmanuel DELLANDRÉA

JURY

Pr. Chabane DJERABA	Université Lille 1	Rapporteur
Dr. Georges QUÉNOT	Laboratoire d'Informatique de Grenoble	Rapporteur
Pr. Su RUAN	Université de Rouen	Examinateur
Pr. Liming CHEN	Ecole Centrale de Lyon	Directeur de thèse
Dr. Emmanuel DELLANDRÉA	Ecole Centrale de Lyon	Co-directeur de thèse

Numéro d'ordre : 2010-44

Acknowledgments

I am greatly in debt to a number of people, without whose help this thesis could not be completed.

First of all, I must show my gratitude to my supervisor Prof. **Liming CHEN** for his instructive advices and useful suggestions during my thesis. Already attracted by his elegant demeanor and profound knowledge when I was a student in Ecole Centrale de Lyon, it is really my honor to have my thesis supervised by him since 2006.

I would like to express also my gratitude here to Prof. **Emmanuel DELLANDRÉA**, my co-supervisor, for his patience, encouragement and priceless advices during the whole work. Anytime I encounter a problem on the research or other aspects, his is always the first person that appears in my head to ask for help. Every time he would give me his precious help with his intrinsic patience and gentilesse.

I owe special thanks to Prof. **Chabane DJERABA** and Dr. **Georges QUÉNOT** who took the time to read and evaluate my work and for their judicious remarks which enabled me to improve this thesis. I also thank Prof. **Su RUAN** for examining my work and giving many meaningful comments.

I am also so grateful to all the persons in the department and in the laboratory LIRIS, with whom I have passed the memorable last four years. The personnel helped me a lot in many problems concerning the administration, the life in France and other intractable situations, while my colleagues have often enlightened me on my research through the exchange of opinions.

At the end, I want to thank my family, who are the most important people for me in this world. My wife **Yan ZHANG**, married me at the beginning of my thesis, has firmly been with me and supported me in the following years in France. My parents-in-law Mr. **Shaoyong ZHANG** and Mrs. **Lianying FAN** have encouraged us not only spiritually but also materially to pass this period relatively difficult. My parents Mr. **Zhiyi FU** and Mrs. **Chundi ZHU** have continually given their support to us just as they had done for me in the past 30 years.

At the end of the end, I would like to thank Mr. God who has sent us his gift

during my thesis, my son the little Mr. **Boxian FU**, who was born with a weight of 3330 grammes on 8:18 on August 28, 2009.

Contents

List of Tables

List of Figures

Abstract

This thesis is dedicated to the active research topic of generic Visual Object Categorization (VOC), which can be widely used in many applications such as video indexation and retrieval, video monitoring, security access control, automobile driving support etc. Due to many realistic difficulties, it is still considered to be one of the most challenging problems in computer vision and pattern recognition. In this context, we have proposed in this thesis our contributions, especially concerning the two main components of the methods addressing VOC problems, namely feature selection and image representation.

Firstly, an Embedded Sequential Forward feature Selection algorithm (ESFS) has been proposed for VOC. Its aim is to select the most discriminant features for obtaining a good performance for the categorization. It is mainly based on the commonly used sub-optimal search method Sequential Forward Selection (SFS), which relies on the simple principle to add incrementally most relevant features. However, ESFS not only adds incrementally most relevant features in each step but also merges them in an embedded way thanks to the concept of combined mass functions from the evidence theory which also offers the benefit of obtaining a computational cost much lower than the one of original SFS.

Secondly, we have proposed novel image representations to model the visual content of an image, namely Polynomial Modeling and Statistical Measures based Image Representation, called PMIR and SMIR respectively. They allow to overcome the main drawback of the popular "bag of features" method which is the difficulty to fix the optimal size of the visual vocabulary. They have been tested along with our proposed region based features and SIFT. Two different fusion strategies, early and late, have also been considered to merge information from different "channels" represented by the different types of features.

Thirdly, we have proposed two approaches for VOC relying on sparse representation, including a reconstructive method (R_SROC) as well as a reconstructive and discriminative one (RD_SROC). Indeed, sparse representation model has been originally used in signal processing as a powerful tool for acquiring, representing and compressing the high-dimensional signals. Thus, we have proposed to adapt these interesting principles to the VOC problem. R_SROC relies on the intuitive assumption that an image can be represented by a linear combination of training images from the same category. Therefore, the sparse representations of images are first computed through solving the ℓ^1 norm minimization problem and then used as new feature vectors for images to be classified by traditional classifiers such as SVM. To improve the discrimination ability of the sparse representation to better fit the classification problem, we have also proposed RD_SROC which includes a discrimination term, such as Fisher discrimination measure or the output of a SVM classifier, to the standard sparse representation objective function in order to learn a reconstructive and discriminative dictionary. Moreover, we have also proposed

to combine the reconstructive and discriminative dictionary and the adapted pure reconstructive dictionary for a given category so that the discrimination power can further be increased.

The efficiency of all the methods proposed in this thesis has been evaluated on popular image datasets including SIMPLIcity, Caltech101 and Pascal2007.

Keywords: visual object categorization, feature selection, image representation, sparse representation.

Résumé

Cette thèse de doctorat est consacrée à un sujet de recherche très porteur : la Catégorisation générique d'Objets Visuels (VOC). En effet, les applications possibles sont très nombreuses, incluant l'indexation d'images et de vidéos, la vidéo surveillance, le contrôle d'accès de sécurité, le soutien à la conduite automobile, etc. En raison de ses nombreux verrous scientifiques, ce sujet est encore considéré comme l'un des problèmes les plus difficiles en vision par ordinateur et en reconnaissance de formes. Dans ce contexte, nous avons proposé dans ce travail de thèse plusieurs contributions, en particulier concernant les deux principaux éléments des méthodes résolvant les problèmes de VOC, notamment la sélection des descripteurs et la représentation d'images.

Premièrement, un algorithme nommé "Embedded Sequential Forward feature Selection" (ESFS) a été proposé pour VOC. Son but est de sélectionner les descripteurs les plus discriminants afin d'obtenir une bonne performance pour la catégorisation. Il est principalement basé sur la méthode de recherche sous-optimale couramment utilisée "Sequential Forward Selection" (SFS), qui repose sur le principe simple d'ajouter progressivement les descripteurs les plus pertinents. Cependant, ESFS non seulement ajoute progressivement les descripteurs les plus pertinents à chaque étape mais de plus les fusionne d'une manière intégrée grace à la notion de fonctions de masses combinées empruntée à la théorie de l'évidence qui offre également l'avantage d'obtenir un coût de calcul beaucoup plus faible que celui de SFS original.

Deuxièmement, nous avons proposé deux nouvelles représentations d'images pour modéliser le contenu visuel d'une image : la Représentation d'Image basée sur la Modélisation Polynomiale et les Mesures Statistiques, appelées respectivement PMIR et SMIR. Elles permettent de surmonter l'inconvénient principal de la méthode populaire "bag of features" qui est la difficulté de fixer la taille optimale du vocabulaire visuel. Elles ont été testées avec nos descripteurs basés région ainsi que les descripteurs SIFT. Deux stratégies différentes de fusion, précoce et tardive, ont également été considérées afin de fusionner les informations venant des "canaux" différents représentés par les différents types de descripteurs.

Troisièmement, nous avons proposé deux approches pour VOC en s'appuyant sur la représentation sparse. La première méthode est reconstructive (R_SROC) alors que la deuxième est reconstructive et discriminative (RD_SROC). En effet, le modèle de représentation sparse a été utilisé originalement dans le domaine du traitement du signal comme un outil puissant pour acquérir, représenter et compresser des signaux de grande dimension. Ainsi, nous avons proposé une adaptation de ces principes intéressants au problème de VOC. R_SROC repose sur l'hypothèse intuitive que l'image peut être représentée par une combinaison linéaire des images d'apprentissage de la même catégorie. Par conséquent, les représentations sparses des images sont d'abord calculées par la résolution du problème de minimisation de la norme ℓ^1 et sont ensuite utilisées en tant que nouveaux vecteurs de descripteur

pour les images afin de permettre la classification de ces dernières par des classificateurs traditionnels tels que SVM. Afin d'améliorer la capacité de discrimination de la représentation sparse pour mieux répondre au problème de classification, nous avons également proposé RD_SROC qui inclue un terme de discrimination, comme la mesure de discrimination Fisher ou la sortie d'un classificateur SVM, à la fonction d'objectif de la représentation sparse standard afin d'entraîner un dictionnaire reconstructif et discriminatif. De plus, nous avons proposé de combiner le dictionnaire reconstructif et discriminatif avec le dictionnaire adapté purement reconstructif pour une catégorie donnée de sorte que la capacité de discrimination puisse être augmentée.

L'efficacité de toutes les méthodes proposées dans cette thèse a été évaluée sur différentes bases populaires d'images comprenant SIMPLIcity, Caltech101 et Pascal2007.

Mots clés: catégorisation d'objets visuels, sélection de descripteurs, représentation d'images, représentation sparse.

Chapter 1

Introduction

Contents

1.1 Context

With the rapid development of new information technology and media, more and more contents presented around us are nowadays changing from text based to multimedia based, especially in the form of images and videos. For example, the famous online photo sharing website Flickr (*www.flickr.com*) was hosting more than 5 billion images on September 2010 with a growing speed of about 1 billion per year.

Facing such huge databases, the need for solutions to effectively manage them and access to the appropriate content when needed becomes more and more urgent. Basically, one would like to label an image manually using the keywords and then search it according to the associated tags for a later use as it is proposed on Flickr website. However, this method quickly becomes inconceivable for large amounts of data. Moreover, many other problems can not be ignored: the database annotation is only possible for a limited number of languages; when an annotation rule changes for a certain application, the annotation process should be performed consequently manually on the whole database; since the annotation can be subjective, there is no guarantee that two different persons produce systematically the same label for one

Figure 1.1: An example of generic visual object categorization

image, which is generally expected in most applications concerning the multimedia data.

In such a context, the research topic of Generic Visual Object Categorization has emerged and attracted more and more attentions in recent years.

1.2 Problems and objective

Generic Visual Object Categorization (VOC) aims at predicting whether at least one or several objects of some given categories are present in an image. More precisely, only categories of objects, or concepts, are taken into account, that is to say that we want to detect any car or any people in an image, rather than a particular car or a particular people which is the goal of object recognition systems. An example is given in Figure 1.1, in which the image should be classified to the predefined category "Person" and "Horse" at the same time as it contains these two objects.

In fact, VOC is a fundamental problem in computer vision and pattern recognition, and has become an important research topic due to the wide range of possible applications such as video monitoring, video coding systems, security access control, automobile driving support as well as automatic image and video indexation and retrieval [Lew *et al.* 2006] [Sayad *et al.* 2010]. Until now, many VOC methods have been proposed and applied to the classification of numerous objects categories like, for example, cars, motorbikes, animals, people, furniture etc. Despite many efforts and much progress that have been made during the past years, it remains an open problem and is still considered as one of the most challenging topics in computer

vision. The reason that it has to deal with problems inherent to object categories like the wide variations of shape and appearance of objects inside a category, and due to the representation of an object in an image, such as various scales and orientations, as well as illumination and occlusion problems. To all these difficulties, we also need to add the one induced by the large number of real world object types that need to be discriminated.

In this context, the objective of our work can be summarized as to propose some innovative contributions to the challenging generic visual object categorization task in particular concerning image representation using either global and local features or the fusion of them. These proposed approaches have been validated through experiments driven on several popular datasets.

1.3 Our approaches and contributions

A typical VOC system is generally composed of two basic stages: one is the extraction of features from an image to represent its visual content and the other is the image classification based on the information carried by these features, according to the considered categories. However, only these two stages are far from enough to construct a successful and efficient VOC system in the practice and supplemental stages are often necessary, namely feature selection and image representation. The former one intends to select the most important and non-redundant features to simplify the classification model and to allow a better classification accuracy. The latter one aims at finding a representation that is, in one hand, able to better model the image visual content which is presented in the form of features extracted from the image and that, in the other hand, gains more discrimination abilities to be easily categorized by a certain classifier later. In the case of using local features, as the number of them often varies from one image to another, image representation also helps changing these original local features to the feature vector with fixed size as usually required by classifiers. Our work mainly concerns these two indispensable aspects and will be listed in the following.

Our first contribution consists in proposing an Embedded Sequential Forward

feature Selection (ESFS) algorithm for VOC use. With the increasing trend of high dimensional data processing, feature selection becomes more and more important and indispensable in pattern recognition and machine learning problems, for the purpose of selecting the most discriminant features. Its objective is three-fold: improving the prediction performance of the predictors, providing faster and more cost-effective predictors, and gaining a deeper insight into the underlying processes that generated the data. In this context, we have implemented in our work a novel embedded feature selection approach based on the commonly used sub-optimal search method SFS [Whitney 1971], called ESFS, which relies on the simple principle to add incrementally most relevant features. We have provided here two advantages comparing to the classical classifier dependent sub-optimal selection method SFS. Firstly, the range of subsets to be evaluated in the forward process is extended to multiple subsets for each size, and the feature set is reduced according to a certain threshold before the selection in order to decrease the computational burden caused by the extension of the subsets in the evaluation. Secondly, we have made use of the term of mass function to consider the feature as a classifier, which is introduced from the evidence theory [Shafer 1976] allowing elegantly to merge feature information in an embedded way, leading to a lower computational cost than original SFS.

Secondly, we have proposed novel image representations for modeling the image visual content. Indeed, the most successful image representation to date is "bag of features". Its main drawback lies in the difficulties one can have to fix the optimal size of visual vocabulary. Moreover, when a GMM is used for a soft assignment, the number of parameters along with the number of Gaussians can quickly lead to the problem of "curse of dimensionality" [Bellman 1961]. Thus, we have proposed novel image representations, namely through polynomial interpolation and statistical measures, for modeling the visual content of an image from another way. Their interest is 3-fold. First, we can circumvent the difficulty of fixing the size of visual vocabulary; secondly we can avoid the inaccurate assumption of Gaussian repartition of features which is not always the case when faced with numerous different applications; finally we are able to cope with a smaller number of feature vectors per image, a situation that can be often encountered.

Our third contribution lies in the proposition of reconstructive and discriminative image adapted sparse representations using classical sparse representation theory. Sparse representation has been originally used in the domain of signal processing as a powerful tool for acquiring, representing and compressing the high-dimensional signals. Its goal is to obtain a compact high-fidelity representation of a given signal, which can be considered as a linear combination of atoms from an overcomplete dictionary. The property of sparsity in the representation of signals has also been approved in human perception by some studies of human vision.

Recently techniques from this theory have significantly impacted the domain of computer vision and pattern recognition [Wright *et al.* 2009a] [Wright *et al.* 2009b] [Mairal *et al.* 2008a], in which we are often more interested in extracting the visual content of an image rather than a compact high-fidelity representation. It has been successfully applied to several vision tasks, including face recognition, image super-resolution and classification, motion segmentation, and background modeling. Thus, he have proposed to adapt the ideas of sparse representation to the problem of VOC.

Two innovations have been proposed in order to improve the classification accuracy using sparse representation theory. Firstly, as the traditional sparse representation is a purely reconstructive method which seems not to perfectly fit the applications of classification, discrimination terms, namely Fisher's discrimination measure and the output of a classifier (in our case SVM), have been introduced to enhance the discrimination ability of the obtained sparse image representation. The dictionary which is initially a subset of training images is updated by K-SVD algorithm [Aharon *et al.* 2006] at the same time. Secondly, inspired by the idea of [Perronnin *et al.* 2006], we have considered first training a reconstructive and discriminative dictionary using both positive training images and negative ones for each category and then training an adapted purely reconstructive dictionary, using the images from that category only. The final dictionary for each category is obtained by combining its reconstructive and discriminative dictionary and the adapted purely reconstructive dictionary. In this case, the assumption is that an image is more appropriately described by the atoms in the adapted dictionary of category C if it belongs to C and otherwise it is better described by the atoms in the reconstruc-

tive and discriminative dictionary. The training of the dictionary is performed by K-SVD algorithm.

1.4 Organization of the thesis

The rest of this thesis is organized as follows. In Chapter 2, we first introduce our feature selection method, namely Embedded Sequential Forward feature Selection (ESFS) algorithm and its use in VOC. Our polynomial modeling and statistical measures based image representations are then presented in the following as well as our proposed region based features which are used in the previous representations.

Chapter 3 deals with sparse representations theory and focuses on the algorithms we have proposed for constructing the reconstructive and discriminative image adapted sparse representations.

Finally, we summarize our conclusions from the results of this work in Chapter 4 and propose some future directions at the same time.

Chapter 2

Feature extraction, selection and image representation for VOC

Contents

2.1 Introduction

Generally, within the VOC process, an image firstly passes through the feature extraction stage to obtain a set of features, on which a possible selection procedure may then be applied to select the most effective features. Then, the image representation for classification can intervene if necessary to model the image visual content and satisfy the input requirement of a certain classifier, which will perform the final classification task. So, feature extraction, selection and image representation are considered to be three principal stages out of four for visual object categorization, the last one being the classification. This chapter deals with these different aspects and the approaches we have proposed for these purposes.

2.2 VOC: a brief state of the art

Before entering the detailed main stages mentioned above, we would like to mention here some representative methods and techniques concerning the visual object categorization, hoping to present an understanding brief overview about this domain.

2.2.1 Feature extraction

The role of feature extraction is to convert the only thing that can be read from images, their colored pixels, to the "low-level" features for subsequent analysis of image content, hoping that there are sufficiently discriminative, effective and with reasonable size. This first step is very important for assuring the final good performance of VOC system and can be considered as the basis of the whole work in some sense. Indeed, after this step, the whole process will rely only on the information given by the features extracted from the image and no longer on the image itself.

The first question that arises is to know where we will extract the effective features for the characterization of image visual content. We can summarize the existing approaches in the literature into two main categories: global feature and local feature.

- **Global feature.** This approach is generally based on the statistical analysis of the whole image pixel by pixel. It assumes implicitly that the searched object occupies ideally the entire image. However, this assumption is so hard to be satisfied in the reality, and the background introduces inevitably noise particularly in the case where the object is very small compared to the size of image. This limitation often justifies to pay more attention to local methods.

- **Local feature.** According to this approach, the feature is calculated from a small neighborhood (called patch in the following) with a predefined size and form around a particular point (pixel) of the image. In this case, the question that arises is "how to detect the particular points (or equivalently patches) around which the local features will be extracted?". In fact, there exist many research works dealing with this problem, including:

1. *Interest points [Mikolajczyk et al. 2005].* Here, we would like to mention the two commonly used local patch detectors: Laplacian detector [Lindeberg 1998] which extracts blob-like patches and Harris-Laplace detector [Mikolajczyk & Schmid 2001] which extracts corner-like patches (see Figure 2.1 for the illustration of these two detectors on two natural images). The Laplacian detector is a scale invariant blob detector, where a blob is defined by a maximum of the normalized Laplacian in scale-space. The Harris-Laplace detector is an extension of the original rotation-invariant Harris detector [Harris & Stephens 1988] by adding the scale-invariant property.

2. *Random sampling.* As the name suggests, patches are selected randomly in this case. It has shown its effectiveness in [Marée *et al.* 2005] and [Nowak *et al.* 2006], where it performs better than interest points detectors according to their experimentations.

3. *Dense sampling.* [Winn *et al.* 2005] and [Fei-Fei & Perona 2005] showed experimentally that using regular grids to select patches could outperforms interest points detectors as well.

Of course, these different strategies can be combined together if necessary, with the purpose of obtaining better performance than using each one separately.

When we known where to extract features, we then want to determine the nature of features to be extracted. Generally, we can categorize them into 3 groups, listed below with some representative features for each of them:

- **Color features**

  - *Color Histogram [Swain & Ballard 1991]*: Histograms are the simplest and most common way for expressing the color characteristics of an image. They aim at modeling the color distribution of image pixels. Generally every channel of a color space, (RGB color space for example), is quantified into "bins". The histogram is built by counting the number of

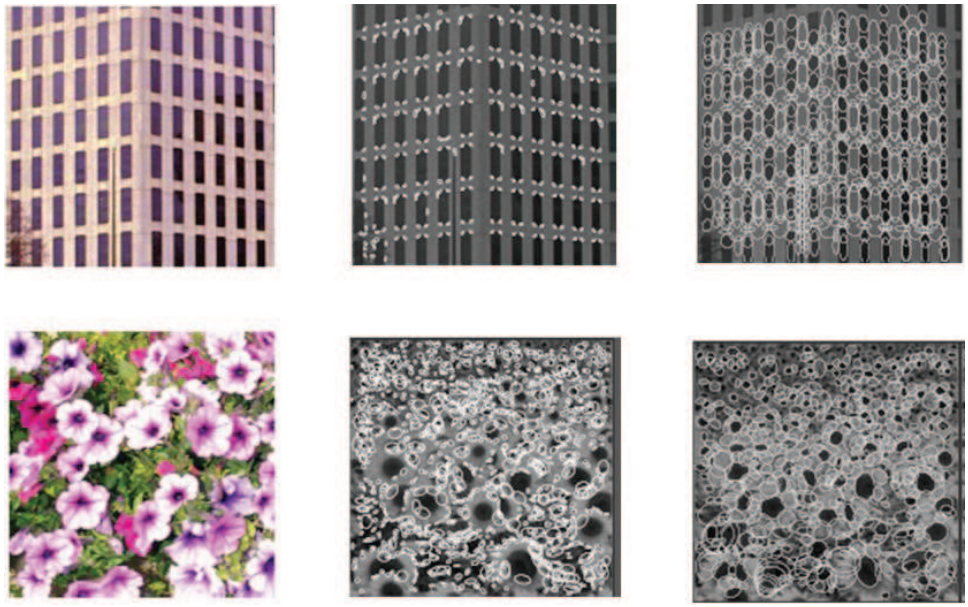

Figure 2.1: Illustration of Harris-Laplace detector and Laplacian detector on two natural images. Left: original images; Middle: Harris-Laplace detector; Right: Laplacian detector. Source: [Zhang *et al.* 2007]

pixels located in each bin. The three 1-D histograms are then concatenated to form the final color histogram. It is easy to compute but ignores the spatial information between pixels.

– *Color Coherence Vectors [Pass & R. Zabih 1997]*: In order to integrate the spatial information of color distribution, color coherence vectors propose to separate the coherent colors and incoherent colors. We say that a color is coherent when its population of pixels located in a spatial neighbor area is bigger than a predefined threshold, otherwise it is incoherent. We thus find a characterization of color information by two histograms: the population of coherent color cells and the populations of incoherent color cells.

– *Color Correlogram and Color Auto Correlogram [Huang et al. 1997]*: As another way to integrate the spatial information of colors, color correlogram can be understood as a 3-dimensional matrix with size $(n \times n \times r)$ where n is the number of colors used and r is the maximal distance of the neighborhood considered. In this matrix, the number of (i, j, k) denotes

the probability of finding a pixel of color i at a distance k away from a pixel of color j. The final feature vector is often obtained by concatenating the rows of the matrix. However, as the size of the color correlogram is usually too large due to its three dimensions, color auto correlogram have been proposed to count only the pair of pixels with the same color i at a distance k, thus allowing to obtain more compact vectors.

– *Color Moments [Stricker & Orengo 1995a]*: Color moments represent the color in a very compact way by a vector containing the mean, variance and skewness (i.e. respectively the moments of order 1, 2 and 3 as shown in (2.1), (2.2) and (2.3)) for each channel of a 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 channel, N is total number of pixels in the image and p_{ij} is the j-th pixel value in channel i. One drawback of color moments is that they are not exclusively representative of what they characterize. Moreover, they are unable to carry the spatial information.

- **Texture features**

  – *Co-occurrence Texture [Tuceryan & Jain 1993]*: Spatial gray level co-occurrence estimates image properties related to second-order statistics. Given a displacement vector $d = (dx, dy)$, the gray level co-occurrence matrix P_d of size $N \times N$ for d is calculated in such a way that the entry (i, j) of P_d is the number of occurrences of the pair of gray levels i and j which are a distance d apart. Here, N denotes the number of gray levels considered. Usually, the matrix P_d is not directly used in

Table 2.1: Some examples of texture features extracted from gray level co-occurrence matrices.

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

an application and a set of more compact features are computed instead from this matrix, such as in Table 2.1. The main problem of gray level co-occurrence matrices is that there is no well established method for selecting the optimal displacement vector d while computing co-occurrence matrices for different values of d is not feasible. 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 1993]*: The basic principle of texture auto-correlation is to compare the original image with a shifted one. Suppose that we consider the displacements according to each axis dx and dy, then the auto-correlation function can be defined as follows:

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

where we consider an image with size $M \times N$ and $I(i, j)$ is the gray level of the pixel in the position (i, j). It measures the coarseness of an image by evaluating the linear spatial relationships between texture primitives. Large primitives give rise to coarse texture (e.g. rock surface) and small primitives give rise to fine texture (e.g. silk surface). If the primitives are large, it decreases slowly while increasing the distance whereas it decreases rapidly if texture consists of small primitives. However, if the

primitives are periodic, then the auto-correlation function increases and decreases periodically with the distance.

– *Local Binary Patterns [Takala et al. 2005]*: Local binary patterns (LBP) are defined as a gray-scale invariant texture measure, derived from a general definition of texture in a local neighborhood. Let g_c be the gray level value of a center pixel (x_c, y_c). We consider a circularly symmetric set of its neighbors $g_p, \quad p = 0, 1, ..., P-1$. Then a P-bit binary number for the center pixel (x_c, y_c) can be computed as follows:

$$(f(g_0 - gc), f(g_1 - gc), ..., f(g_{P-1} - gc)) \tag{2.5}$$

where

$$f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x \leq 0 \end{cases} \tag{2.6}$$

Now, a binomial weight 2^p is assigned to each sign $f(g_p - g_c)$, transforming the differences in a neighborhood into a unique LBP code. The code characterizes the local image texture around (x_c, y_c):

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p f(g_p - g_c) \tag{2.7}$$

After calculating the LBP code for each pixel of an image, we can finally compute a histogram with 2^P bins for the whole image. A typical value of P is 8, meaning that the 8 direct neighbor pixels around the center pixel are considered. Moreover, multiple scales LBP can be obtained by enlarging the radius of the neighbor circle.

– *Gabor [Manjunath & Ma 1996]*: Gabor filter (or Gabor wavelet) is widely adopted to extract texture features from the images for image analysis and has been shown to be very efficient [Manjunath & Ma 1996] [Zhang *et al.* 2000]. Basically, Gabor filters are a group of wavelets, with each wavelet capturing energy at a specific frequency and a specific direction. Expanding a signal using this basis provides a localized frequency

description, therefore capturing local features/energy of the signal. Texture features can then be extracted from this group of energy distributions. The scale (frequency) and orientation tunable property of Gabor filter makes it especially useful for texture analysis. Experimental evidence on human and mammalian vision supports the notion of spatial-frequency (multi-scale) analysis that maximizes the simultaneous localization of energy in both spatial and frequency domains [Daugman 1985].

- **Shape features**

  - *Edge Histogram [Won 2004]*: The edge histogram descriptor describes edge distribution with a histogram based on local edge distribution in an image. It basically represents the distribution of 5 types of edges (namely vertical, horizontal, 45-degree diagonal, 135-degree diagonal and non-directional edges) in each local area called a sub-image, which is defined by dividing the image space into 4×4 nonoverlapping blocks. Thus, the image partition always yields 16 equal-sized sub-images regardless of the size of the original image. In each of them a histogram of edge distribution with 5 bins corresponding to the 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 the same authors to partition the image into 4×1, 1×4 and 2×2 sub-images in order to include the information about edge distribution in different scales.

  - *Histogram of Oriented Gradients [Dalal & Triggs 2005]*: Histogram of oriented gradients is based on evaluating well-normalized local histograms of image gradient orientations in a dense grid. The main idea is that local object appearance and shape can often be characterized rather well by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions. In practice this is implemented by dividing the whole image into small sub-images, for each one accumulating a local 1-D histogram of gradient directions or edge orientations over the pixels of the sub-image. The com-

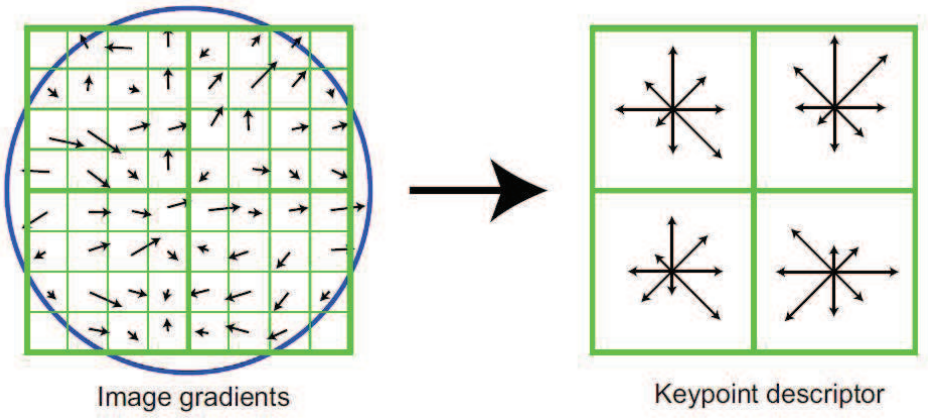

Figure 2.2: The extraction of SIFT feature

> bined histogram entries form the representation. For better invariance to illumination and shadowing, it is also useful to contrast-normalize the local responses before using them. This can be done by accumulating a measure of local histogram "energy" over somewhat larger spatial blocks and using the results to normalize all of the sub-images in the block.

In addition to all these features, we would like to mention an extremely powerful and widely used feature: Scale Invariant Feature Transform (SIFT), proposed by David G. Lowe [Lowe 2004]. SIFT is invariant to image scale and rotation, and is shown to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. Moreover it 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. All these properties ensure its universal success in computer vision and pattern recognition, especially for visual object categorization tasks, such as in the Pascal challenge [Everingham *et al.* 2007].

A typical SIFT descriptor, as presented in [Lowe 2004], is obtained by dividing the local patch into $4 \times 4 = 16$ subregions and then by computing a histogram with 8 orientation bins of local oriented gradients in each of these subregions, thus forming a $16 \times 8 = 128$ dimensional vector. Its extraction principle is illustrated in Figure 2.2.

Although the original SIFT is dedicated to gray-level images, recently, it has been naturally extended to color spaces by running SIFT extraction algorithm in each color channel respectively and then by concatenating the obtained vectors. A series of color SIFT descriptors has been evaluated for object recognition [van de Sande *et al.* 2008] and some of them have been used to construct the winning VOC system in [Everingham *et al.* 2007]. Some examples are listed below:

- **RGB-SIFT**: SIFT descriptors are extracted over all three channels of RGB color space and then concatenate them to form the final representation.

- **HSV-SIFT**: HSV stands for Hue, Saturation, and Value, and is also often called HSB (B for Brightness). It is a cylindrical-coordinate representation of points in a RGB color space and can be transformed from RGB using the following formulae. Let consider $M = \max(R, G, B)$, $m = \min(R, G, B)$ and $C = M - m$, then

$$H = \begin{cases} 0 & \text{if} \quad C = 0 \\ (60^\circ \times \dfrac{G-B}{C} + 360^\circ) \mod 360^\circ & \text{if} \quad M = R \\ 60^\circ \times \dfrac{B-R}{C} + 120^\circ & \text{if} \quad M = G \\ 60^\circ \times \dfrac{R-G}{C} + 240^\circ & \text{if} \quad M = B \end{cases} \tag{2.8}$$

$$S = \begin{cases} 0 & \text{if} \quad M = 0 \\ 1 - \dfrac{m}{M} & \text{otherwise} \end{cases} \tag{2.9}$$

$$V = M \tag{2.10}$$

  The same feature extraction technique as RGB-SIFT is applied on HSV color space to generate HSV-SIFT.

- **Opponent SIFT (O-SIFT)**: O-SIFT describes all the channels using SIFT descriptors in the opponent color space, which is transformed from RGB as

$$O_1 = \frac{R-G}{\sqrt{2}} \tag{2.11}$$

$$O_2 = \frac{R + G - 2B}{\sqrt{6}} \tag{2.12}$$

$$O_3 = \frac{R + G + B}{\sqrt{3}} \tag{2.13}$$

The information in the O_3 channel is equal to the intensity information, while the other channels describe the color information in the image.

- **C-SIFT**: C-SIFT can be seen as a normalized version of O-SIFT, which works in the normalized opponent color space $(\frac{O_1}{O_3}, \frac{O_2}{O_3}, O_3)$, eliminating the remaining intensity information from O_1 and O_2 channel thus being invariant to intensity changes.

2.2.2 Classification strategies

2.2.2.1 Global appearance and sliding window

The earliest works concerning visual object categorization have mainly focused on the global description of images by using color or texture histogram [Niblack *et al.* 1993] [Schiele & Crowley 2000] for example, which is generally based on the statistical analysis of the whole image (or image regions) pixel by pixel. This representation can cooperate with the so-called "sliding window" technique [Papageorgiou & Poggio 2000] [Viola & Jones 2001] to perform generic object categorization. As the principle of this technique is to slide a window across the image at different scales and to classify each such sub-window as containing the target object or not, its advantages are that it can find the localization of the object at the same time and is easy to implement because of its simple detection protocol. However, it often fails to detect non-rigid deformable objects or the objects that can not be shaped by a rectangular. In practice, it usually needs a large dataset of cropped images for training and thus requires a high computational cost. All these limitations have encouraged researchers to pay more attention to the part-based methods.

2.2.2.2 Part-based models

One theory of biological vision [Palmer 1977] [Logothetis & Sheinberg. 1996] gives a theoretical support for such part-based methods [Agarwal & Roth 2002] [Mohan *et al.* 2001] [Weber *et al.* 2000] [Felzenszwalb & Huttenlocher 2005] [Ullman *et al.* 2001]. According to this theory, the representation used by humans for identifying an object consists of the parts that constitute the object, together with structural relations over these parts that define the global geometry of the object.

In this category of methods, images are represented by a set of object parts and their spatial connectivity in the image. [Mohan *et al.* 2001] considers distinctive higher-level parts that are rich in information content for a specific class of interest, namely person. It uses separate classifiers to detect different parts of person in the image, such as heads, arms and legs, and then train a final classifier to give the final decision. But the fact that it requires the object parts to be manually defined and separated for training the individual part classifiers makes it difficult to be used with other object classes. So [Weber *et al.* 2000] tries to automatically identify distinctive parts in the training set by applying a clustering algorithm to patterns selected by an interest operator and the objects are represented as flexible constellations of rigid parts. Then a generative probabilistic model is learned over these parts to get the final result. [Agarwal & Roth 2002] follows globally the same approach as [Weber *et al.* 2000], but in this case, a classifier is learned over parts instead of using a probabilistic model. Other approaches, including [Ullman *et al.* 2001] in which objects within a class are represented in terms of common image fragments and [Felzenszwalb & Huttenlocher 2005] which represent an object by a collection of parts arranged in a deformable configuration (spring-like connections between pairs parts) using the pictorial structure, have also shown to be effective.

However, all those methods are not designed to handle large viewpoint variations or severe object deformations. Moreover, learning and inference problems for spatial relations remain very complex and computationally expensive.

2.2.2.3 Bag of features models

Recently, most works in the literature make use of a "bag of features" kind of approach [Dance *et al.* 2004] [Rothganger *et al.* 2006] and has shown its effectiveness, obtaining the best performance in Pascal VOC contest [Everingham *et al.* 2007] [Everingham *et al.* 2008]. Its general principle is to adapt the "bag of words" representation for text categorization [Salton & McGill 1983] to VOC problem and has first been applied on images on texture recognition [Leung & Malik 2001]. In fact, this kind of models can be seen in some sense as a special part-based model, without considering the spatial connectivity between parts.

These methods view images as an orderless distribution of local image features, typically using the popular SIFT features [Lowe 2004] extracted from salient image regions, called "interest points" [Lowe 2004] [Mikolajczyk & Schmid 2004] or more simply from points extracted using a grid [Fei-Fei & Perona 2005]. The set of these local features is then characterized by a histogram of "visual keywords" from a visual vocabulary which is learned from the training set by a hard assignment (quantization) or a soft assignment through Gaussian Mixture Model (GMM). These distributions can thus be compared to estimate the similarities between images and categorized through a machine learning process, for instance SVM.

Although the "bag of features" approach has achieved the best performance in the last Pascal VOC contests, the overall performance, with an average precision around 60% over 20 classes achieved by the best classifier in [Everingham *et al.* 2007], is still far from real application-oriented requirements. Moreover, the size of visual vocabulary which is the basis of this approach is hard to be fixed as there are no evident similar concepts in images as compared to a textual document.

2.2.3 Generative and discriminative methods

There exist generally two main kinds of approaches in the literature for making the final decision of classification: generative method and discriminative method. Suppose x being the set of data representing an image to be classified and $C_m, \quad m =$

$1, ..., M$ being a set of class labels in consideration, the generative model will estimate the posterior probability $p(C_m|x)$ in the probabilistic framework, according to which x will be classified into the target class (for instance, if we wish to minimize the number of misclassifications, we assign x to the class having the largest posterior probability). In the case of discriminative models, the objective is to learn the precise boundaries between the different classes of samples in a multi-dimensional space (often the feature space) so that the classification can be performed by considering the position of the image projection in this space.

2.2.3.1 Generative method

Using Bayes theorem, the posterior probability $p(C_m|x)$ can be expressed in the following form:

$$p(C_m|x) = \frac{p(x|C_m)p(C_m)}{p(x)} \tag{2.14}$$

where $p(C_m)$ is the prior probability of the class C_m and $p(x|C_m)$ is probability density of class C_m, called likelihood. $p(x)$ is the probability density over all the classes. As it is 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.

A typical generative method relies on a Gaussian Mixture Model (GMM) [Bishop 2007] to model the distribution of the training samples. The set of a GMM parameters can be efficiently learned by using Expectation Maximization algorithm (EM) [Dellaert 2002]. Recall the a GMM distribution in the form:

$$\begin{aligned} p(x) &= \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \\ &= \sum_{k=1}^{K} \pi_k \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma_k|^{\frac{1}{2}}} exp[-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)] \end{aligned} \tag{2.15}$$

where μ_k and Σ_k are respectively mean and covariance of the k-th gaussian (k-th component of a GMM which contains a total of K gaussians) and D is the

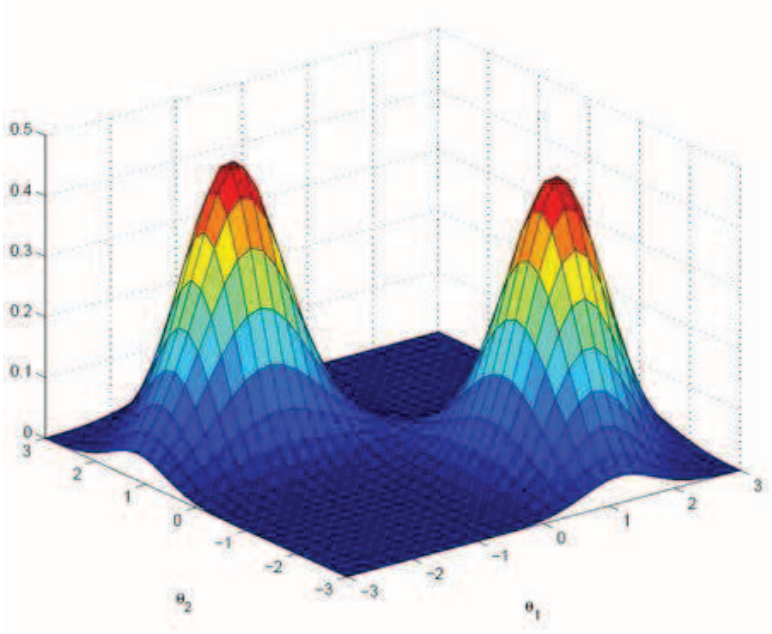

Figure 2.3: A graphical example of a 2-components GMM

dimensionality of data. The parameters π_k are called mixing coefficients and must satisfy

$$0 \leq \pi_k \leq 1 \qquad \text{together with} \quad \sum_{k=1}^{K} \pi_k = 1 \tag{2.16}$$

Figure 2.3 shows graphically an example of a 2-components GMM.

If we consider a GMM for modeling the specific class C_m, then the log of the likelihood function is given by:

$$\begin{aligned} ln(p(x|C_m)) = ln(p(x|\mu, \Sigma, \pi)) &= ln \prod_{n=1}^{N} \{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)\} \\ &= \sum_{n=1}^{N} ln\{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)\} \end{aligned} \tag{2.17}$$

where N is the number of feature vectors in x. Then, we can employ the EM algorithm to maximize the likelihood function for class C_m with respect to the parameters of the GMM, according to the following steps (see details in [Bishop 2007]):

1. Initialize all the parameters and evaluate the initial value of the log likelihood.

2. **E step**. Evaluate the responsibilities using the current parameter values:

$$\gamma_n^k = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_k \mathcal{N}(x_n|\mu_j, \Sigma_j)} \tag{2.18}$$

3. **M step**. Re-estimate the parameters using the current responsibilities

$$\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k x_n \tag{2.19}$$

$$\Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k (x_n - \mu_k^{new})(x_n - \mu_k^{new})^T \tag{2.20}$$

$$\pi_k^{new} = \frac{N_k}{N} \tag{2.21}$$

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

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

After having obtained the optimized GMMs for all the classes, a new sample x^{new} is assigned to the class having the largest value of the log likelihood function given this x^{new}.

The generative method offers the advantage to easily handle 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. However, the discriminative method has been shown to be more efficient for the classification problems, especially with a relatively large number of training samples [Bouchard & Triggs 2004].

2.2.3.2 Discriminative method

Discriminative method directly estimates the posterior probabilities without attempting to model the underlying probability distributions. Many discriminative classifiers are reported in the literature. Some of the most representative ones are presented below.

Support vector machines Among all the kernel-based discriminative classifiers, Support Vector Machines (SVM) proposed by Vanpik [Cortes & Vapnik 1995] based on his statistical learning theory [Vapnik 1995] is the most famous and popular one [Cortes & Vapnik 2005] [Cristianini & Shawe-Taylor 2000] [Ruan *et al.* 2010]. Let consider a set of N labeled training samples $(x_i, y_i) \quad i = 1, ...N$ where $x_i \in R^D$ is the feature vector representing an image with D dimension while $y_i \in \{1, -1\}$ is the image label. SVM constructs a hyperplane with maximal margin in a high or infinite dimensional space to linearly separate these samples into 2 predefined categories respectively through solving the following optimization problem:

$$\begin{aligned} &\min_{w,b,\xi} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} \xi_i \\ &subject \quad to \quad y_i(w^T \phi(x_i) - b) \geq 1 - \xi_i, \qquad i = 1, ..., N \\ &\xi_i \geq 0. \end{aligned} \tag{2.22}$$

Here training samples x_i are mapped into a higher or infinite dimensional space by the function ϕ, in which the separation of these training samples is presumably linear and much easier than in the original finite dimensional space. Indeed, in most of situations, classes are not linearly separable in the original space. C is the penalty parameter of the error term which controls the penalty level of the misclassified samples. Finally we can get the decision function in the form:

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i K(x, x_i) + b \tag{2.23}$$

where α_i and b are obtained parameters in the solving procedure, x is a new sample to be classified. Here we should especially mention the kernel function K as in (2.24), which is extremely important to achieve a good performance using SVM for classification. The choice of this kernel function and the tuning of its parameters will directly impact the final result. We will introduce some basic and commonly used kernel functions later.

$$K(x, x_i) = \phi(x)^T \phi(x_i) \tag{2.24}$$

The original SVM is binary classifier, whereas many image classification problems have multiple classes, much more than 2. Two common strategies are designed to deal with this situation: one-against-all and one against-one. The former strategy will construct one SVM binary classifier for each class taking the samples in this considered class as the positive samples and all the other as the negative ones. However, the latter strategy will construct one SVM binary classifier for each pair of classes. Classification is done in a max-wins voting way, in which every classifier assigns the sample to one of the two classes, then the vote for the assigned class is increased by one vote, and finally the class with most votes determines the sample classification, such as C-SVC in LIBSVM package [Chang & Lin 2001].

Multiple kernel learning SVM uses only one kernel for solving learning problems like classification or regression and thus is short of some flexibility. Therefore, using multiple kernels instead of a single one is now largely researched and some works have already demonstrated its ability of improving classification performance [Lanckriet *et al.* 2004]. The combination of multiple kernels is defined as follows:

$$\begin{aligned} K(x, x_i) &= \sum_{m=1}^{M} \beta_m K_m(x, x_i) \\ with \quad \beta_m \geq 0, &\quad \sum_{m=1}^{M} \beta_m = 1 \end{aligned} \tag{2.25}$$

where M is the total number of kernels, β_m is kernel weight which is optimized during training. Each basis kernel K_m can either be different kernels with different parameter configurations or use different subsets of the extracted features. So MKL can also be interpreted as a fusion technique in some sense. The final decision function of MKL can be in the following form, very similar to the one of SVM except the combined kernels:

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i \sum_{m=1}^{M} \beta_m K_m(x, x_i) + b \tag{2.26}$$

where α_i and b are the obtained parameters after training, the same as in SVM problem. Here α_i and β_m can be learned in a joint optimization problem as in [Bach *et al.* 2004] [Rakotomamonjy *et al.* 2008].

A natural extension of the precedent Simple MKL, called Group-Sensitive MKL (GS-MKL) by the authors, is presented in [Yang *et al.* 2009a]. An intermediate notion "group" between object categories and individual images has been introduced to MKL framework to seek a trade-off between capturing the diversity and keeping the invariance for each class in training classifiers. In GS-MKL, the kernel weights β_m not only depend on the corresponding kernel functions, but also on the groups that two compared images belong to. Thus, the combined kernel in (2.25) and the decision function in (2.26) are respectively rewritten as

$$K(x, x_i) = \sum_{m=1}^{M} \beta_m^{c(x)} \beta_m^{c(x_i)} K_m(x, x_i) \tag{2.27}$$

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i \sum_{m=1}^{M} \beta_m^{c(x)} \beta_m^{c(x_i)} K_m(x, x_i) + b \tag{2.28}$$

where $c(x)$ and $c(x_i)$ are the group ids of image x and x_i respectively. Although GS-MKL is shown to be very efficient for image classification in the experiments on several datasets, the optimal way to get group ids remains debatable. Actually, the authors use some clustering methods, namely K-means [Gersho & Gray 1991] and probabilistic Latent Semantic Analysis (pLSA) [Hofmann 1998], to get a set of groups whose number is manually defined. However, there is no obvious proof which can help to choose the optimal number of groups and the corresponding clustering method.

Kernel functions The discriminative power of SVM depends for a large part on the kernel selection. Thus, the choice for an appropriate kernel is of first importance. Unfortunately, to the best of our knowledge, until now, kernel selection for a certain application is generally done empirically and experimentally, or in some case accomplished by cross-validation. There exist many kernel functions in the literature. The most representative ones are the followings:

- Linear: $K(x, x_i) = x^T x_i$
- Polynomial: $K(x, x_i) = (\gamma x^T x_i + r)^p, \quad \gamma > 0$
- Radial Basis Function (RBF): $K(x, x_i) = exp(-\gamma \|x - x_i\|^2), \quad \gamma > 0$
- Sigmoid: $K(x, x_i) = tanh(\gamma x^T x_i + r)$
- chi-square: $K(x, x_i) = 1 - \sum_{j=1}^{n} \frac{(x^j - x_i^j)^2}{\frac{1}{2}(x^j + x_i^j)}$
- Pyramid match [Grauman & Darrell 2005] : 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. Suppose x and x_i have n dimensions and H_x^l and $H_{x_i}^l$ denote the histogram of x and x_i at the resolution l in which we have 2^l bins along each dimension, $l = 0, ..., L$, so that $H_x^l(j)$ and $H_{x_i}^l(j)$ are the number of points from x and x_i that fall into the j-th bin of the grid. Then the number of matches at level l is given by the histogram intersection function:

$$I(H_x^l, H_{x_i}^l) = \sum_{j=1}^{2^{nl}} \min(H_x^l(j), H_{x_i}^l(j)) \tag{2.29}$$

  If we abbreviate $I(H_x^l, H_{x_i}^l)$ to I^l, finally we get the pyramid match kernel:

$$\begin{aligned} K^L(x, x_i) &= I^L + \sum_{l=0}^{L-1} \frac{1}{2^{L-l}} (I^l - I^{l+1}) \\ &= \frac{1}{2^L} I^0 + \sum_{l=1}^{L} \frac{1}{2^{L-l+1}} I^l \end{aligned} \tag{2.30}$$

Here, the above γ, r, p and L are all kernel parameters.

Other typical discriminative classifiers We will briefly present here several other typical discriminative classifiers, some of them being used later in our experimentations.

- 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: 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 neighbor [Shakhnarovich *et al.* 2005]: 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: First introduced by Freund and Schapire [Freund & Schapire 1997], it calls a weak classifier repeatedly in a series of rounds $t = 1, ..., T$. For each round, the 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.

2.2.4 Fusion strategies

Fusion strategy is usually used in multimedia data analysis [Ayache *et al.* 2007a]. Indeed, generally three modalities have to be handled in videos, namely the auditory, the textual, and the visual modality. Thus, a fusion step is necessary to combine the results of the analysis of these modalities considered independently in a first step [Snoek *et al.* 2005]. The same idea can be employed in visual object categorization, since, in order to extract a visual information as exhaustive as possible, different types of features from the same image can be computed to form several information streams. These streams need to be fused in order to elaborate a single decision from

several sources of information. This fusion of different types of features can follow several strategies:

- Early fusion: An early fusion is obtained when grouping all the features together in order to build a single feature vector that will feed the classifier.
- Late fusion: A late fusion makes use of "channels" with a separate classifier for each kind of features, the outputs of these classifiers being merged later [Snoek *et al.* 2005] in a process similar to boosting [Freund & Schapire 1999].

Between these two strategies, numerous intermediate strategies can be conceivable which consist in generating intermediate classes from different sources and to take a final decision based on these intermediate classes [Ayache *et al.* 2007b]. If we take our 3 types of feature which are used in our experimentation as an example, namely SIFT, Region based Color Moments (RCM) and Region based Histogram of Segments (RHS), the scheme of early fusion and late fusion can be illustrated as in Figure 2.4

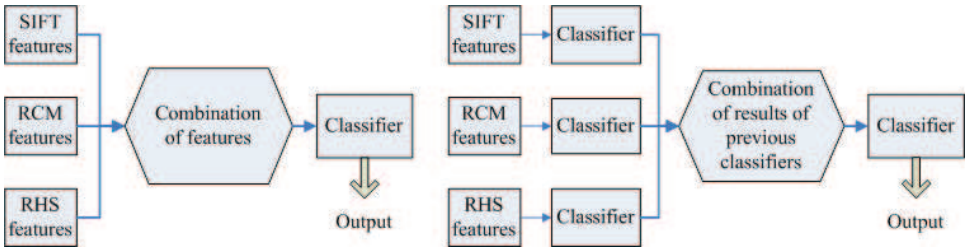

Figure 2.4: General scheme for early fusion (left) and late fusion (right)

2.3 Feature selection

With the increasing trend of high dimensional data processing, feature selection becomes more and more important and even essential in most of pattern recognition and machine learning problems. Indeed, when a pattern classification problem has to be solved, the common approach for the feature extraction is to compute a wide variety of features that will carry as much as possible different information to perform the classification of samples. Thus, numerous features are used whereas,

generally, only a few of them are relevant for the considered classification task. However, algorithms in these domains are often known to suffer from the so-called "curse of dimensionality" [Bellman 1961] if too many input features extracted from the samples are directly fed into the classifier without selection, especially when these features are redundant and irrelevant to the considered problem. Concretely, including these irrelevant features in the feature set used to represent the samples to classify may lead to a slower execution of the classifier, less understandable results, and much reduced accuracy [Hall & Smith 1997]. In this context, the aim of feature selection is three-fold: improving the prediction performance of the predictors, providing faster and more cost-effective predictors, and gaining a deeper insight into the underlying processes that generated the data.

With the objective of selecting the most discriminant features to improve the classification accuracy with a low complexity, we present in this section a novel embedded feature selection approach, called ESFS, based on the well-known search method SFS [Whitney 1971]. It relies on the simple principle to add incrementally most relevant features and merge them in an embedded way thanks to the concept of combined mass functions from the evidence theory which also offers the benefit of obtaining a computational cost much lower than the one of original SFS.

2.3.1 Literature review

There exist considerable works in the literature dealing with feature selection. Interesting overviews include [Kohavi & John 1997] [Guyon & Elisseff 2003] [Combarro *et al.* 2005] [Liu & Yu 2005]. In recent studies, *evaluation criterion* and *search strategy* are the two main aspects attracting attention and we will also follow these two aspects to begin the presentation of related works.

2.3.1.1 Evaluation criterion

Indeed, the notion of "optimal" subset is always related to a certain evaluation criterion and, generally, different evaluation criteria would give different "optimal" subsets. Typically, the evaluation criterion is used to evaluate the efficiency of

feature subsets selected from the original set within a particular feature selection process. In other words, it is the indication of the discrimination ability of a feature subset for classifying a sample into the corresponding class.

According to the evaluation criterion used and the dependence to the classifier, feature selection methods can be categorized into three main categories: filter approaches, wrapper approaches and embedded approaches [Kojadinovic & Wottka 2000].

Filter methods include Fisher filter [Narendra & Fukunaga 1977], Relief method [Arauzo-Azofra *et al.* 2004], Focus algorithm [Almuallim & Dietterich 1991], Orthogonal Forward Selection [Mao 2004], etc. They generally evaluate the statistical performance of the features over the data without considering the proper classifiers and use their intrinsic properties as the evaluation criterion, such as class separability measures. The irrelevant features are filtered out before the classification process [Hall & Smith 1997]. Their main advantage is their low computational complexity which makes them very fast. Their main drawback is that they are not optimized to be used with a particular classifier as they are completely independent of the classification stage.

Wrapper methods, on the contrary, evaluate feature subsets with the classification algorithm in order to measure their efficiency according to the classification results (the correct classification rate is usually used as the evaluation criterion) [Kohavi & John 1997]. Thus, feature subsets are generated thanks to some search strategy, and the feature subset which leads to the best correct classification rate is kept. Among algorithms widely used, one can mention Genetic Algorithm (GA) [Yang & Honavar 1998] [Huang *et al.* 2007], Sequential Forward Selection (SFS) [Whitney 1971], Plus l - Take away r algorithm [Stearns 1976], Sequential Floating Forward Selection (SFFS) [Pudil *et al.* 1994b] and Oscillating Selection (OS) [Somol & Pudil 2000]. The computational complexity is higher than the one of filter methods but selected subsets are generally more efficient, even if they remain sub-optimal [Spence & Sajda 1998].

In *embedded methods*, similarly to wrapper methods, the feature selection is linked to the classification stage and uses the classification result as the evaluation

criterion. This link is in this case much stronger as the feature selection in embedded methods is included into the classifier construction. Such methods include recursive partitioning methods for decision trees such as ID3 [Quinlan 1986], C4.5 [Quinlan 1993] [Quinlan 1996] and CART [Breiman *et al.* 1984], or the recently proposed Recursive Feature Elimination (RFE) approach, which is based on the support vector machines (SVM) theory and has shown its good performance for the gene selection [Guyon *et al.* 2002] [Rakotomamonjy 2003]. Embedded methods offer the same advantages as wrapper methods concerning the interaction between the feature selection and the classification. Moreover, they present a better computational complexity since the selection of features is directly included into the classifier construction during the training process.

2.3.1.2 Search strategy

As mentioned previously, another important aspect concerning feature selection is the search strategy, which aims at finding the best subset based on a given evaluation criterion. *Optimal search methods* and *Sub-optimal search methods* are generally considered as the two main strategies for this purpose [Pudil *et al.* 2002], and will be detailed below.

Exhaustive search approach is intuitively the first choice when one hopes to find an optimal subset. All the possible combinations of all candidate features are thus evaluated. However, the combinatorial property of such methods requires a large amount of computational effort, especially for large scale problems, which makes them unusable in most of practical applications. Some other accelerated search approaches, such as the Branch and Bound (B&B) algorithm [Narendra & Fukunaga 1977], also guarantee to find the optimal subset without exhaustive search. But the main drawback of B&B is that it requires the evaluation criterion used in the procedure to be monotonic. Indeed, the evaluation criterion value should not decrease when a new feature is added into the current subset. Obviously, this requirement has limited its range of applications since most of evaluation criteria used for feature selection could not satisfy the monotonicity condition. Moreover, even if Monte Carlo methods based on simulated annealing [Doak 1992] and

some genetic algorithms [Yang & Honavar 1998] can also reach global optimal solution, they are also computationally impractical if the number of potential features is large.

Since it exists severe constraints on the computation for optimal search methods, the mainstream of research on feature selection has thus been oriented to the numerous sub-optimal search methods, among which the Sequential Feature Selection is considered to be the basic one. SFS (or correspondingly Sequential Backward Selection, SBS) starts with the empty feature set (full feature set) and incrementally add (delete) the most effective (irrelevant) feature at each stage until reaching the desired number of features. However, once a feature is selected in SFS (removed in SBS), it can not be deleted (re-selected) in the following stages. Thus, these methods suffer from the so-called "nesting problem" and may fail in some situations (fall into local minima). In order to overcome this drawback, the plus l - take away r algorithm [Stearns 1976] and SFFS [Pudil *et al.* 1994b] [Pudil *et al.* 1994a] have been proposed by combining SFS and SBS together.

The plus l - take away r method consists in applying SFS l times followed by r steps of SBS with this fixed cycle of forward and backward selection repeated until the required number of features is reached. Consequently, SFS and SBS can be seen as the special plus l - take away r method in which (l, r) equals $(1, 0)$ and $(0, 1)$ respectively. But here a new question arises: how can (l, r) be set to the appropriate values? Actually, there does not exist an explicit way of predicting the best values of l and r to obtain good enough solutions with a moderate amount of computation. This has motivated researchers to consider the conditional inclusion and exclusion of features controlled by the value of the evaluation criterion itself which is key idea of SFFS. It consists in applying after each forward step several backward steps, the number of which is automatically determined according to the rule that the resulting subsets are better than the previously evaluated ones at that level. As a result, there is no parameter tuning needed for SFFS and it can make more than one sweep to obtain good performance compared to plus l - take away r algorithm. Jain and Zongker's study [Jain & Zongker 1997] has demonstrated the effectiveness of SFFS through a comparison with other search strategies of feature selection. Unlike the

two methods presented above, OS [Somol & Pudil 2000] directly works on the subset with a desired size and repeatedly modifies it by applying oscillation cycle composed of a down-swing for removing worst features followed by a up-swing for adding best features. Depending on the way the initial subset is built, OS may be looked upon as a universal tuning mechanism to improve solutions obtained beforehand by any other methods, or can be treated as a traditional feature selection method if a random initialization is used. Furthermore, OS algorithm can be stop after running for a predefined time and still allows to obtain a reasonable solution. Thanks to this property, it can be used in both of the quality first and speed first applications.

2.3.2 ESFS: an Embedded Sequential Forward Selection

Since an exhaustive search for the best subset of features, leading to explore a space of 2^n subsets (n being the number of candidate features), is not feasible in most of practical applications, we have turned to a heuristic approach for the feature selection. In this section, we propose a new embedded feature selection method called ESFS [Fu *et al.* 2009a], inspired from the wrapper method SFS since it relies on the simple principle to add incrementally most relevant features. Moreover, we have provided here two innovations compared to the classical classifier dependent sub-optimal selection method SFS. Firstly, the range of subsets to be evaluated in the forward process is extended to multiple subsets for each size in order to improve the search quality. The computational cost increase is compensated by considering at each step only a subset composed of the best individual features. Secondly, we make use of the concept of mass function from the evidence theory which allows to elegantly merge feature information and process classification in an embedded way, leading to a lower computational cost than original SFS.

In our feature selection scheme, the concept of "belief mass" from the evidence theory is introduced into the processing of features and plays an important role. In order to better understand this notion and how it is integrated into our approach, we would like first of all to present a brief overview of the evidence theory before going deeper into ESFS scheme.

2.3.2.1 Overview of the evidence theory

The evidence theory introduced by Dempster [Dempster 1968] and completed by Shafer [Shafer 1976] offers a framework allowing the reasoning on knowledge that can be uncertain, incomplete, ambiguous and leading to conflicts. This theory relies on belief mass functions which are a generalization of probability and possibility measures.

To do this, a set of definition Ω is defined as a set of n hypotheses H_i that are mutually exclusive:

$$\Omega = \{H_1, H_2, \ldots, H_n\} \tag{2.31}$$

The reasoning does not only concern hypotheses of Ω but is much richer as it allows to consider all possible combinations of the hypotheses in Ω which are contained in the set of discernment 2^Ω:

$$\begin{aligned} 2^\Omega &= \{A/A \subseteq \Omega\} \\ &= \{\emptyset, \{H_1\}, \{H_2\}, \ldots, \{H_n\}, \{H_1, H_2\}, \ldots, \Omega\} \end{aligned}$$

The confidence, or belief, we can have in a proposition $A \subseteq \Omega$ considering a given source of information is provided by the mass function associated with this source of information. A mass function is defined as follows:

$$m^\Omega : 2^\Omega \rightarrow [0, 1] \tag{2.32}$$

$$A \rightarrow m^\Omega(A) \tag{2.33}$$

where:

$$m^\Omega(\emptyset) = 0 \tag{2.34}$$

$$\sum_{A \subseteq \Omega} m^\Omega(A) = 1 \tag{2.35}$$

Focal elements are propositions A such that $m^\Omega(A) > 0$. Thus, $m^\Omega(A)$ expresses

the confidence we have in proposition A according to the source of information modelled by m^{Ω}. If $m^{\Omega}(\Omega) = 1$, then the source is completely uncertain whereas if $m^{\Omega}(H_i) = 1$, then the source is perfect for hypothesis H_i.

One of the most interesting feature of the evidence theory is its ability to combine different mass functions from several sources of information. The most commonly used fusion operator is a conjunctive orthogonal sum called TBM (Transferable Belief Model). Let m^{Ω}_{S1} and m^{Ω}_{S1} be two mass functions from two independent sources of information $S1$ and $S2$. Then, the TBM combined mass function $m^{\Omega}_{S1\cap S2}$ is given by:

$$m^{\Omega}_{S1\cap S2}(A) = \sum_{B\cap C=A} m^{\Omega}_{S1}(B).m^{\Omega}_{S2}(C) \tag{2.36}$$

where A, B and C are subsets of Ω.

A conflict can appear if $m^{\Omega}_{S1\cap S2}(\emptyset) \neq 0$. This indicates that the two sources of information $S1$ and $S2$ lead to contradictory propositions. Thus, checking the conflict value allows to determine if measures are reliable and coherent.

The previous combination rule does not make use of any possible conflict. So other rules has been defined to overcome this drawback and we would like to mention the Dempster's combination rule and the Yager's combination rule here for examples. Let m^{Ω}_{S1} and m^{Ω}_{S1} be two mass functions from two independent sources of information $S1$ and $S2$. Then, the Dempster's combined mass function $m^{\Omega}_{S1\oplus S2}$ is computed for a proposition $A \subseteq \Omega \setminus \emptyset$ as follows:

$$m^{\Omega}_{S1\oplus S2}(A) = \frac{\sum_{B\cap C=A} m^{\Omega}_{S1}(B).m^{\Omega}_{S2}(C)}{1 - m^{\Omega}_{S1\oplus S2}(\emptyset)} \tag{2.37}$$

where

$$m^{\Omega}_{S1\oplus S2}(\emptyset) = \sum_{B\cap C=\emptyset} m^{\Omega}_{S1}(B).m^{\Omega}_{S2}(C) \tag{2.38}$$

With this rule, the conflict $m^{\Omega}_{S1\oplus S2}(\emptyset)$ is used to weigh the masses of the mass function after combination. However, the Yager's rule treats the conflict in another

way and reassigns it to the whole set of definition Ω whose formula is as follows:

$$\forall A \subseteq \Omega \setminus \{\emptyset, \Omega\}, \quad m^{\Omega}_{S1,S2}(A) = \sum_{B \cap C = A} m^{\Omega}_{S1}(B).m^{\Omega}_{S2}(C) \tag{2.39}$$

$$m^{\Omega}_{S1,S2}(\emptyset) = 0 \tag{2.40}$$

$$m^{\Omega}_{S1,S2}(\Omega) = \sum_{B \cap C = \Omega} m^{\Omega}_{S1}(B).m^{\Omega}_{S2}(C) + \sum_{B \cap C = \emptyset} m^{\Omega}_{S1}(B).m^{\Omega}_{S2}(C) \tag{2.41}$$

Once mass functions from the different sources of information at our disposal have been combined into a single mass function, using one of the previous rules, a final decision should be taken regarding the choice of a proposition. To do this, several decision measures can be used based on the evidence mass function, the belief, the plausibility or the pignistic probability. In each case, the proposition having the highest value will be chosen. The belief (credibility) of a proposition A is given by:

$$\forall A \subseteq \Omega, bel(A) = \sum_{\emptyset \neq B \subseteq A} m^{\Omega}(B) \tag{2.42}$$

The plausibility of a proposition A is given by:

$$\forall A \subseteq \Omega, pl(A) = \sum_{A \cap B \neq \emptyset} m^{\Omega}(B) \tag{2.43}$$

The plausibility verifies $pl(A) = 1 - bel(\bar{A})$ and $bel(A) \leq P(A) \leq pl(A)$ where $P(A)$ is the probability of the proposition A.

At last, the pignistic probability is given by:

$$BetP(A) = \sum_{B \subseteq \Omega} \frac{\|A \cap B\|}{\|B\|} m^{*\Omega}(B) \tag{2.44}$$

where $\|A\|$ is the cardinal of A and $m^{*\Omega}(A) = \frac{m^{\Omega}(A)}{1 - m^{\Omega}(\emptyset)}$.

This definition of mass functions from the evidence is used in our model in order to represent the source of information given by each feature, to combine them easily and to provide a decision values which allows to use them as embedded classifiers.

2.3.2.2 ESFS scheme

A heuristic feature selection algorithm can be characterized by its stance on four basic issues that determine the nature of the heuristic search process. First, one must determine the starting point in the space of feature subsets, which influences the direction of search and operators used to generate successor states. The second decision involves the organization of the search. As an exhaustive search in a space of 2^n feature subsets is impractical, one needs to rely on a more realistic approach such as greedy methods to traverse the space. At each point of the search, one considers local changes to the current state of the features, selects one and iterates. The third issue concerns the strategy used to evaluate alternative subsets of features. Finally, one must decide on some criterion for halting the search. In the following, we bring our answers to the previous four questions.

As we have mentioned previously, the SFS algorithm starts with an empty subset of features. The new subset S_k with k features is obtained by adding a single new feature to the subset S_{k-1} which performs the best among the subsets with $k-1$ features. The correct classification rate achieved by the selected feature subset is used as the selection criterion. In the original algorithm of SFS, there are totally $n(n+1)/2$ subsets which need to be evaluated and unfortunately the optimal subset may not be reached.

In order to avoid departure too far from the optimal performance, we proposed an improvement of the original SFS method by extending the subsets to be evaluated. At each step of forward selection, instead of keeping only one subset for each size of subsets, several good quality subsets (performance above a given threshold) are considered to be evaluated during the next step. Since remaining multiple subsets at each step may lead to heavy computational burden, only the features selected during the first step (subsets with a single feature), thus having the best abilities to discriminate among classes that occur in the training data, are used for the evaluation in posterior steps. As the features are added to the potential subsets one by one in the SFS process, the forward process of creating a feature subset with size k can be seen as a combination between two elements: a subset with size $k-1$ and

a single feature.

A wrapper feature selection scheme such as SFS relies on a classifier in order to evaluate the improvement of classification accuracy as feature selection criterion. This classifier needs to be trained and then tested at each step for each possible feature subset. We propose to improve this process and make it less time consuming by embedding the feature selection into the classifier construction. This is realized by representing each feature thanks to a mass function (introduced in section 2.3.2.1) obtained from its distribution for each class in the training data. This representation allows not only to easily combine features (and thus to built feature subsets at each iteration of the search process) thanks to the fusion of their corresponding mass function, but also to make use of the combined mass function as a decision value for classification. Thus, each subset can be considered as a new feature resulting from the combination of a feature obtained from the previous step with a single feature from the original selected feature set.

This procedure is detailed in the following.

Feature selection procedure The feature selection procedure by ESFS consists of four stages that are detailed below.

Stage 1: Computation of the belief masses for the single features.

Since the features may have very different domains of variation, they are first of all normalized into $[0, 1]$. Let F^n represents the n-th feature with $n \in 1, ..., N$ where N is the total number of features. Then, the normalization is performed according to following equation:

$$f^n = \frac{f_0^n - min(F^n)}{max(F^n) - min(F^n)} \tag{2.45}$$

where f_0^n is the original value of the feature F^n, whereas f^n is its normalized value.

The belief mass which is associated to a source of information and represents the belief we have in a statement to be true can be obtained by different ways. In this paper, we have considered each single feature as a source of information, and the corresponding mass function is computed from their PDF (Probability Density Functions). To do so, the distribution of each feature over all classes is calculated

from the training data. Their PDF is then obtained by approximating the distribution thanks to a polynomial interpolation.

Taking the case of a 2-class classifier as an example, the classes are defined as subset A and its complement subset A^C in Ω. First, the probability densities of the features in each of the 2 subsets are estimated from the training samples. We define the probability density of the feature F^n in subset A as $P^n(A, f^n)$ and the probability density in subset A^C as $P^n(A^C, f^n)$. According to the probability densities, the masses of feature F^n on these two subsets can be defined as

$$m^n(A, f^n) = \frac{P^n(A, f^n)}{P^n(A, f^n) + P^n(A^C, f^n)} \tag{2.46}$$

$$m^n(A^C, f^n) = \frac{P^n(A^C, f^n)}{P^n(A, f^n) + P^n(A^C, f^n)} \tag{2.47}$$

where at any possible value of the n-th feature f^n, $m^n(A, f^n) + m^n(A^C, f^n) = 1$.

In the case of M classes, the classes are defined as $A_1, A_2, ..., A_M$. The masses of feature F^n of the i-th class A_i can be obtained as

$$m^n(A_i, f^n) = \frac{P^n(A_i, f^n)}{\sum_{j=1}^{M} P^n(A_j, f^n)} \tag{2.48}$$

which satisfies

$$\sum_{j=1}^{M} m^n(A_j, f^n) = 1 \tag{2.49}$$

For convenience, we will simplify $m^n(A_i, f^n)$ as $m^n(A_i)$.

Stage 2: Evaluation of the single features and selection of the initial set of discriminative features.

Once the belief masses for the single features among the different classes have been extracted from the training data, it is possible to evaluate the discriminative power of the single features. Indeed, the mass function for a given feature can be considered as a decision value for the classification, as mentioned in section 2.3.2.1. Thus, each sample of a validation set is considered and the corresponding belief over the different classes is computed from the mass function. The sample is then assigned to the class having the highest belief. Performing this for all available

samples in the validation set allows to compute the correct classification rate, and thus the discriminative power, for a given single feature.

Since our goal at that step is to select the best single features, they are ordered in descending order according to their correct classification rates $R_{single}(F^n)$ as $F_s^1, F_s^2, ..., F_s^N$, where N is the total number of features in the whole feature set.

In order to reduce the computational burden during the feature selection, an initial feature set FS_{ini} is built of the L best features in the re-ordered feature set according to a certain threshold for classification rates as $FS_{ini} = \{F_s^1, F_s^2, ..., F_s^L\}$.

The threshold is obtained according to the best classification rate as:

$$R_{single}(F_s^L) \geq thres_1 * R_{best_1} \tag{2.50}$$

where $R_{best_1} = R_{single}(F_s^1)$. The value of $thres_1$ in this formula may vary for different problems in order to reach a balance between the overall performance and the calculation time for experiments. For example, in our work, $thres_1$ is experimentally set to 0.7 and around 100 features are kept above this value in our application of image categorization.

Only the features selected in the set FS_{ini} will attend in the latter steps of feature selection process. The elements (features) in FS_{ini} are considered as subsets of size 1.

Stage 3: Combination of features for the generation of the feature subsets.

For iterations dealing with subsets of size k with $k \geq 2$, the generation of a new feature subset consists in the creation of a new feature by the fusion of two original features (more precisely, their mass function) thanks to the application of an operator of combination. Then, the resulting subsets are re-ordered and selected according to their discriminative power as in the case of single features in stage 2.

Let note the set of all the feature subsets of size k as FS_k and the set of the selected subsets of size k as FS'_k. Thus, FS_1 corresponds to the original whole feature set, and $FS'_1 = FS_{ini}$. For $k \geq 2$, the set of the feature subsets FS_k is noted

as:

$$\begin{aligned} FS_k &= Combine(FS'_{k-1}, FS_{ini}) \\ &= \{F_{ck}^1, F_{ck}^2, ..., F_{ck}^{N_k}\} \end{aligned} \tag{2.51}$$

where the operator "Combine" represents the generation of new features by combining features from each of the two sets FS'_{k-1} and FS_{ini} with all the possible combinations except those in which the elements from FS_{ini} appear in the original features during the generation process leading to the elements of FS'_{k-1}. F_{ck}^n represents the n-th generated new feature and N_k is the number of elements in the set FS_k.

Assume that M classes are considered in the classification problem. For the i-th class A_i, the mass m_{ck}^n for the new feature F_{ck}^n, which is generated from F_{ck-1}^u of FS'_{k-1} and F_s^v of FS_{ini} is computed as

$$m_{ck}^n(A_i) = Comb(m_{ck-1}^u(A_i), m_s^v(A_i)) \tag{2.52}$$

where $m_{ck-1}^u(A_i)$ and $m_s^v(A_i)$ are mass functions associated respectively with features F_{ck-1}^u and F_s^v. $Comb(x, y)$ is one of the possible combination operators (TBM for example).

The correct classification rates of the combined new features can be obtained from their belief masses, considered as decision values. Indeed, the class with the highest belief mass is assigned to the data samples. The combined new features can then be ordered in descending order according to the correct classification rates as for FS_{ini}.

Let note the best feature from FS_k as F_{ck}^{best} having the highest recognition rate R_{best_k}.

Following the same process as the selection of FS_{ini} during the evaluation of the single features, a threshold is set to select a certain number of subsets with size k to take part into the next step of forward selection. The set of the best ordered

features according to the recognition rate is noted as

$$FS'_k = \{F_{ck}^{1\,\prime}, F_{ck}^{2\,\prime}, ..., F_{ck}^{L_k\,\prime}\} \tag{2.53}$$

where $F_{ck}^{1\,\prime} = F_{ck}^{best}$ and L_k is the size of FS'_k set, being chosen so that $R(F_{ck}^{L_k\,\prime}) \geq thres_k * R_{best_k}$. In order to simplify the selection, the threshold value $thres_k$ is set to be the same value as $thres_1$ (0.7) in every step without any adaptation.

Stage 4: Stop criterion and selection of the best feature subset.

The stop criterion of ESFS is reached when the best classification rate begins to decrease while increasing the size of the feature subsets. In order to reduce the sensitivity to local variations, the forward selection stops when the classification performance continues to decrease during two steps, $R_{best_k} < min(R_{best_{k-1}}, R_{best_{k-2}})$.

2.3.3 Experimental results

For elaborating an image categorization system, efficient classifier need to be trained using pertinent information in the image carried by features. As generally numerous features are extracted, a selection of the most discriminative ones is often essential in order to simplify the models and allow a better efficiency both in terms of computational cost and recognition ability.

To evaluate our feature selection method within this context, four configurations of experiments have been driven on an image dataset: one with all the features without selection; the second with features selected by filter methods, such as Fisher filter [Narendra & Fukunaga 1977] and Principal Component Analysis (PCA) [Jolliffe 2002]; the third with features selected using a wrapper method, such as SFS [Whitney 1971], SFFS [Pudil *et al.* 1994b] and OS [Somol & Pudil 2000]; the last one with the best features selected by ESFS. As numerous combination rules exist for combining different mass functions from several sources of information in ESFS, we have tested the TBM rule, Dempster's rule and Yager's rule. Besides all these three rules, one triangular norm (T-norm) [Schweizer & Sklar 1983] has been also

Figure 2.5: Some sample images from SIMPLIcity dataset (from top to bottom, from left to right, they belong to Beach, Building, Bus, Flower, Horse and Mountain).

considered here for comparison purpose, whose formulae is as follows:

$$\forall A \subseteq \Omega \quad m^{\Omega}_{S1,S2}(A) = \max\{1 - [(1 - m^{\Omega}_{S1}(A))^p + (1 - m^{\Omega}_{S2}(A))^p]^{\frac{1}{p}}, 0\} \qquad (2.54)$$

where $p > 0$ is a parameter.

Moreover, four types of one step global classifiers have been considered: Multilayer Perceptron (Neural Network, denoted as MP in the following text), Decision Tree (C4.5), K-Nearest Neighbors (K-NN), and multi-class SVM (C-SVC). Each classifier has been tested with several parameter configurations, and only the best results are kept. The experiments are carried out on TANAGRA platform [Rakotomalala 2005] with 4-fold cross-validation. The detailed experiments are presented in the following subsections.

2.3.3.1 Dataset

Our experiments dealing with image classification have been performed on the SIMPLIcity dataset [Wang *et al.* 2001a]. It is a subset of the COREL database, consisting of 10 image categories, each containing 100 images. For the purpose of evaluating our ESFS based feature selection for image categorization, 6 categories containing totally 600 images have been chosen in our experiments: Beach, Building, Bus, Flower, Horse, and Mountain. Some sample images are presented in Figure 2.5.

2.3.3.2 Feature extraction

In order to carry visual information according to color, texture and shape, a total number of 1056 features have been computed to represent each image sample from SIMPLIcity dataset. The corresponding feature set thus includes Color Coherence Vectors (CCV) [Pass & R. Zabih 1997], Color Auto Correlogram (CAC) [Huang *et al.* 1997], Color Moments (CM) [Stricker & Orengo 1995a], Texture Auto-Correlation (TAC) [Tuceryan & Jain 1993], Grey Level Co-occurrence Matrix (GLCM) [Tuceryan & Jain 1993] and Edge Histogram (EH) [Won 2004]. The high number of features compared to the relatively low number of samples available for training classifiers strongly suggests the use of a feature selection method to decrease the classification models complexity and thus to improve classification accuracy.

2.3.3.3 Results

Table 2.2 presents the mean correct classification rates (or classification accuracy) for all the classifiers tested in this experiment.

ESFS_filter indicates that the embedded feature selection method ESFS has been used in a filter way to provide discriminative features used in a second step by the classifiers. Moreover, as it has been mentioned in the previous section, ESFS can also be used as a classifier, which is denoted in Table 2.2 as ESFS_cls. The difference of combination rules used for ESFS is furthermore marked by _TBM, _Dempster, _Yager and _T-norm following ESFS_filter or ESFS_cls in the table, representing respectively TBM rule, Dempster's rule, Yager's rule and T-norm presented previously.

Let us first of all focus on the different combination rules in the category of ESFS_filter. We can note that TBM, Dempster and T-norm give almost the same performance, with a little advantage for TBM used with K-NN, MP, C-SVC and for T-norm in C4.5. However, Yager failed to get the results in the same level as other combination rules. This means that reassigning the conflict to the whole set of definition when it is detected during the combination might not be a reasonable

Table 2.2: Comparison between the classification accuracy without feature selection and with the features selected by different methods for image categorization.

Classification rate	C4.5	K-NN	MP	C-SVC
No Selection	69.4%	80.0%	79.7%	87.3%
Fisher Filter	68.9%	79.8%	83.2%	82.9%
PCA	68.3%	52.1%	80.5%	51.9%
SFS	69.4%	79.5%	80.6%	81.2%
SFFS	71.7%	44.2%	79.5%	86.9%
OS	**71.8%**	77.9%	83.8%	86.4%
ESFS_filter_TBM	69.6%	**83.9%**	**87.1%**	**87.7%**
ESFS_filter_Dempster	69.2%	83.1%	87.0%	87.4%
ESFS_filter_Yager	65.1%	68.2%	70.8%	70.4%
ESFS_filter_T-norm	70.8%	83.0%	87.1%	87.3%
ESFS_cls_TBM	60.0%			
ESFS_cls_Dempster	63.3%			
ESFS_cls_Yager	61.7%			
ESFS_cls_T-norm	71.0%			

choice in our case. Moreover, we find that T-norm also provides good results in spite of its simpler principle for computation.

Now let us move to the comparison between ESFS_filter and other feature selection methods. The results show that for all of the classifiers tested in this experiment, the features selected by ESFS used in a filter way offer better classification results than both the original features without selection and the features selected by other methods, with the exception of C4.5. Since C4.5 can also be itself considered as an embedded feature selection method, the performances of all the feature selection methods associated to it are very close and are not improved so much. We can also observe from the table that for some classifiers, such as K-NN and MP, the superiority of ESFS_filter is obvious and presents an improvement from 4% to 8% in the classification rate compared to other methods. Moreover, focusing on C-SVC, we find that the classification rate using the feature selection methods decreased compared to that of "No Selection" except in the case of ESFS_filter, which performed the same as "No Selection". This phenomenon is probably due to the high ability of SVM to handle small datasets, high dimensional pattern recognition problems and even in this case, our ESFS_filter approach has still maintained the highest performance. Thus, these experimental results have shown that ESFS has been the most efficient to select the discriminative features for this image categorization problem.

Finally, if we turn to ESFS_cls in which ESFS is also used to classify the test samples, we found regrettably that the best rate of 71.0% obtained with T-norm is worse than other approaches and is only comparable to the one of C4.5. The results of other combination rules are even much worse than T-norm, which suggests that ESFS_cls is not suitable to this image categorization task.

Besides the classification performance, another essential criterion for a classification system is its computational complexity. If we compare the computational cost between original SFS and ESFS, as the first one works as a wrapper feature selection method, a training of the classifier (MP for example) needs to be performed for each possible combination of features, at each step of the SFS process, whereas ESFS carries its own classifier thanks to mass functions which are used both for feature combination and as decision value, and thus does not need any training during

the selection process. So, the computational cost of ESFS is much lower than the one of SFS. Moreover, as SFFS and OS are also wrapper methods, they are computationally very expensive and in some cases even more expensive than SFS. Taking the comparison of ESFS and SFS as an example, experiments presented previously have been realized on a PC computer equipped with Intel Core Duo T7200/2GHz and 2GB memory using Windows XP system. In this case, the selection process with ESFS takes around 50 minutes whereas the selection by SFS lasts from 8 hours for C-SVC to two weeks for MP.

2.3.4 Conclusion on feature selection

ESFS has been presented in this section as a novel feature selection method, which relies on the simple principle to add incrementally most relevant features. For this purpose, each feature is represented by a mass function from the evidence theory, which allows to merge the information carried by features in an embedded way, and so leading to a lower computational cost than wrapper method. Being evaluated in the visual object categorization, the obtained results shown that selecting relevant features improves the classification accuracy, and for this purpose, ESFS, used as a filter selection method, performs better than the traditional filter method, namely Fisher and PCA algorithm, and wrapper method, namely SFS, SFFS and OS. As different combination rules are available to merge the information carried by features within ESFS, we have also tested 4 rules here, namely TBM, Dempster, Yager and T-norm. We can see from the results that Dempster, Yager and T-norm give us almost the same performance whereas Yager seriously hurts it, suggesting us to integrate the conflict information in a more efficient way in future. Finally, although ESFS can also be directly used a classifier, it failed to obtain comparable results as other classifiers in our experiments.

2.4 Image representation

The aim of the image representation for classification is to construct a discriminative representation which models the distribution of the extracted local features, with

the purpose of being efficiently classified by a certain classifier later. Recently, the most successful approach for this topic is called "bag of features", which has been largely used in Pascal challenge [Everingham *et al.* 2007] [Everingham *et al.* 2008] and many other works. In this section, we will first give a literature review about this approach and its variations and extensions, and then present our proper solution for image representation, as well as our proposed region based features to be used with our image representations.

2.4.1 Literature review

The term "bag of features" comes from the "bag of words", firstly introduced for the analysis of text documents [Salton & McGill 1983] [McCallum & Nigam 1998]. In such a representation, a text document is encoded as a histogram of the number of occurrences of each word. Similarly, one can characterize an image by a histogram of visual words count. It effectively provides a mid-level representation which helps to bridge the semantic gap between the low-level features extracted from an image and the high-level concepts to be categorized.

A typical "bag of features" approach consists of two main stages: vocabulary construction and histogram computation. Visual vocabulary is usually learned from the training local features extracted from the training set of images using unsupervised or supervised methods. The histogram computation aims at computing a discriminative histogram representing an image given a learned visual vocabulary. We introduce in the following some representative methods for each of these two stages. As the traditional "bag of features" discards all spatial information between the extracted local features, some methods aiming at reusing this precious information are also presented.

2.4.1.1 Vocabulary construction

The k-means algorithm has been originally employed to cluster the local features into k bins with k predefined empirically, thus constructing a visual vocabulary in which each centroid corresponds to a visual word [Dance *et al.* 2004] [Lazebnik *et al.* 2006].

This algorithm proceeds by iterated assignments of points to their closest cluster centers and re-computation of the cluster centers, and is known for its simple and efficient implementation. However, it has the defect that cluster centers are drawn irresistibly towards denser regions of the sample distribution which do not necessarily corresponds to discriminative patches. [Jurie & Triggs 2005] proposed a radius-based clustering, which avoids setting all clusters into high density areas and assigns all features within a fixed radius of r to one cluster. Another approach developed in Xerox Research Center Europe (XRCE) consists in using GMM to model the distribution of the local features extracted from the training images [Perronnin & Dance 2007] [Perronnin *et al.* 2006]. The optimized GMM is then considered as a visual vocabulary where each gaussian (with its parameters: weight π, mean μ, co-variance Σ) corresponds to a visual word.

As proven in most of the articles presented in the previous paragraph, the best performance is always obtained using a vocabulary with large size, ranging from several hundreds to several thousands. But considering the computational cost of the following histogram computation stage which directly depends on the number of visual words, one may prefer to get a more compact vocabulary. In [Winn *et al.* 2005], the initial visual vocabulary with several thousands of words is further compressed to its optimal size (approximatively 200 words), without any loss of its discriminative ability, through a supervised iterative merging technique inspired by the information bottleneck principle [Tishby *et al.* 1999]. Another interesting work in the direction of reducing the computational cost is [Moosmann *et al.* 2007], which organizes the vocabulary in a tree structure using randomized clustering forests. However, the obtained vocabulary in both cases has been fitted to the set of categories under consideration and should be retrained when some new category appears.

Some researchers have departed from the idea of having one universal vocabulary for all the training images from the whole set of categories, such as [Zhang *et al.* 2007] [Perronnin *et al.* 2006] [Farquhar *et al.* 2005]. [Zhang *et al.* 2007] uses k-means to cluster the local features of each image to a vocabulary (called signature in the paper) with a fixed number of words, and then measures the similarity between each pair of signatures using Earth Mover's Distance

(EMD) [Rubner *et al.* 2000] or χ^2 distance which will be used later by kernel-based classifier to perform the classification, such as SVM. However, the use of per image vocabulary in such approach requires online learning for the image vocabulary, which may lead to a high computational cost. [Farquhar *et al.* 2005] proposes to train one vocabulary for each category and then merges these category specific vocabularies together to build the final single vocabulary. Despite the promising results they have obtained, it is not practical face to the problem with large number of categories. Indeed, the size of the merged vocabulary and the corresponding histogram representation grows linearly with the number of categories, thus quickly leading to the "curse of dimensionality" problem and increasing the histogram computation cost. In [Perronnin *et al.* 2006], a universal vocabulary, which describes the visual content of all the considered categories, and a series of category specific vocabularies, which are obtained through the adaptation of the universal vocabulary using category specific data, are trained consecutively. Then an image is represented by a set of histograms of size $2 \times K$ (K is the size of the vocabulary), one per category. Each histogram describes whether an image is more suitably modeled by the universal vocabulary or the corresponding adapted vocabulary.

Another group of methods claimed that semantic relation between the features is useful for image categorization and attempted to bring the semantic information into visual vocabulary construction [Vogel & Schiele 2004] [Yang *et al.* 2008b] [Liu *et al.* 2009]. A semantic vocabulary is constructed by manually associating the local patches to certain semantic concepts such as "stone", "sky", "grass" etc in [Vogel & Schiele 2004]. But the fact that it requires huge manual labor for labeling the local patches when large amount of training data should be treated make it impractical in such cases. [Yang *et al.* 2008b] proposes to unify the visual vocabulary generation and classifier training processes, and then encoding an image by a a sequence of visual bits which capture different aspects of image feature and constitute the semantic vocabulary. The method of [Liu *et al.* 2009] can automatically learn a semantic visual vocabulary using diffusion maps which capture the semantic and geometric relations of the feature space.

2.4.1.2 Histogram computation

Once the visual vocabulary determined, it is now to characterize the visual content of an image by a histogram of visual words frequencies. In the literature, two strategies have been commonly used for histogram computation: hard assignment and soft assignment.

Hard assignment simply assigns the feature vectors extracted from an image to their nearest visual words respectively, according to a certain distance measure, as shown in (2.55):

$$HA(w) = \frac{1}{N}\sum_{n=1}^{N}\begin{cases} 1 & \text{if } w = \arg\min_{v \in V}(D(v, r_n)) \\ 0 & \text{otherwise} \end{cases} \tag{2.55}$$

where w is a visual word in the vocabulary V, N is the number of local patches in an image, r_n is the feature vector extracted from the n-th local patch, and $D(v, r_n)$ is the distance between v and r_n. However, problems occur for feature vectors that are located in the ambiguous areas. [van Gemert *et al.* 2008] and [van Gemert *et al.* 2010] propose to distinguish two different issues associated with hard assignment: 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 code plausibility denotes the problem of selecting a visual word without a suitable candidate in the vocabulary. An illustration of these two issues is shown in Figure 2.6.

Concerning the soft assignment, there are basically two approaches. The first one consists in performing probabilistic clustering, namely GMM, and then each image feature vector contributes to multiple visual words according to its posterior probability given the visual word [Farquhar *et al.* 2005] [Perronnin *et al.* 2006] (see 2.2.3.1 for more details). Although these works are able to deal with word uncertainty by considering multiple visual words, they ignore the word plausibility. On the contrary, [Boiman *et al.* 2008] copes with the word plausibility by using the distance to the single best neighbor in feature space without taking into account the word uncertainty. In [van Gemert *et al.* 2008] and [van Gemert *et al.* 2010], they

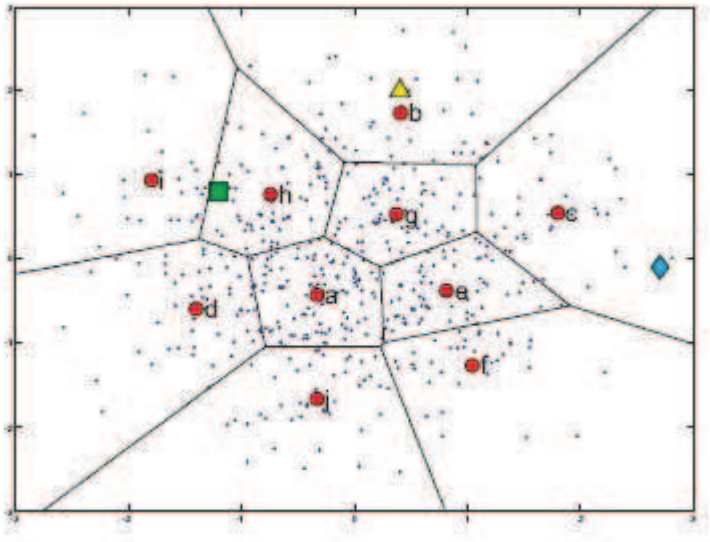

Figure 2.6: Illustration of visual word uncertainty and plausibility. The small dots represent image features, the labeled red circles are visual words found by unsupervised clustering. The triangle represents a data sample that is well suited to hard assignment approach. The difficulty with word uncertainty is shown by the square, and the problem of word plausibility is illustrated by the diamond. Source: [van Gemert *et al.* 2008]

make the assignment a decreasing function of the Euclidean distance between the feature vector and the word centroid, paired with a gaussian kernel:

$$G_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{1}{2}\frac{x^2}{\sigma^2}) \tag{2.56}$$

where σ is the smoothing parameter of kernel G. Thus they propose three different formula to cope with word uncertainty (UNC), word plausibility (PLA) and both of them (KCB) respectively:

$$UNC(w) = \frac{1}{N}\sum_{n=1}^{N}\frac{G_\sigma(D(w,r_n))}{\sum_{k=1}^{|V|}G_\sigma(D(v_k,r_n))} \tag{2.57}$$

$$PLA(w) = \frac{1}{N}\sum_{n=1}^{N}\begin{cases} G_\sigma(D(w,r_n)) & \text{if } w = \arg\min_{v\in V}(D(v,r_n)) \\ 0 & \text{otherwise} \end{cases} \tag{2.58}$$

$$KCB(w) = \frac{1}{N}\sum_{n=1}^{N} G_\sigma(D(w, r_n)) \tag{2.59}$$

2.4.1.3 Spatial information

The "bag of features" approach views images as an orderless distribution 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] proposes 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 2005] which build pyramid in feature space while discarding the spatial information (see 2.2.3.2 for more details). The spatial pyramid consists in performing pyramid matching in the two-dimensional image space, and uses 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 channel kernels:

$$\kappa^L(x, y) = \sum_{m=1}^{M} K^L(x_m, y_m) \tag{2.60}$$

where $K^L(x_m, y_m)$ is the pyramid match kernel. This approach has the advantage of maintaining continuity with the "bag of features" paradigm. In fact, it reduces to a standard bag of features when $L = 0$. Figure 2.7 shows an example of constructing a three-level spatial pyramid.

The winning system of image classification session in [Everingham *et al.* 2008] provides some improvements on the spatial pyramid method in order to adapt it more appropriately to the VOC use. They first of all divide the image into 2×2 and 1×3 level, as shown in Figure 2.8. Then one unique vocabulary is trained for the whole image and the histograms are computed on this vocabulary for each subregion, which are later fused using the extended gaussian kernel.

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

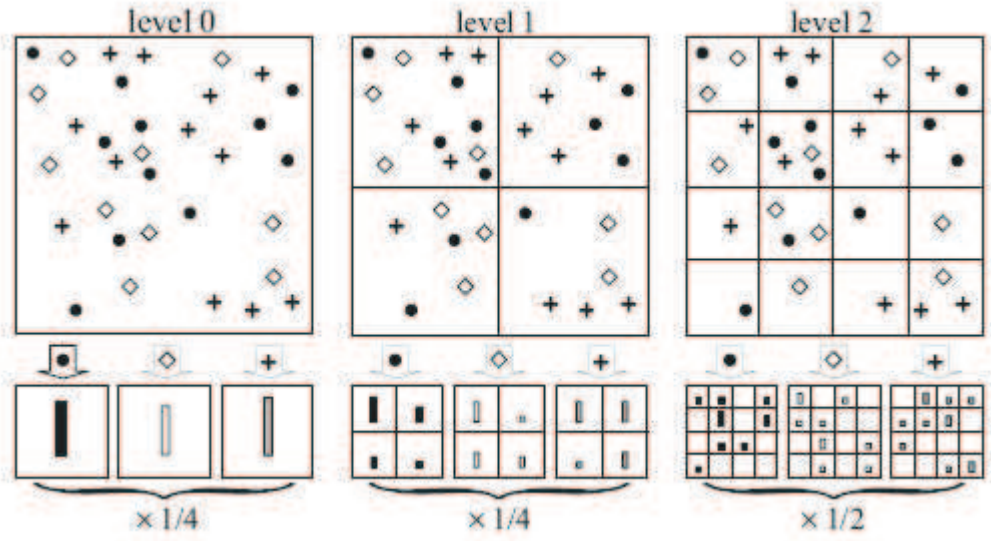

Figure 2.7: An example of constructing a three-level spatial pyramid. The image has three feature types, indicated by circles, diamonds, and crosses. At the top, the image is subdivided at three different levels of resolution. Next, for each level of resolution and each channel, the features that fall in each spatial bin are counted. Finally, each spatial histogram is weighted according to equation (2.30). Source: [Lazebnik *et al.* 2006]

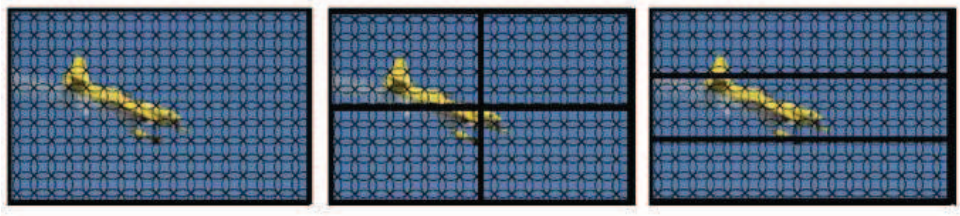

Figure 2.8: The spatial pyramid used in the winning system of image classification session in [Everingham *et al.* 2008]

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.

2.4.2 PMIR: a Polynomial Modeling based Image Representation

Once having extracted a set of local feature vectors from an image, an efficient characterization of the visual content represented by this information needs to be elaborated. A simple approach would be to concatenate these feature vectors to build a huge single vector. However, the number of local feature vectors extracted can vary from one image to another. Since machine-based learning schemes require input data to have a constant size, a solution is to model the distribution of feature vectors and to use the parameters of this distribution as new features for the classification. The popular "bag of features" approach follows this strategy: the distribution of original features is modeled thanks to a histogram for each image on the basis of a "visual vocabulary", which can be built either by using a clustering algorithm or by using a parametric distribution such as GMM.

The basic problem is that the "bag of features" approach, while adapting the best practice from text categorization, does not necessarily correspond to a human visual perception process which seems to be ruled by some Gestalt principles according to several studies on visual perception [Kaniza 1997] [Wertheirmer 1923] and supposed to perform a holistic analysis combined with a local one through a fusion process. Moreover, the optimal size of this visual vocabulary is hard to be fixed as there is no easy intuitive counterpart in image compared to keywords in text document. Regarding GMM as an example, if the number of gaussians is too small then it can not supply enough normal distributions for a large amount of diversified feature vectors to be modeled, while a too high number of gaussians suffers from an insufficient number of training feature vectors to optimize the model parameters.

Therefore, we first propose to make use of some region-based meaningful features extracted from visual regions with neighborhood information, in addition to the popular SIFT feature. These region based features result from perceptually significant

"Gestalts" segmented according to some basic Gestalt grouping laws. Secondly, we present a novel image representation method, namely Polynomial Modeling based Image Representation (PMIR), to cooperate with our proposed region-based features and SIFT feature. Their interest is three-fold. First, we circumvent the difficulty of fixing arbitrarily the size of visual vocabulary; secondly, we avoid the inaccurate assumption of Gaussian distribution of feature vectors and thirdly we can cope with a small number of feature vectors per image as it is particularly the case with our region-based features.

2.4.2.1 Our proposed region-based features

Our basic hypothesis is that effective visual object classification or detection should be inspired by some basic human image interpretation principles. Thus we make use of some basic principles from the Gestalt theory for feature extraction, in particular the well known Gestalt laws of Perceptual Organization which suggest both the grouping of pixels into homogeneous regions as well as the interaction between regions.

Desolneux et al. have given in [Desolneux *et al.* 2008] a comprehensive introduction to Gestalt theory in an image analysis perspective. Gestalt theory relies on the assumption of active grouping laws in visual perception which recursively cluster basic primitives into a new, larger visual object, called *gestalt*. These grouping laws follow criteria such as spatial proximity, color similarity. These laws also highlight the interaction between regions. This interaction is confirmed by Navon [Navon 1977] who showed the preponderance of global perception over local perception. Following these basic Gestalt perception laws, we also claims that an effective description of the visual content of an image needs to model the partial gestalts and their interactions. We feel that lacking these principles, the popular "bag of features" approaches deprive themselves of meaningful information. One exception is the work of Barnard et al. [Barnard *et al.* 2003] which is a region-based approach where regions are labeled with probable categories. However, they don't take into account the interaction among regions.

As the regions resulted from a segmentation process may not be consistent

with object boundaries, individual regions are not labeled as did Barnard et al. [Barnard *et al.* 2003]. Regions produce a feature vector which is supposed to have no meaning on its own but that can contribute to one or more classes. Regarding features, we propose using visually meaningful features, such as color and line segment based features which we will extend to provide information from neighboring regions. In the following, we first introduced our Gestalt-inspired region segmentation scheme [Fu *et al.* 2008] and then the color and segment based features we extract from the region map given by our segmentation scheme.

Region segmentation scheme As we have seen previously, studies on human perception strongly hint at a region based approach. On the other hand, introducing region segmentation brings about a host of new problems regarding segmentation robustness and accuracy. Thus, while this approach suits human perception better, we have no guarantees that its benefits will overcome its drawbacks. Here we specifically designed a robust region segmentation method that aims at automatically producing coarse regions from which we can consistently extract feature vectors [Fu *et al.* 2008]. We will now briefly describe the outline of the algorithm.

The principle of our region segmentation algorithm is to segment an image into partial gestalts for further visual object recognition. We thus made use of the following Gestalt basic grouping laws in our gestalt construction process: the color constancy law stating that connected regions where color does not vary strongly are unified; the similarity law leading to group similar objects into higher scale object; the vicinity law suggesting grouping close primitives with respect to the others; and finally good continuation law saying that reconstructed amodal object, i.e partially perceived physical structure which is reconstructed through understanding, should be as homogenous as possible. Because those laws are defined between regions and their context, at each step we assess the possibility to merge regions according to global information.

The algorithm is based on color clustering but also includes an extra post-processing step to ensure spatial consistency of the regions. In order to apply previously mentioned Gestalt laws, we defined a 3-step process: first we filter the image

and reduce color depth, then we perform adaptive determination of the number of clusters and cluster color data and finally we perform spatial processing to split unconnected clusters and merge smaller regions.

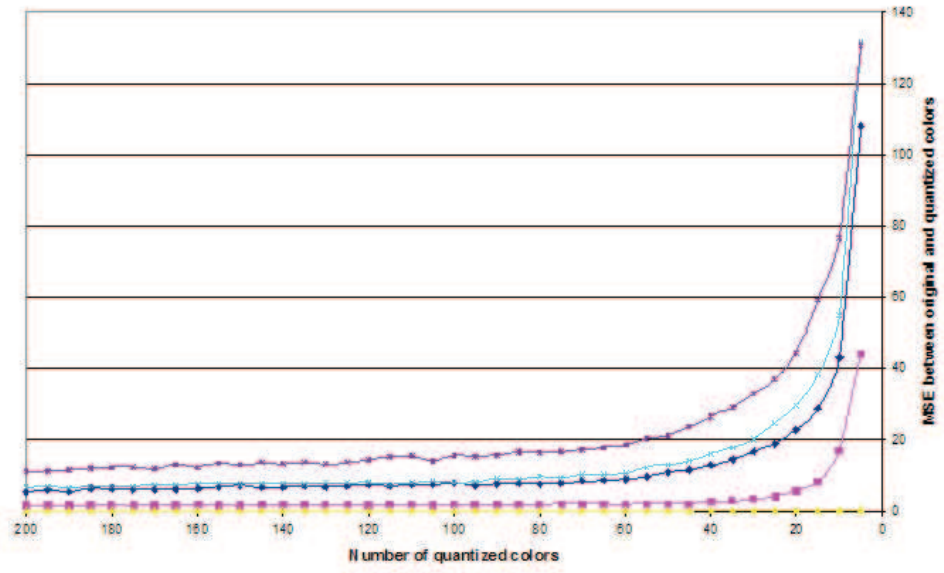

Figure 2.9: Evolution of MSE between quantized and original colors.

Images are first filtered for robustness to noise; colors are then quantified by following a first, fast color reduction scheme using an accumulator array in CIELab color space to agglomerate colors that are perceptually similar. In the second step, we use an iterative algorithm to determine a good color count which limits the quantization error. Indeed, quantization error measured by MSE between original and quantized colors evolves as per Figure 2.9 according to the number of clusters.

This clearly shows a threshold cluster number under which quantization MSE begins to rise sharply. By performing several fast coarse clustering operations using Neural Gas algorithm [Martinetz & Schulten 1991], which is fast and less sensitive to initialization than its counterparts such as K-means, we are able to compute the corresponding MSE values and generate a target cluster count. We then use hierarchical ascendant clustering which is more accurate but much slower thus executed only once in our case, to achieve segmentation. The third step consists in splitting spatially unconnected regions, merging similar regions and constraining segmenta-

tion coarseness. Merging of similar regions is achieved through the use of the squared Fisher's distance as (2.61) (used for a similar task in [Zhu & Yuille 1996]). where n_i, μ_i, σ_i^2 are respectively the number of pixels, the average color and the variance of colors within region i. This distance still stays independent towards image dynamics as it involves intra-cluster distance vs. inter-cluster distance. Finally, regions which are too small to provide significant features are discarded.

$$D(R_1, R_2) = \frac{(n_1 + n_2)(\mu_1 - \mu_2)^2}{n_1\sigma_1^2 n_2\sigma_2^2} \tag{2.61}$$

Figure 2.10: Examples of segmented images.

With this algorithm we obtain consistent coarse regions that can be used for our classification system. Sample segmentation results on Pascal challenge dataset images are given in Figure 2.10. As we can see, our Gestalt-inspired segmentation algorithm has automatically adapted its segmentation process to the color depth of the images, producing significant partial gestalts.

Region-based feature extraction In order to represent the information carried by regions, we make use of two kinds of features: color features and segment features. Region based color features aim at capturing a coarse perception

of partial gestalts, in the form of color moments (mean, variance and skewness) [Stricker & Orengo 1995b] for each color channel. These features are quite compact and have proven as efficient as a high dimension histogram [Deng *et al.* 2001]. Various color spaces were experimented for the computation of these features and best results were achieved in the CIELch color space which is derived from CIElab as in (2.62) and best fits to the human perception [Trémeau *et al.* 2004].

$$L_{Lch} = L_{Lab} \qquad c = \sqrt{a^2 + b^2} \qquad h = \arctan \frac{b}{a} \tag{2.62}$$

The segment features aim at capturing some textual and geometrical properties of partial gestalts. We thus developed segment based features relying on a fast connective Hough transform [Ardabilian & Chen 2001] that performed well in global image classification [Pujol & Chen 2007] and more specifically provided more significant information than gradient based features. These features are relevant regarding our approach of following human visual interpretation as, most of the time, there are few segments within a region but, on the other hand, they represent features that stand out visually and their simple presence is significant.

The principle of our segment based feature extractor is the following. As for any other Hough transform, we start from an edge map of the processed image. Because we wish to avoid problems related to edge thickness, we use a Canny Edge Detector [Canny 1986] to process our image in order to ensure a one pixel thickness for our edge map. For an edge point on the edge map, we examine its neighborhood identified by its relative angular position (r, θ): each direction θ is processed while a connected edge is found at distance $r + 1$, which gives us a list of segments by orientation for this edge point. Once we have this list, we store the longest segment and remove it from the edge map. To avoid hindering intersecting segment detection, we use two separate edge maps: one for segment source point detection and one for connected points detection. Removed segments are only removed from the source point map, which avoids detecting the same segment twice while preserving intersecting segments. These segment features are extracted once for the whole image.

During this extraction step, we can build a map from image coordinates to the corresponding segments. Therefore, we can quickly detect segments within a region. For validation purposes, our "segment" shape features are a simple histogram combining length and orientation. In order to obtain scale invariant features, we normalize lengths by dividing them by the longest segment length. We then obtain rotation invariance by computing an average orientation in order to have a stable average and by expressing all angles with respect to this average direction. We therefore obtain a feature that is invariant to translation, scale as well as rotation. The size of the histograms was experimentally determined and set to 6 bins for orientation and 4 for length.

Finally, in order to include neighborhood information, our region based features (color moments and Hough segment features), are expressed at four different levels: original region, region + neighbors, region + neighbors + neighbor's neighbors, etc. Those levels are concatenated in the final feature vector. This is a basic way to integrate spatial relationship but also to include global information in each feature vector. On most images, the fourth level will represent features extracted over the whole image. This process leads to our two final region-based features, that are called in the subsequent Region based Color Moments (RCM) and Region based Histogram of Segments (RHS).

2.4.2.2 PMIR principle

We now turn to the problem of image modeling and classification. Instead of building a "visual vocabulary" as in the "bag of features" approach, we propose here a simple polynomial modeling to characterize the visual content represented by the set of feature vectors extracted in the previous section. The basic idea is to consider the distribution of values in each component of these feature vectors and to model such a distribution by a simple polynomial. The coefficients of these polynomials will then be considered as the feature vector characterizing the visual content of an image.

The polynomial model for a given feature distribution is computed as follows. Given the set D of the distribution values $D = \{(x_1, y_1), ..., (x_M, y_M)\}$ (M is the number of values), a polynomial $f(x)$ of degree N, described by its set of coefficients

$P = \{p_0, p_1, ...p_N\}$, is computed to interpolate the data, by fitting $f(x_i)$ to y_i in a least squares sense. Thus, vector P can be used to characterize the distribution D. An example is given in Figure 2.11. Once the distribution of each component from the feature set has been modeled thanks to a polynomial, a new image feature vector Q is produced by concatenating the coefficients of all polynomials.

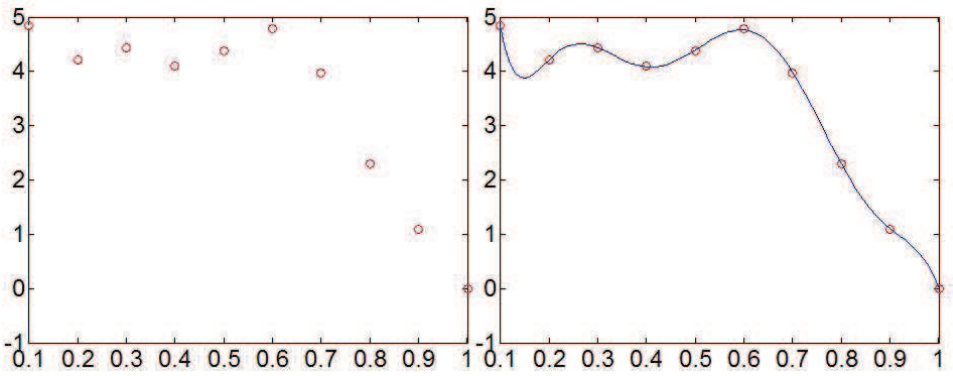

Figure 2.11: (Left) Distribution values for one component of the image feature set. (Right) A polynomial curve for modeling the distribution in (Left). The horizontal axis represents the values of bins equally partitioning the interval [0,1] while the vertical axis is the number of data points located in the corresponding bin.

Assuming that our feature vector has L components and each component is modeled by a polynomial of degree N, then the vector Q has a dimension of $(N + 1) * L$, which generally ranges from hundreds to thousands. A vector of such high dimensionality used for classification can also lead to the "curse of dimensionality" [Bellman 1961]. Consequently, we further apply the dimensionality reduction methods on this new vector. Several methods may be conceivable for this purpose [Saeys *et al.* 2007], and some of them are presented in section 2.3. We have chosen the Canonical Discriminant Analysis (CDA) [Fisher 1936] as it is fast and generally enables a strong reduction of the feature vector dimensionality since the new representation space which distinguishes the best the different classes contains in most cases $K - 1$ axes, K being the number of classes. Thus with the help of this method, the overall feature vector Q becomes a much more simplified vector which is called in the subsequent *Polynomial Modeling based Image Representation* (PMIR) [Fu *et al.* 2008].

2.4.2.3 Experimental results

Given an image to classify, we first need to characterize its visual content by extracting a set of feature vectors as proposed in section 2.4.2.1. However, other feature vectors, such as for instance SIFT, can also be used in our PMIR and the following classification process. Our purpose is not only to compare the use of these region based features with popular SIFT features but also to check the efficiency of their combination. Upon these feature vectors, our proposed image representation, PMIR, is computed, leading to a single global feature vector for the input image. This new single global feature vector is then fed to a classifier beforehand trained to judge whether this image contains a specified object. Any classifier, such as neural networks or SVM, can be used for categorization of such an image representation.

We have used in our experiments the dataset of Pascal challenge 2007 [Everingham *et al.* 2007]. The goal of this challenge is to recognize objects from a number of visual object categories in realistic scenes (i.e. not pre-segmented objects). It is fundamentally a supervised learning learning problem in that a training set of labeled images is provided. More concretely, this dataset consists of 20 object categories and contains 2501 images taken in real world provided for training, 2510 for validation and 4952 for testing. The 20 object categories are: Aeroplane, Bicycle, Bird, Boat, Bottle, Bus, Car, Cat, Chair, Cow, Diningtable, Dog, Horse, Motorbike, Person, Pottedplant, Sheep, Sofa, Train, Tvmonitor. One main characteristic of this dataset is that multiple objects from multiple categories may be present in the same image, which makes it more realistic and difficult. A total of 9 groups has participated in this challenge 2007 and they have submitted 17 different methods. There are two main competitions, and two smaller scale "taster" competitions in the challenge:

- **Main competitions**
  - Classification: for each of the 20 categories, predicting presence/absence of an example of that category in the test image. This is just the target task of this thesis.
  - Detection: predicting the bounding box and label of each object from the

20 target categories in the test image.

- **Taster competitions**
  - Segmentation: generating pixel-wise segmentations giving the category of the object visible at each pixel, or "background" otherwise.
  - Person Layout: predicting the bounding box and label of each part of a person (head, hands, feet).

For the purpose of evaluating our classification approaches, we have chosen 5 semantic representative classes namely aeroplane (238 images for training), bicycle (243 images for training), bus (186 images for training), horse (287 images for training) and person (2008 images for training). Some image samples for these 5 classes are given in Figure 2.12.

Figure 2.12: Some sample images of 5 representative classes from Pascal challenge 2007 dataset (from left to right: Aeroplane, Bicycle, Bus, Horse, Person)

As we have mentioned previously, our two region-based features, namely RCM and RHS, as well as the popular SIFT features (computed using the C# "libsift" implemented by Sebastian Nowozin [Nowozin 2005] for their extraction) have been used in these experiments. Since they represent features of different natures, we believe that these features can be considered as complementary modalities whose fusion can lead to a better accuracy in a classification process. So we have also compared two fusion strategies in our image categorization experiments, namely early

fusion strategy by grouping all the features together to fed a single classifier, and late fusion strategy that makes use of "channels" with a separate classifier for each kind of features, the outputs of these classifiers being merged later [Snoek *et al.* 2005]. RCM and RHS have first been merged by the strategies of Early Fusion and Late Fusion, noted as EF(RCM+RHS) and LF(RCM+RHS), and then SIFT has been combined to obtain EF(RCM+RHS+SIFT) and LF(RCM+RHS +SIFT).

Finally, one-against-all multilayer perceptron has been built on a balanced dataset for each class with a 4-fold cross-validation, for its ability to draw complex separating class borders. The structure of these perceptrons is composed of one hidden layer for all the experiments, and the number of neurons in the hidden layer that varies according to the number of inputs can have three different values: 5, 15, 2 for single channel, early fusion and late fusion respectively. The degree of the polynomial for modeling the visual content of an image has been empirically set to 8. The performance of the evaluated methods has been measured through three classical rates, namely classification rate, recall rate and precision rate. The detailed results are presented in Table 2.3, Table 2.4, Table 2.5 respectively.

Table 2.3: Classification rate obtained for 5 representative classes

Classification rate	Plane	Bicycle	Bus	Horse	Person
SIFT	65.0%	55.2%	60.8%	65.5%	58.9%
RCM	72.7%	61.6%	67.9%	65.8%	62.8%
RHS	76.6%	62.0%	66.1%	62.6%	63.5%
EF(RCM+RHS)	80.3%	64.0%	70.8%	65.6%	65.2%
EF(RCM+RHS+SIFT)	81.5%	64.6%	69.3%	66.4%	65.5%
LF(RCM+RHS)	82.0%	71.0%	92.0%	79.7%	66.7%
LF(RCM+RHS+SIFT)	85.2%	72.7%	92.7%	81.5%	69.4%

In these result tables, experimented classifiers can be categorized into 3 classes: Single Channel (SC) which means make use of only one kind of features, Early Fusion (EF) and Late Fusion (LF). As we can see, our region-based features, RCM and RHS, with an improvement of 5 points in average, perform better than SIFT features. These results tend to show the effectiveness of our RCM and RHS features using the polynomial modeling based image representation. Between RCM and

Table 2.4: Recall rate obtained for 5 representative classes

Recall rate	Plane	Bicycle	Bus	Horse	Person
SIFT	68.7%	57.9%	62.6%	71.6%	60.9%
RCM	73.5%	64.1%	68.1%	66.5%	67.3%
RHS	76.6%	68.3%	71.6%	67.1%	69.1%
EF(RCM+RHS)	80.2%	65.7%	70.9%	67.1%	68.4%
EF(RCM+RHS+SIFT)	81.4%	66.7%	70.5%	70.1%	68.6%
LF(RCM+RHS)	84.2%	73.9%	89.4%	79.5%	70.0%
LF(RCM+RHS+SIFT)	85.4%	74.9%	89.8%	83.9%	72.9%

Table 2.5: Precision rate obtained for 5 representative classes

Precision rate	Plane	Bicycle	Bus	Horse	Person
SIFT	64.0%	54.9%	60.4%	63.8%	58.6%
RCM	72.4%	61.0%	67.9%	65.6%	61.7%
RHS	76.6%	60.6%	64.5%	61.5%	62.1%
EF(RCM+RHS)	80.4%	63.5%	70.7%	65.1%	64.2%
EF(RCM+RHS+SIFT)	81.5%	64.0%	68.8%	65.3%	64.6%
LF(RCM+RHS)	80.7%	69.8%	94.3%	79.7%	65.7%
LF(RCM+RHS+SIFT)	85.1%	71.8%	95.4%	80.1%	68.1%

RHS, we find that RHS is slightly better after comparing all 3 rates and RCM tends to favor negative side. Now focusing on EF and LF, we can note that the best classification rates are obtained when the 3 channels are merged using LF strategy which performs much better than SC and EF. The classes bus and horse, for instance, record a classification rate increase by about 22 points and 15 points respectively compared to the second higher rate obtained with EF. This result seems to suggest that the three different channels carry complementary visual information to describe the image content and their fusion helps to improve the final classification accuracy. Another reason might be that EF may suffer from conflicts between different features, leading to a blurring of the boundary between classes. This can also explain that why EF performs only slightly better than SC and much worse than LF.

Encouraged by these promising results using PMIR, we have then evaluated its

efficiency using the recommended evaluation criterion of Pascal challenge, i.e. Average Precision (AP). As a measure of classification efficiency, AP represents the average of precisions over the entire range of recalls. A good score of AP requires both high recall and high precision, which is particularly interesting for classification problems. All the experimental configurations have been conserved except the technique of cross-validation. This time, we have trained one-against-all multilayer perceptrons on the balanced dataset of each class, combined with late fusion strategy for its effectiveness shown in the previous experiments, and then used this trained classifier to classify the whole set of test images. Unfortunately, we have obtained particularly low results compared to others reported in the challenge (see Table 2.10), which are shown in Table 2.6. This has motivated us to propose another image representation method which is presented in the next subsection.

Table 2.6: Average precision obtained for 5 representative classes using PMIR.

AP	Plane	Bicycle	Bus	Horse	Person
PMIR	0.138	0.076	0.080	0.201	0.518

2.4.3 SMIR: a Statistical Measures based Image Representation

As PMIR failed to get reasonable results on Pascal 2007 dataset, we propose here a simpler and more computational efficient image representation inspired by some principles of PMIR, called Statistical Measures based Image Representation (SMIR) [Fu *et al.* 2010]. Some dimensionality reduction methods as well as several classification techniques have also been evaluated with SMIR in order to find a satisfying combination of these different components allowing to achieve a good score in terms of AP.

2.4.3.1 SMIR principle

The basic idea of SMIR is to model the distribution of values for each component of the feature vectors by descriptive statistical measures instead of a polynomial modeling as in PMIR, and then to concatenate these statistical measures into one

new single feature vector that will characterize the visual content of an image and will be used for object categorization in the next step.

Table 2.7: Descriptive statistical measures used in SMIR

Name of statistics	**Description or formula**
Arithmetic average	$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
Harmonic mean	$m = n/\sum_{i=1}^{n} \frac{1}{x_i}$
Trimmed mean	mean of X excluding the highest and lowest 10% of observations
Range	max(X)-min(X)
Mean absolute deviation	$y = \frac{1}{n}\sum_{i=1}^{n} \lvert x_i - \bar{x}\rvert$
Standard deviation	$s = (\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2)^{\frac{1}{2}}$
Percentiles	quantiles of X with orders that are multiples of 0.25 (5 values obtained in the interval [0, 1])

Totally 12 statistical measures have been used to describe the distribution of data for each component, among which stands the number of zeros. Indeed, due to the computation process of our visual features as well as the one of SIFT features, the data contains a high number of zeros that may disturb the computation of the data distribution. Thus, this information is carried in the feature called "number of zeros" and then zeros are removed from the new data that is characterized by the remaining 11 statistical measures which mainly belong to 3 groups: 1, Measures of central tendency to locate a distribution of data along an appropriate scale; 2, Measures of dispersion to find out how spread out the data values; 3, Percentiles to provide information about the shape of data as well as its location and spread. A detailed presentation of these 11 statistical measures is given in Table 2.7, where $X = \{x_i\}, i = 1...n$ is a set of observations for one component.

After having modeled the distribution of each component of the feature set using the statistical measures, they will be concatenated to form a new image feature vector Q, which we call *Statistical Measures based Image Representation* (SMIR). This new vector may also lead to the "curse of dimensionality" problem [Bellman 1961] because its length is in the same level as the one of PMIR. Therefore, a dimension-

ality reduction method should be used. Several approaches have been evaluated for this purpose in order to identify the most appropriate one for SMIR. This will be discussed in next subsection.

2.4.3.2 Experimental results

The classification schemes proposed so far in the literature for automatic generic visual object categorization often suffer from the problem due to a small and biased training dataset, in particular with an unbalanced ratio of positive versus negative samples. Thus, contrary to the limited experiments driven for PMIR where the dataset is balanced and only one dimensionality reduction method is used, we would like to evaluate in the experiments for SMIR various classification schemes as well as different dimensionality reduction methods. These are presented in the following section followed by the corresponding experimental results.

Classification schemes The classification process, in the context of visual object categorization, aims at predicting whether at least one or several objects of some given classes are present in an image. The elaboration of such classification schemes is generally empirical as its efficiency will depend on numerous factors such as the nature of visual features used to carry the information in images, the high dimensionality of the distribution of these features and the complexity of the frontiers between classes in the feature space. Thus, we present here several classification schemes representing a general overview of conceivable classification techniques that will be further evaluated for visual object categorization purposes.

Recall the general classification process: given an image to classify, we first detect points of interest or regions from which the visual features are extracted. These features are then transformed to form a new feature vector through statistical measures based image representation using the method introduced in 2.4.3.1. Finally, this new feature vector will pass through the classifier beforehand trained or pass through a set of classifiers, according to the fusion strategy, to judge whether this image contains or not a given object. In this procedure, two particular problems should be taken into consideration. The first one is that only a biased dataset

(usually there are much more negative samples than positive ones) may be available during the training stage, especially when a one-against-all strategy is used for multi-class classification. Such an unbalanced dataset generally leads to a decrease of the classifier performance as the training set has to be as representative as possible. As a result, we have envisaged three principal ways to address this problem: 1, the simplest one is to construct a balanced dataset using only a subset of negative samples through sub-sampling (randomly for example); 2, a series of classifiers is built up according to a cascade technique, all classifiers having at their disposal balanced dataset created using different samplings; 3, the "weak" side of the dataset is compensated by giving it a higher weight during the training. The second issue is the dimensionality reduction method which aims at reducing effectively the feature vector dimension in order to avoid the potential "curse of dimensionality" while keeping its discrimination ability. In the following experiments, four different solutions are considered in order to evaluate their respective efficiency for our image categorization problem: 1, no dimensionality reduction method is used; 2, a canonical discriminant analysis [Fisher 1936] is used; 3, a principal component analysis [Pearson 1901] [Jolliffe 2002] is used; 4; an adaboost algorithm is used [Freund & Schapire 1999] [Freund & Schapire 1997] [Shen & Bai 2004]. A brief introduction of all these techniques is given in the following paragraphs.

- **Balanced classifier:** In this case, a subset of negative samples is chosen through random sampling. Its size is equal to the one of the positive sample set.

- **Cascade of classifiers:** This is a series of balanced classifiers in each of which the positive samples are always the same whereas the negative samples are composed of the false positives of the previous balanced classifier and new added negative samples until the two sides reach a new balance (see Figure 2.13). The process terminates when no more new negative sample are left. The final score is the sum of the scores given by each balanced classifier.

- **Biased classifier:** This corresponds to a single global classifier which is trained using all available samples. However, in order to handle the unbal-

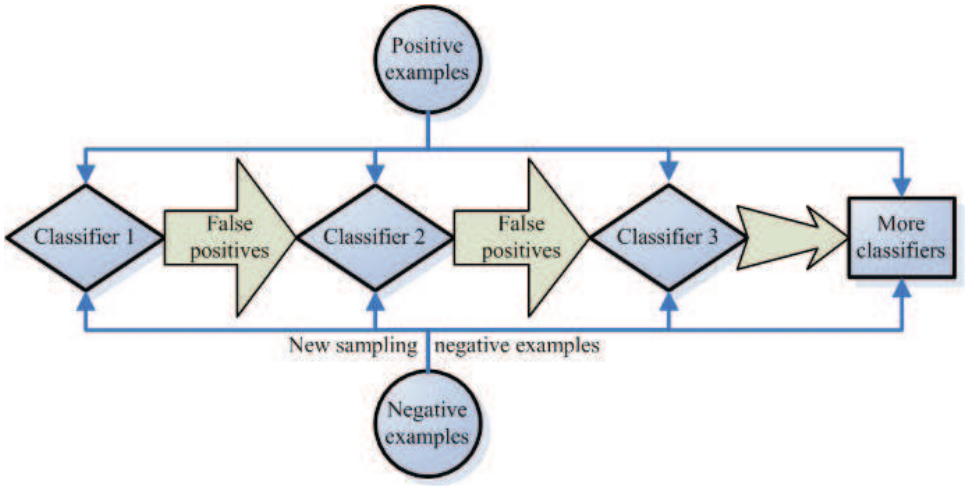

Figure 2.13: Illustration of the cascade of classifiers.

anced effect of the dataset, different weights are given to the positive and negative samples. As weight values are classifier and dataset dependent, they are determined experimentally.

- **Principal Component Analysis (PCA):** It is a simple, widely-used and non-parametric method for extracting relevant information from confusing dataset. With minimal additional effort PCA provides a roadmap for how to reduce a complex dataset to a lower dimension to reveal the sometimes hidden, simplified structure that often underlie it, that is to say it transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal component.

- **Canonical Discriminant Analysis (CDA):** It is a quick algorithm which allows reducing the dimension by producing a new representation space which distinguishes the best the different classes. Its principle is to produce a series of uncorrelated discriminative variables, in order to have individuals in the same class projected on these axes as close as possible and individuals from different classes as distant as possible. In most cases, $K-1$ axes are obtained where K is the number of classes. This method has been used previously for PMIR.

- **ADAboost algorithm (ADA):** The adaboost algorithm is presented in section 2.2.3.2 as a classifier. However, in our approach here, we use it as a dimensionality reduction method since each weak classifier can also be seen as a selected single feature which best separates positive and negative samples. Thus, after T rounds, the best T features for the classification have been selected, and they can feed other classifiers, such as SVM or Neural Networks.

Implementation Concerning the classifier, we have chosen the popular SVM, presented in section 2.2.3.2, for its high ability in solving the small dataset, nonlinear and high dimensional pattern recognition problems (LIBSVM package [Chang & Lin 2001] is employed here). However, the choice of the kernel and its parameter optimization are two crucial aspects for object categorization using SVM. According to [Chang & Lin 2001], 3 reasons have encouraged us to use the Radial Basis Function (RBF) kernel. The first reason is that the RBF kernel has similar performances as the linear kernel [Keerthi & Lin 2003] or the sigmoid kernel [Lin & Lin 2003] for certain parameters. Secondly, its small number of hyperparameters facilitates the following parameter optimization task. Finally, it has less numerical difficulties.

We have performed SVM parameter optimization thanks to a grid search using a 4-fold cross-validation technique in order to find out the best-fit group of parameters (C, γ), where $C > 0$ is the penalty parameter of the error term. This parameter also offers the possibility to construct a biased classifier mentioned in 2.4.3.2 by giving different weights on C for the positive and negative side. A good estimation of the weights has been obtained according to (2.63) through several preliminary experiments, where w_{pos} and w_{neg} are the weights applying on C for the positive and negative side respectively, p and n are the number of positive and negative samples.

$$w_{pos} = (p+n)/p \qquad w_{neg} = (p+n)/n \tag{2.63}$$

One-against-all SVM classifiers have been built for each class and evaluated in terms of AP. In order to save computation time, the 4 dimensionality reduction

approaches presented before have only been applied on the balanced classifier, and the best approach has then be used with the cascade of classifiers and the biased classifier.

Results We have used in these experiments the Pascal challenge 2007 image dataset ([Everingham *et al.* 2007]), which has also been used for evaluating PMIR in section 2.4.2.3. However, in this case the whole set of test images has been considered (4952 images) to evaluate the different approaches designed to handle unbalanced data.

As the experiments of PMIR have proven the effectiveness of our proposed features, namely RCM and RCS, and their complementarity to SIFT, the same set of these three types of features have been considered here. Moreover, 2 fusion strategies, early and late (noted respectively as EF and LF), have also been evaluated together with the 4 dimensionality reduction approaches (noted as NON when no dimensionality reduction approach is used, PCA, CAD and ADA) using these feature sets with the balanced classifier, in order to evaluate their efficiency in our case of visual object categorization. Finally, the number of features for 3 channels SIFT, RCM and RHS is respectively 1536, 432 and 1152 after the modeling by statistical measures without dimensionality reduction, which is the case in NON. ADA selects the best 50% of the original features in NON sorted according to adaboost algorithm for all the 3 channels. However, PCA and CDA would greatly reduce this number to about a few tens.

From Table 2.8, which shows the results for 5 representative classes using the combinations of 2 fusion strategies and 4 dimensionality reduction approaches with a balanced classifier, we can see that NON generally performs best among all the 4 dimensionality reduction approaches, even if results of ADA are somewhat comparable. However, PCA and CDA seriously hurt the performance in our case. Considering the number of features in different dimensionality reduction approaches as well, we found that the approaches that have a huge number of features (for example, EF_NON has 1536+432+1152=3120 features) generally perform better than the ones having a small number of features. This fact is probably due to the boundary

Table 2.8: Average precision for 5 representative classes using the combinations of 2 fusion strategies and 4 dimensionality reduction approaches with a balanced classifier.

AP	Plane	Bicycle	Bus	Horse	Person
LF_NON	0.409	0.193	0.192	0.330	0.722
EF_NON	**0.423**	**0.252**	**0.281**	0.386	**0.750**
LF_PCA	0.405	0.135	0.109	0.192	0.708
EF_PCA	0.374	0.210	0.215	0.225	0.725
LF_CDA	0.199	0.077	0.058	0.097	0.549
EF_CDA	0.188	0.089	0.054	0.219	0.545
LF_ADA	0.348	0.187	0.095	**0.404**	0.695
EF_ADA	0.415	0.237	0.223	0.373	0.736

blurring between classes occurring when realizing the transformations of PCA and CDA. Then focusing on LF and EF, the results show that EF performs better than the second fusion strategy. One of the reasons might be the good ability of SVM in solving high dimensional problems so that it benefits EF in which all the features are merged to form a long feature vector. This conclusion is also consistent to the fact observed previously when comparing different dimensionality reduction approaches. As a result, early fusion together with no dimensionality reduction will be applied on the cascade of classifiers and biased classifier, whose results are listed in Table 2.9.

Table 2.9: Average precision for 5 representative classes using early fusion with balanced classifiers, cascades of classifiers and biased classifiers.

AP	Plane	Bicycle	Bus	Horse	Person
EF_Balanced	0.423	0.252	0.281	0.386	0.750
EF_Cascade	0.504	0.287	0.303	0.453	0.750
EF_Biased	**0.517**	**0.351**	**0.318**	**0.585**	**0.755**

In Table 2.9, EF_Cascade and EF_Biased get an AP much higher than EF_Balanced for all the classes. An increasing of 13% to 51% can be observed between EF_Biased and EF_Balanced, depending on the class except "person" in which only 1.41% augmentation has been observed. An explanation consists in the

fact that persons appear in almost all the training images so that the training set of EF_Balanced doesn't differ very much from the other two. Until now, we have got the best results using EF_Biased which are comparable to some of results reported in [Everingham *et al.* 2007], shown in Table 2.10. But we are also conscious that there is still a relatively large gap to the best results, meaning that much efforts are always needed to across it in the future.

Table 2.10: Average precision for 5 representative classes reported in the Pascal challenge 2007, extracted from the site of [Everingham *et al.* 2007].

AP	Plane	Bicycle	Bus	Horse	Person
INRIA_Larlus	0.626	0.540	0.464	0.660	0.772
INRIA_Flat	0.748	0.625	0.604	0.765	0.845
INRIA_Genetic	0.775	0.636	0.606	0.775	0.859
MPI_BOW	0.589	0.460	0.405	0.636	0.757
PRIPUVA	0.486	0.209	0.142	0.301	0.620
QMUL_HSLS	0.706	0.548	0.511	0.715	0.806
QMUL_LSPCH	0.716	0.550	0.511	0.715	0.808
TKK	0.714	0.517	0.499	0.726	0.822
ToshCam_rdf	0.599	0.368	0.333	0.639	0.779
ToshCam_svm	0.540	0.271	0.223	0.480	0.781
Tsinghua	0.629	0.424	0.407	0.650	0.769
UVA_Bigrams	0.612	0.332	0.376	0.616	0.746
UVA_FuseAll	0.671	0.481	0.463	0.698	0.794
UVA_MCIP	0.665	0.479	0.440	0.664	0.786
UVA_SFS	0.663	0.497	0.449	0.715	0.804
UVA_WGT	0.597	0.337	0.329	0.651	0.742
XRCE	0.723	0.575	0.575	0.757	0.840

The improvement recorded between single channels and early fusion in Table 2.11 means that our region based features managed to extract information which is complementary to the one of SIFT features so that the fusion of these single channels helps to improve the classifier performance. This conclusion is also consistent to the one drawn from the experiments of PMIR. Among single channels, their performances are more or less the same using statistical measures based image representation, but vary significantly from one class to another.

Table 2.11: Average precision for 5 representative classes between single channels (SIFT, RCM, RHS) and early fusion with biased classifiers.

AP	Plane	Bicycle	Bus	Horse	Person
SIFT	0.402	0.212	0.181	0.352	0.652
RCM	0.443	0.209	0.174	0.427	0.651
RHS	0.307	0.179	0.213	0.298	0.656
EF	**0.517**	**0.351**	**0.318**	**0.585**	**0.755**

2.4.4 Conclusion on image representation

In this section, we have mainly worked with image representations which consist in modeling efficiently the visual content of an image after having extracted features, especially our proposed region based features and SIFT. PMIR has been firstly proposed and evaluated on a balanced subset of Pascal 2007 dataset, together with two widely used fusion strategies, namely early fusion and late fusion. Experimental results have shown us the promising performance achieved by PMIR and the complementarity of information carried by our region based features and SIFT. However, It could not persist its success when being evaluated on the whole test set of Pascal 2007 dataset, thus inducing us to consider SMIR. This time, a set of different classification schemes and dimensionality reduction techniques has been considered as well, in order to find a best pair of them to work with SMIR. Moreover, two concurrent fusion strategies, early and late fusion, have also been studied. Experiments carried out on the same dataset as PMIR have revealed that good classification results can be obtained, which is comparable to the results reported in the Pascal challenge, and the fact that our region based features carry complementary information to SIFT has been proven again.

2.5 Conclusion

We have presented in this chapter the three principal stages of a typical visual object categorization system, namely feature extraction, selection and image representation. Based on the well-known feature selection method SFS, a novel embedded

feature selection approach, called ESFS, has been first introduced. It relies on the simple principle to add incrementally most relevant features and merge them in an embedded way thanks to the concept of combined mass functions from the evidence theory which also offers the benefit of obtaining a computational cost much lower than the one of original SFS. Experimental results have shown that selecting relevant features improves the classification accuracy, and for this purpose, ESFS, used as a filter selection method, performs better than widely used state of the art approaches such as Fisher and PCA for the filter methods and SFS, SFFS and OS for the wrapper approaches. Moreover, ESFS can be used not only as a feature selection method, but also directly as a classifier.

We envisage in our future work to investigate alternative solutions for building mass functions associated to each single feature within ESFS. Indeed, for the moment masses are distributed on single classes for a given feature. However, the evidence theory allows the reasoning on union of classes, which may be more accurate. Moreover, an interesting issue would be to integrate into the feature selection process the conflict information that can be obtained from combined mass functions and which may allow to avoid combining features that give contradictory information. Indeed, even if several fusion operators we considered integrate the notion of conflict, such as the one of Dempster and Yager, their performance has not been significantly improved compared to the performance of TBM which does not handle the conflict. Therefore further research is needed in order to integrate the conflict information in a more efficient way.

Concerning image representation, we have also proposed two methods for visual object categorization. The first one consist in using polynomial modeling based image representation with our proposed new region based features, which circumvent some drawbacks of the popular "bag of features" approach, especially the difficulty of fixing the size of visual vocabulary. Two different fusion strategies, early and late, have been considered to merge information from different "channels" represented by the different types of features. Results on a subset of Pascal 2007 dataset have shown that good performance can be achieved with our approach and that our segment features carry information which is complementary to SIFT features.

However, faced with unreasonable results obtained in the evaluation on the whole test set of Pascal 2007 dataset using PMIR, we have later presented a deeper evaluation of different classification schemes leading to the proposition of another novel approach for visual object categorization, using statistical measures based image representation which is inspired by the same principle as PMIR where the polynomial modeling of the feature distribution is replaced by computational more efficient statistical measures. Thus, an evaluation of several dimensionality reduction methods and classifier construction techniques facing unbalanced dataset has also been carried out. Moreover, two concurrent fusion strategies, early and late fusion, have been studied as well. Experiments performed on Pascal 2007 dataset have drawn the same conclusion as in the case of PMIR: a good classification accuracy, which is comparable to the results reported in the challenge, can be achieved with the image representation we propose and our region based features associated with popular SIFT features allow to improve the classification accuracy.

Although the choice of fusion strategy remains difficult and unclear, as it depends significantly on the features and classifier used, the fact that the fusion of different types of features can effectively improve the classification performance has been confirmed in both of experiments using PMIR and SMIR, encouraging us to consider more features and fuse them in an intelligent way for building future VOC systems.

Chapter 3

Sparse representation for VOC

Contents

3.1 Introduction

Sparse representation model of signals have received a lot of attentions and is a very active research area in recent years. It is originally used as a powerful tool

for acquiring, representing and compressing high-dimensional signals in the signal processing applications and has achieved great successes. These successes are mainly due to the fact that important classes of signals have naturally sparse representations with respect to fixed bases, or concatenations of such bases. Moreover, a set of efficient and effective algorithms based on convex optimization or greedy pursuit has been proposed for solving the sparse representation problem and computing such representations with high fidelity [Bruckstein *et al.* 2009].

In such a context, we present in this chapter our approaches inspired by the principles of sparse representation theory that we have adapted to the problem of VOC.

3.2 Literature review

The goal of sparse representation is to obtain a compact high-fidelity representation of a given signal, which can be considered as a linear combination of atoms from an overcomplete dictionary [Mallat & Zhang 1993]. The property of sparsity in the representation of signals has also been approved in human perception by some studies of human vision [Olshausen & Field 1996] [Olshausen & Field 1997]. In fact, many neurons in the visual pathway are selective for a variety of specific stimuli in the human vision and then can be considered as an overcomplete dictionary. Thus, the firing of the neurons with respect to a given input image is typically highly sparse. Recent research on wavelet, ridgelet, curvelet and contourlet transforms has also greatly accelerated and promoted the development of sparse representation model. Until now, it has been widely used and obtained promising results in many different applications, such as signal separation [Starck *et al.* 2005], denoising [Elad & Aharon 2006], coding [Olshausen *et al.* 2001], image inpainting and restoration [Mairal *et al.* 2008c] and magnetic resonance spectroscopy quantification [Guo *et al.* 2010].

Recently techniques from sparse signal representation have significantly impacted the domain of computer vision and pattern recognition [Wright *et al.* 2009a] [Wright *et al.* 2009b] [Mairal *et al.* 2008a], in which we are often more interested in

extracting the visual content of an image rather than a compact high-fidelity representation. Variations and extensions of ℓ^1-minimization have been widely used in many vision tasks, including face recognition [Wright *et al.* 2009b], image super-resolution and classification [Yang *et al.* 2008a] [Mairal *et al.* 2008a], motion segmentation [Rao *et al.* 2008], background modeling [Dikmen & Huang 2008]. In almost all of these applications, the sparse representation based methods has provided encouraging results which are comparable to the state of the art ones. This has motivated us to propose approaches adapting these principles to the problem of VOC.

Before presenting our proposed approaches, we would like first of all to give a brief introduction of sparse representation model below, followed by the related works which consider images as signals to be processed.

3.2.1 Sparse representation model

Let consider a signal $y \in \mathbb{R}^n$, which will be represented as a linear combination of basic elements from a dictionary $D \in \mathbb{R}^{n \times K}$ composed by atoms in columns $\{d_j\}_{j=1}^K$. We say that a representation of the signal y based on this specific dictionary D is any vector $x \in \mathbb{R}^K$ which satisfies:

$$y = Dx \tag{3.1}$$

In the case where $n < K$, the dictionary D is said to be overcomplete and this equation is underdetermined thus having many possible solutions. Conventionally, in this case, the minimum ℓ^2 norm solution is chosen:

$$\min_x(||x||_2) \ \ subject \ \ to \ \ Dx = y \tag{3.2}$$

where $||x||_2$ is the ℓ^2 norm of x. The above problem can easily be solved and it has a unique solution as follow:

$$x = D^+ y = D^T (DD^T)^{-1} y \tag{3.3}$$

where D^+ is the pseudoinverse of D. However, this solution is generally non sparse with many nonzero elements corresponding to the atoms from the dictionary and consequently does not satisfy our expectation. Indeed, we would rather prefer a sparse solution, that is to say we want to find a linear combination of only a few atoms to approximate the signal y. This problem can be formally described by

$$\min_x(||x||_0) \ \ subject \ \ to \ \ Dx = y \tag{3.4}$$

where $||x||_0$ is ℓ^0 norm of x and equals the number of nonzero elements in the vector x. Solving the equation (3.4) is a NP hard problem because of its nature of combinational optimization. Nevertheless, there exist many approximation techniques for this task such as Matching Pursuit (MP) [Mallat & Zhang 1993] which consists in selecting one atom at each stage based on the minimization of the residue in a greedy way, and Orthogonal Matching Pursuit (OMP) [Pati *et al.* 1993]. If the dictionary is an orthogonal vector set and the signal is indeed a sparse combination of atoms, OMP is guaranteed to find this sparse set.

Another way to address the problem of (3.4) is to replace the ℓ^0 norm minimization by ℓ^1 norm minimization:

$$\min_x(||x||_1) \ \ subject \ \ to \ \ Dx = y \tag{3.5}$$

where $||x||_1$ is the ℓ^1 norm of x. As in several recent works [Donoho & Huo 2001] [Donoho 2004], it is proved that if certain conditions on the sparsity are satisfied, i.e. the solution is sparse enough, then these two norm minimization problems are equivalent. As (3.5) is a convex optimization problem, it has a unique solution and can be efficiently solved by standard linear programming methods such as Basis Pursuit (BP) [Chen *et al.* 1998]. The main drawback of BP algorithm is that it is extremely time-consuming, especially for the image processing. Thus, numerous other methods have been proposed for ℓ^1 norm minimization problem due to its wide range of possible applications in the domain of statistics and signal processing such as LARS/LASSO [Tibshirani 1996], Homotopy [Malioutov *et al.* 2005], GPSR

[Figueiredo *et al.* 2007], L1-Ls [Kim *et al.* 2007], IST [Daubechies *et al.* 2004] etc. Between ℓ^0 norm and ℓ^1 norm, the focal underdetermined system solver (FOCUSS) is proposed [Gorodnitsky & Rao 1997], using the ℓ^p norm with $0 < p \leq 1$ to replace ℓ^0 norm. Here, for $p < 1$, the similarity to the true sparsity measure is better but the overall problem becomes nonconvex, giving rise to local minima that may mislead in the search for solutions.

The OMP algorithm involves the computation of inner products between the signal and dictionary columns. It is very simple to be implemented and fast to be executed while keeping good performances. Therefore, numerous works rely on it. It is also the case for our experiments where we have made use of OMP to perform the sparse coding, i.e. computing the sparse coefficients x of signal y given a dictionary D. The principle of OMP algorithm [Blumensath & Davies 2007] is as follows:

Algorithm: Orthogonal Matching Pursuit (OMP)

- Task: Given the dictionary $D \in \mathbb{R}^{n \times K}$ and the signal y to be represented by a linear combination of atoms from D, find the corresponding coefficients x so that Dx best approximates y.

- Initialization: Set the initial residual $r^0 = y$, the initial index set $\Gamma^0 = \emptyset$, $s^0 = 0$. Set the indicator of iteration $t = 1$.

- Repeat until stopping rule (usually the number of atoms used):

  - $\alpha_i = d_i^T r^{t-1}$ for all $i \notin \Gamma^{t-1}$ and $i \in \{1, 2, ..., K\}$
  - $i_{max} = arg_i max|\alpha_i|$
  - $\Gamma^t = \Gamma^{t-1} \cup i_{max}$
  - $s^t_{\Gamma^t} = D^+_{\Gamma^t} y$ where D_{Γ^t} is a reduced dictionary composed by the columns in D whose indices are in Γ^t
  - $r^t = y - D s^t_{\Gamma^t}$
  - $t = t + 1$

- Calculate the sparse coefficients $x = D^+_{\Gamma^t} y$.

Another crucial aspect for applying sparse representation model successfully on the signals (images) is the design of the dictionary, namely D in the equation (3.1). One type of approaches consists in using the preconstructed dictionaries which do not change during the problem solving. Such dictionaries based on the transforms mentioned above, i.e. ridgelet, curvelet and contourlet, have been widely used in signal processing. Another possibility consists in using the dictionary composed by the training images themselves, which has also given promising results as in [Wright *et al.* 2009b] and [Fu *et al.* 2009b].

However, this conventional setting may not be suitable to be directly employed in the domain of computer vision and pattern recognition as there is no given basis with good property compared to signal processing [Wright *et al.* 2009a]. In order to address this new situation, another type of approaches has been proposed in order to learn a task-specific dictionary from given samples by updating the dictionary, with the purpose of describing the image content more effectively. We can mention here two appealing and widely used methods: Method of Optimal Directions (MOD) [Engan *et al.* 1999] and K-SVD [Aharon *et al.* 2006]. Both of them are iterative methods, containing a sparse coding stage which finds the corresponding coefficients x of a signal y based on the current dictionary and a dictionary update stage which updates the dictionary using coefficients obtained from previous stage to better fit the data. The objective function for these two methods can be expressed as in (3.6) which is a reformulation of (3.4).

$$\min_{D,X}\{||Y - DX||_F^2\} \quad subject \quad to \quad ||x_i||_0 \leq L \quad \forall i \tag{3.6}$$

where Y is a matrix containing all the signals $\{y_i\}_{i=1}^N$ in columns and X is the corresponding coefficient matrix composed by $\{x_i\}_{i=1}^N$. The notation$||A||_F$ is the Frobenius norm, defined as $||A||_F = \sqrt{\sum_{ij} a_{ij}^2}$. L is a positive number which controls the sparsity level. As any pursuit algorithm can be used to do the sparse coding for both of them, typically OMP, their main difference lies in the dictionary update stage. Assuming that X is fixed, MOD takes the derivative of $||Y - DX||_F^2$ to get

the relation $(Y - DX)X^T = 0$, leading to

$$D^{t+1} = YX^{t^T}(X^t X^{t^T})^{-1} \tag{3.7}$$

Thus, MOD simply updates the dictionary in an entire way without changing the coefficients in this stage. On the contrary, K-SVD updates D sequentially, one column (atom) by one column, combined with an update of the sparse coefficients, thereby accelerating convergence and yielding more accurate results. So finally, K-SVD has been chosen as a dictionary update method in our experiments, whose algorithm can be described as follows [Aharon *et al.* 2006]:

Algorithm: K-SVD

- Task: Find the best dictionary to represent the signals $\{y_i\}_{i=1}^N$ as sparse compositions, by solving

$$\min_{D,X}\{||Y - DX||_F^2\} \quad subject \quad to \quad ||x_i||_0 \leq L \quad \forall i$$

- Initialization: Set the dictionary $D^0 \in \mathbb{R}^{n \times K}$ with ℓ^2 normalized columns (randomly selected from the training dataset for example). Set the indicator of iteration $t = 1$.

- Repeat until stopping rule (convergence for example):

  - *Sparse Coding Stage*: Use any pursuit algorithm (typically OMP) to compute the coefficient vectors x_i for each signal y_i, by approximating the solution of

$$\forall i = 1, 2, ..., N, \quad \min_{x_i}\{||y_i - D^{t-1}x_i||_2^2\} \quad subject \quad to \quad ||x_i||_0 \leq L.$$

  - *Dictionary Update Stage*: For each column $k = 1, 2, ..., K$ in D^{t-1}, update it by

    * Define the group of signals that use this atom $\omega_k : \{i|1 \leq i \leq N, x_T^k(i) \neq 0\}$ where x_T^k is the k-th row of X.

* Compute the overall representation error matrix, E_k, by

$$E_k = Y - \sum_{j \neq k} d_j x_T^j.$$

* Restrict E_k by choosing only the columns corresponding to ω_k, and obtain E_k^R.
* Apply SVD decomposition $E_k^R = U \Delta V^T$. Choose the updated dictionary column $\tilde{d}_k$ to be the first column of U. Update the coefficient vector x_R^k to be the first column of V multiplied by $\Delta(1,1)$. Here x_R^k is a reduced version of the row vector x_T^k by discarding of the zero entries.

- $t = t + 1$.

3.2.2 Reconstructive methods

In the standard framework of sparse representation, the objective is to reconstruct the signal using as few number of atoms as possible while minimizing the reconstruction error at the same time. The methods derived from this philosophy are called *reconstructive methods*.

[Wright *et al.* 2009b] proposes to represent the test sample using a dictionary composed by the training samples themselves. They argue that if sufficient training samples are available from each class, it will be possible to represent the test samples as a linear combination of just those training samples from the same class. So this representation is naturally sparse, involving only a small fraction of the overall training dataset. They apply this approach on face recognition that is realized according to the reconstruction errors for different categories after the sparse representation of test samples having been recovered via ℓ^1 minimization.

[Candès 2006] presents the Compressive Sensing (CS) theory by introducing a sensing matrix in the traditional sparse representation model, which shows that the signals can be recovered from far less samples than those required by the classical Shannon-Nyquist Theorem. Then in [Duarte-Carvajalino & Sapiro 2009] a framework for simultaneously learning the overcomplete non-parametric dictionary and

the sensing matrix is introduced, obtaining good results for image restoration.

[Raina *et al.* 2007] makes use of sparse representation model to learn a dictionary from the unlabeled data for a reconstruction task, without assuming that these unlabeled data follow the same category labels as the labeled data, an approach they called "self-taught learning". Then the sparse decompositions of signals are used as posteriori within a classifier.

Another ingenious approach presented in [Yang *et al.* 2009b] incorporates the classical "bag of features" model with sparse coding which is used to replace the K-means clustering algorithm. In fact, sparse coding can be viewed as a generalization of K-means by relaxing two constraints: 1, each signal is allowed to be represented by a linear combination of codewords instead of one in K-means; 2, the value of coefficients is allowed to vary instead of being fixed to 1 in K-means. So this replacement can achieve a much lower reconstruction error due to the less restrictive constraint, leading to a possible improvement of performance.

Although the reconstructive methods presented above have obtained promising results for many applications, their efficiency for the classification task is not guaranteed. Indeed the goals of reconstruction and classification are naturally different. One immediate solution to extend reconstructive approaches to the classification task may consist in learning a dictionary for one category in a reconstructive way so that the reconstruction error of a signal in this category is minimized. However, we can not ensure that the reconstruction error of a signal from a different category on this specific dictionary is bigger than the signals from the same category.

Thus, discriminative methods have been proposed to generate a signal representation that maximizes the separation of signals from different categories, being usually sensitive to corruption in signals due to lacking crucial properties for signal reconstruction. Therefore a better choice is to combine the reconstructive term and discriminative term together in the objective function of sparse representation model for classification task, thus yielding the following *reconstructive and discriminative methods*.

3.2.3 Reconstructive and discriminative methods

[Huang & Aviyente 2006] proposes to integrate a Fisher discrimination term, which tries to maximize the inter-class variance while minimize the intra-class one, to the standard reconstructive sparse representation formulation. Their approach is proved to yield robust and discriminant image representations through the experiments on synthetic signals and handwritten digits recognition task with different levels of noise. However, there is no dictionary learning in their work as they use preconstructed dictionaries and sparse coding over them. However, the actual dictionary plays a critical role, and it has been shown that learned and data adaptive dictionaries significantly outperform off-the-shelf ones. Therefore, one may prefer to learn a task-specific dictionary for classification.

In [Mairal *et al.* 2008a] multiple dictionaries are learned, one per category, so that each category dictionary provides a good reconstruction for its corresponding category and a poor one for the other categories. During the learning procedure, they introduced a softmax discriminative cost function to reconstructive sparse representation:

$$\mathcal{C}_i^{\gamma}(y_1, y_2, ..., y_N) = \log(\sum_{j=1}^{N} e^{-\gamma(y_j - y_i)}) \tag{3.8}$$

which is close to zero when y_i is the smallest value among the y_j. Increasing the value of the parameter $\gamma > 0$ provides a higher relative penalty cost for each misclassified patch whereas the final classification process itself is based on the corresponding reconstruction error, rather than exploiting the actual decomposition coefficients, which seems to be more reasonable to feed them into a discriminative classifier. Moreover, the strategy of learning one dictionary for each category requires more computational resource. The same authors investigate in another work [Mairal *et al.* 2008b] the possibility to learn simultaneously a single shared dictionary as well as multiple decision functions for different signal categories, one function for each category, instead of learning multiple dictionaries in [Mairal *et al.* 2008a].

Contrary to [Mairal *et al.* 2008a] who modifies the dictionary update stage of K-SVD, [Rodriguez & Sapiro 2007] proposes to improve the discrimination power through modifying the sparse coding stage. It is mainly based on the concept of

obtaining simultaneous sparse decompositions within each category by representing all the signals from that category at once as a linear combination of a common subset of atoms. The objective is to capture the common internal structure of these signals and eliminate their internal variation, while keeping a global discrimination term among different categories at the same time.

Contrary to these reconstructive and discriminative methods that have been designed for local image analysis, such as texture classification, digits recognition and local patch analysis, we would like to present in the following our proposed reconstructive and discriminative approach inspired by sparse representation for generic visual object categorization.

3.3 R_SROC: a Reconstructive Sparse Representation based Object Categorization

Before using directly reconstructive and discriminative sparse representation for visual object categorization, we have firstly proposed a simple preliminary reconstructive approach [Fu *et al.* 2009b], inspired by [Wright *et al.* 2009b], to evaluate the effectiveness of sparse representation model for our interested task, VOC. Assuming the intuitive hypothesis that an image could be represented by a linear combination of the training images from the same class, a sparse representation of the image is first of all obtained by solving a ℓ^1 (or ℓ^0)-minimization problem and then fed into a traditional classifier such as SVM to finally perform the classification task. Experimental results obtained on the SIMPLIcity dataset have shown that this new approach can improve the classification performance compared to standard SVM using directly features extracted from the image. The details of the approach is presented below.

3.3.1 R_SROC principle

Inspired by the principles of sparse representation, an image can be represented by a linear combination of elements from a dictionary composed of training images

themselves. Suppose that

$$\{f_{1,1}, f_{1,2}, ..., f_{i,j}, ..., f_{M,N_M-1}, f_{M,N_M}\} \in \mathbb{R}^n. \tag{3.9}$$

are feature vectors extracted from the training images for totally M categories representing the distribution of their visual content, where $N_i, i = 1, 2, ..., M$ is the number of images for the i-th object category. Then, a new image feature vector y can be expressed as follows:

$$\begin{aligned} y =& \omega_{1,1} f_{1,1} + \omega_{1,2} f_{1,2} + ... + \omega_{i,j} f_{i,j} + ... \\ &+ \omega_{M,N_M-1} f_{M,N_M-1} + \omega_{M,N_M} f_{M,N_M}. \end{aligned} \tag{3.10}$$

where $\omega_{i,j}$ is the weight for the j-th image of the i-th category. Let D be a new matrix (dictionary) built of all $N = N_1 + N_2 + ... + N_M$ training images for these M categories:

$$D = [D_1, D_2, ..., D_M] = [f_{1,1}, f_{1,2}, ..., f_{M,N_M}]. \tag{3.11}$$

and let x be a $N \times 1$ coefficient vector:

$$x = [\omega_{1,1}, \omega_{1,2}, ..., \omega_{M,N_M}]^T \in \mathbb{R}^N. \tag{3.12}$$

Then, the equation (3.10) can be rewritten using the following matrix notation:

$$y = Dx \quad \in \mathbb{R}^n. \tag{3.13}$$

If sufficient representative training images are available for each category, we can assume that the image y can be represented by a linear combination of only the training images from the same category as y. Suppose that y belongs to the i-th category, thus the coefficient vector x is supposed to have the following form:

$$x = [0, 0, ..., 0, \omega_{i,1}, \omega_{i,2}, ..., \omega_{i,N_i}, 0, ..., 0, 0]^T. \tag{3.14}$$

whose values are zero for the images that do not belong to the i-th category. We

can obviously observe that x is naturally sparse if the number of categories to be classified M is sufficiently large. For instance, in our case $M = 10$, then only 10% of the entries of x has nonzero value hence sparse. Based on this observation, finding the sparsest solution x for the equation (3.13) is equivalent to the problem of (3.5) and can be solved by numerous methods mentioned in section 3.2.1.

Once the sparse representation x of all the images has been obtained by computing a ℓ^0 norm minimization problem or the equivalent ℓ^1 norm minimization problem, they can be used to feed a traditional classifier such as Neural Networks (NN), Linear Discriminant Analysis (LDA) or SVM to perform the final classification task. The complete categorization process is described as follows:

R_SROC algorithm

1. Extract the feature vector representing the image visual content for all the training images: $f_{i,j} \in \mathbb{R}^n, i \in \{1, 2, ..., M\}, j \in \{1, 2, ..., N_i\}$, where M is the number of categories and N_i is the number of training images for i-th category. (For example, suppose we have 3 categories and there are 9, 10, 11 training images in each category respectively. The feature vector representing the image has a dimension of 10. Thus, we have in this case $M = 3$, $N_1 = 9$, $N_2 = 10$, $N_3 = 11$, $n = 10$.)

2. Regroup all these feature vectors to build the dictionary $D = [D_1, D_2, ..., D_M] = [f_{1,1}, f_{1,2}, ..., f_{M,N_M}] \in \mathbb{R}^{n \times N}$ where $N = \sum_{i=1}^{M} N_i$ is the total number of training images. (Retaking the previous case, we get $N = 30$ and D is a 10×30 matrix while D_1, D_2, D_3 are respectively 10×9, 10×10, 10×11 sub-matrices.)

3. Normalize the columns of D to have unit ℓ^2 norm.

4. Solve the ℓ^1 norm minimization problem to obtain the sparsest solution x for the equation $y = Dx$:

$$\min_x(||x||_1) \ \ subject \ \ to \ \ Dx = y.$$

where y is the image for which we want to obtain its sparse representation

Figure 3.1: Some sample images from SIMPLIcity dataset (from left to right, from top to bottom, they belong to African & village, Beach, Building, Bus, Dinosaur, Elephant, Flower, Horse, Mountain & glacier and Food respectively).

for classification. (Or alternatively, solve the ℓ^0 norm minimization problem: $\min_x(||x||_0)$ *subject to* $Dx = y$.)

5. Feed the obtained sparse representation of images x as input of a classifier (SVM in our case).

6. Assign the category label to images according the output of the classifier.

3.3.2 Experimental results

Our experiments using R_SROC are performed on the SIMPLIcity dataset [Wang *et al.* 2001b] with the whole ten categories. They are: African & village, Beach, Building, Bus, Dinosaur, Elephant, Flower, Horse, Mountain & glacier and Food. Thus, a total of 1000 images from these 10 categories has been used. Half of the images are used for training and another half for test, these two subsets being chosen randomly. Some sample images are presented in Figure 3.1.

A total number of 2446 features has been computed to represent each image from SIMPLIcity dataset. The corresponding feature set includes Color Auto-Correlogram (CAC), Color Coherence Vectors (CCV), Color Histogram (CH), Color Moments (CM), Edge Histogram (EH), Grey Level Co-occurrence Matrix (GLCM), Texture Auto-Correlation (TAC) and Local Binary Pattern (LBP). Compared to the feature set we have used in section 2.3, CH [Swain & Ballard 1991] and LBP [Takala *et al.* 2005] have been added here for their good performance in [Zhu *et al.* 2010].

Table 3.1: Classification Rate (CR) for visual object categorization on SIMPLIcity using SVM.

Class	C1	C2	C3	C4	C5
CR	80%	82%	62%	84%	100%
Class	C6	C7	C8	C9	C10
CR	86%	84%	98%	72%	86%
Average CR	83.4%				

Table 3.2: Classification Rate (CR) for visual object categorization on SIMPLIcity using R_SROC.

Class	C1	C2	C3	C4	C5
CR	84%	74%	84%	98%	100%
Class	C6	C7	C8	C9	C10
CR	86%	90%	98%	72%	88%
Average CR	87.4%				

As we have mentioned in section 3.2.1, OMP algorithm [Pati *et al.* 1993] has been chosen for obtaining the sparse representation of the images because of its efficiency and rapidity. Concerning the classifier, we have chosen the multi-class SVM (C-SVC in LIBSVM package [Chang & Lin 2001]) with RBF kernel to perform one step global classification. SVM parameter optimization task has been done thanks to a grid search using 4-fold cross-validation technique within the training set, the same as in section 2.4.3.2.

Two experiments have been carried out in our work. In the first one, we have used SVM directly on the feature vectors extracted from images to classify a test image into the corresponding category according to the object it contains. The detailed results are shown in Table 3.1 where $C_i, i \in \{1, 2, ..., 10\}$ represent i-th class with respect to the order used to present 10 classes in the previous paragraphs (the same for Table 3.2). In the second experiment, we have first computed the sparse representation of images according to the algorithm presented in the previous section and then used these sparse representations to feed SVM classifiers to perform the classification task. The detailed results are givne in Table 3.2. The classification rate has been employed to measure the performance of the classifier.

From these 2 tables, focusing on the average classification rate for 10 categories

at first, we can clearly see that our R_SROC performs significantly better than the traditional method using SVM, and presents an improvement of 4%. We should also notice that the powerful SVM has already obtained a relatively high classification rate, so the superiority of 4% achieved by R_SROC is obvious considering the relative small improvement space. Then, when going further into the results of each category, we can notice that R_SROC has enhanced the classification performance for almost all 10 categories, and especially for C3 and C4, i.e. Building and Bus, which present a large improvement. The only category that has been degraded is C2 Beach. However, the level of degradation is much lower compared to the level of improvement for other categories. Given the above observations, we can conclude that using a sparse representation of images thanks to R_SROC allows to improve the classification compared to a standard approach where the image features would have been used directly to feed SVM classifiers.

3.4 RD_SROC: a Reconstructive and Discriminative Sparse Representation based Object Categorization

Encouraged by the promising results obtained using R_SROC, we have decided to go further into the direction of this sparse representation based visual object categorization. Thus, we have proposed an approach based on a reconstructive and discriminative sparse representation for VOC, called RD_SROC. In this section, we will first formulate the problem mathematically and then propose the corresponding algorithm to solve it. Then, the evaluation of the corresponding RD_SROC approach will be presented.

3.4.1 RD_SROC principle

Recall the notation: we have a set of N training signals $\{y_i\}_{i=1}^{N}$ belonging to M categories. $Y = [y_1, y_2, ..., y_N]$ is a signal matrix with the corresponding sparse coefficients based on the dictionary D as $X = [x_1, x_2, ..., x_N]$. Moreover, we suppose that N_i signals are in the category M_i, for $1 \leq i \leq M$.

The objective function of the standard reconstructive sparse representation can

be expressed as in (3.6):

$$\min_{D,X}\{||Y - DX||_F^2\} \quad subject \quad to \quad ||x_i||_0 \leq L \quad \forall i \tag{3.15}$$

If we integrate the sparsity constraint into the function, it can be reformulated as:

$$\begin{aligned} &\min_{D,X,\Lambda}\{\lambda_1||Y - DX||_F^2 \ + \lambda_2 \sum_{i=1}^{N} ||x_i||_0\} \\ \Rightarrow &\min_{D,X,\Lambda}\{\lambda_1 \sum_{i=1}^{N} ||y_i - Dx_i||_2^2 \ + \lambda_2 \sum_{i=1}^{N} ||x_i||_0\} \end{aligned} \tag{3.16}$$

where $\Lambda = \{\lambda_1, \lambda_2\}$ is a set of regularization parameters which adjust the tradeoff between the reconstruction error and the sparsity.

The main goal of our approach is to learn a reconstructive and discriminative dictionary which helps to increase the discrimination power of the signal sparse representation based on this dictionary, while keeping a relative low reconstruction error, i.e. the reconstructed signal using the obtained sparse coefficients being as close to the original signal as possible. Therefore, inspired by [Huang & Aviyente 2006], the Fisher discrimination term [Bishop 2007] is introduced to the objective function.

Suppose S_W is the "intra-class scatter" which measures the within-class covariance:

$$S_W = \sum_{i=1}^{M} S_i \tag{3.17}$$

where

$$S_i = \sum_{x_j \in M_i} (x_j - m_i)(x_j - m_i)^T \tag{3.18}$$

$$m_i = \frac{1}{N_i} \sum_{x_j \in M_i} x_j \tag{3.19}$$

m_i is the mean of the signals belonging to category M_i. Let S_B denote the "inter-class scatter" which we identify as a measure of the between-class covariance

$$S_B = \sum_{i=1}^{M} N_i(m_i - m)(m_i - m)^T \tag{3.20}$$

where m is the mean of all signals

$$m = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{3.21}$$

Then, the Fisher discrimination score can be expressed as

$$F(X) = \frac{||S_B||_2^2}{||S_W||_2^2} = \frac{||\sum_{i=1}^{M} N_i(m_i - m)(m_i - m)^T||_2^2}{||\sum_{i=1}^{M}\sum_{x_j \in M_i}(x_j - m_i)(x_j - m_i)^T||_2^2} \tag{3.22}$$

The Fisher score is maximized when the distance between different categories is maximized while that within a category is minimized, thus making the classification task easier.

Integrating the Fisher discrimination term to (3.16) gives:

$$\min_{D,X,\Lambda} \{\lambda_1 \sum_{i=1}^{N} ||y_i - Dx_i||_2^2 + \lambda_2 \sum_{i=1}^{N} ||x_i||_0 - \lambda_3 F(X)\} \tag{3.23}$$

where $\Lambda = \{\lambda_1, \lambda_2, \lambda_3\}$ is, similarly to (3.16), the set of regularization parameters used to tune the tradeoff between the reconstruction error $\sum_{i=1}^{N} ||y_i - Dx_i||_2^2$, the sparsity $\sum_{i=1}^{N} ||x_i||_0$ and the discrimination power $F(X)$. The expected reconstructive and discriminative dictionary can be learned by solving properly the previous minimization problem. Thus, the signal sparse representation which gains the discrimination ability while retaining its faithfulness to the original signal can also be obtained through sparse coding based on the learned dictionary.

As mentioned previously, most of works in the literature use an iterative method to solve the dictionary learning problem. They generally contains two stages: sparse coding and dictionary update. We have followed this strategy for solving the minimization problem in (3.23). The first question that arises is "Given the dictionary, how to do the sparse coding faced with our reconstructive and discriminative objective function?". Since it involves not only a single signal but also all the training signals, the traditional sparse coding methods, such as BP and OMP, can not be directly applied to (3.23). Therefore we propose here a Sequential Forward Sparse Coding algorithm (SFSC) to do this task.

Let G being the function to be minimized:

$$G = \lambda_1 \sum_{i=1}^{N} ||y_i - Dx_i||_2^2 + \lambda_2 \sum_{i=1}^{N} ||x_i||_0 - \lambda_3 F(X) \tag{3.24}$$

The first step of SFSC consists in selecting one atom from the dictionary D with the smallest value of function G which is calculated by assuming that only that specific atom has been used for the sparse decomposition to obtain the sparse coefficients of all signals $\{x_i\}_{i=1}^{N}$ as well as X. Indeed, if we know beforehand the subset Γ of indices of atoms which are used for sparse decomposition, the sparse coefficients can easily be obtained using

$$X = D_\Gamma^+ Y \tag{3.25}$$

where D_Γ is a reduced dictionary composed only by the atoms whose indices are in Γ. Then in each following step, we continue to select one atom among the remaining ones, which yield the smallest value of G based on the subset of atoms formed by the combination of pre-selected atoms and this new one, until reaching the stopping rule. Here, the stopping rule can consist in achieving the predefined number of atoms used for sparse decomposition or stopping when the value of G begins to increase. The detailed algorithm is as follows:

SFSC algorithm

- Task: Given the dictionary $D \in \mathbb{R}^{n \times K}$, the regularization parameter set Λ and the set of signal $Y = [y_1, y_2, ..., y_N]$ to be represented by a linear combination of atoms from D, find the corresponding coefficients $X = [x_1, x_2, ..., x_N]$ that minimize G

$$G = \lambda_1 \sum_{i=1}^{N} ||y_i - Dx_i||_2^2 + \lambda_2 \sum_{i=1}^{N} ||x_i||_0 - \lambda_3 F(X).$$

- Initialization: Set the initial index set $\Gamma^0 = \emptyset$ and the indicator of iteration $t = 1$.

- Repeat until stopping rule:

  - For each $i \notin \Gamma^{t-1}$ and $i \in \{1, 2, ..., K\}$, let $\Psi = \Gamma^{t-1} \cup i$. Then calculate the sparse coefficients $X = D_{\psi}^{+} Y$ as well as the value of G_i based on X. D_{Ψ} represents the reduced dictionary composed by the columns in D whose indices are in Ψ
  - $i_{min} = \arg_i \min(G_i)$
  - $\Gamma^t = \Gamma^{t-1} \cup i_{min}$
  - $t = t + 1$

- Calculate the sparse coefficients $X = D_{\Gamma^t}^{+} Y$.

Concerning the dictionary update stage, we can employ the method of K-SVD introduced in section 3.2.1. Thus one complete dictionary learning algorithm is formed and ready to be used for generic visual object categorization. The entire classification algorithm is described as follows:

RD_SROC algorithm

1. Extract the feature vector representing the image visual content for all the images: $f_{i,j} \in \mathbb{R}^n, i \in \{1, 2, ..., M\}, j \in \{1, 2, ..., N_i\}$, where M is the number of categories and N_i is the number of images for i-th category.

2. Normalize all $f_{i,j}$ to have unit ℓ^2 norm.

3. Learn a reconstructive and discriminative dictionary D of sparse representation based on training images, by iteratively running the following two stages with the purpose of minimizing the objective function G. D is initialized by a subset of training image vectors, chosen randomly.

   - *Sparse Coding* using SFSC.
   - *Dictionary Update* similar to the dictionary update stage of K-SVD.

4. Compute the sparse coefficients of all the images based on the learned dictionary D, including the training images and test images.

5. Use a classifier (SVM for example) to accomplish the classification task, using the obtained sparse coefficients as input.

One advantage of our proposed RD_SROC is that other discrimination criteria can be easily employed by replacing $F(X)$ into the objective function, without changing the whole classification scheme. For example, we have tested the use a SVM accuracy as the discrimination term. All these experiments are presented in the next subsection.

3.4.2 Experimental results

Our experiments using RD_SROC have been performed on several commonly used datasets, including SIMPLIcity [Wang *et al.* 2001a], Caltech101 [L. Fei-Fei & Perona 2004] and Pascal 2007 [Everingham *et al.* 2007], in order to evaluate its discrimination ability. They will be presented respectively in the followings subsections.

3.4.2.1 Results on SIMPLIcity dataset

The results reported on the SIMPLIcity dataset [Wang *et al.* 2001b] are obtained with a 4-fold cross-validation. The same experimental configuration as the one used for evaluating R_SROC has been used (see details in section 3.3.2). Considering different discrimination criteria integrated in the objective function, three tests have been done to evaluate the performance of our proposed RD_SROC: using Fisher discrimination measure (noted as Fisher in the following); using the output of a SVM classifier with RBF kernel (noted as SVM_RBF in the following); using the output of a SVM classifier with linear kernel (noted as SVM_Linear in the following). All the regularization parameters are empirically set to have the same value, meaning that all the three terms, namely the reconstruction error, the sparsity and the discrimination power, in the objective function G have the same weight. The stopping rule of SFSC is set to use 60 atoms for sparse coding.

Before carrying out our experiments on SIMPLIcity, we would like to pay more attention to the dictionary size, which is considered to be a crucial parameter affecting the performance. Being different to the case of R_SROC where the dictionary is preconstructed of all training images and its size is determined directly, RD_SROC

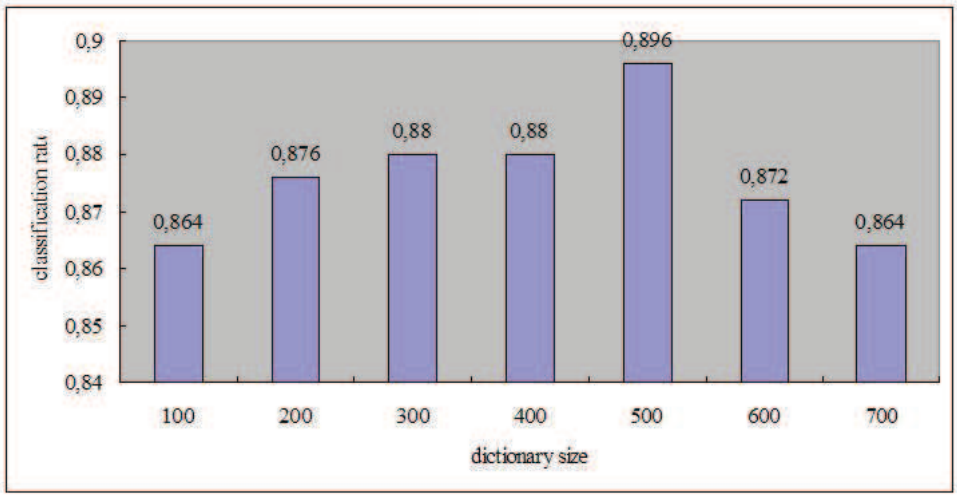

Figure 3.2: The classification rates using RD_SROC with different sizes of dictionary.

relies on the learning of a reconstructive and discriminative dictionary to better fit the classification task, whose optimal size can not be determined theoretically. Therefore, we have used 75% of all the images, namely 750 images, to train the dictionary with different sizes, from 100 to 700 with a step of 100. The classification test is done on the other 25% images and the size associated to the highest classification rate is retained for the whole experiments. From the Figure 3.2, we can clearly see that the classification rate rises with the increase of dictionary size from the beginning to 500, where it reaches the highest value 89.6%, and then it decreases if we continue to increase the dictionary size. Therefore, we have chosen the size of 500 for the following experiments.

Results using Fisher for visual object categorization on SIMPLIcity using RD_SROC are given in Table 3.3. Results using R_SROC have also been presented here for comparison purpose, using the same experimental configuration as RD_SROC. The different parts correspond to 4-fold cross-validation while (C1, C2, ..., C10) corresponds to the 10 categories. So "Average" in column is the classification rate averaging all the parts for a certain category and "Average" in line is inversely the classification rate averaging all the categories for a certain part. As a result, the value in the intersection of two "Average"s represents the final overall classification rate.

From the two tables, we note that the overall classification rate increases from 87.5% of R_SROC to 89.1% of RD_SROC with Fisher, which means that the classification ability of RD_SROC is really reinforced by adding the Fisher discrimination term to the standard sparse representation framework. Although the improvement is not so significant if we only take into account the absolute value of augmentation between them, we should say that it is still quite important and can not be neglected considering the relative small improvement space left. Because the higher the classification rate is, the more difficult it will be to let it be increased. Now let us look at the last column, i.e. the classification rate for single category. We can see that the superiority of RD_SROC is mainly due to the large improvement for difficult categories, namely the ones with lower rate such as C2 (Beach), C3 (Buildings) and C9 (Mountains & glaciers). For instance, 9% of augmentation has been observed for C9 using RD_SROC compared to R_SROC.

Table 3.5 and Table 3.6 present respectively the results of SVM_RBF and SVM_Linear. They have provided almost the same result with the overall classification rate of 87.6% for both of them, showing no advantage compared to R_SROC and being worse than RD_SROC with Fisher. This is probably due to the "overfitting" during the dictionary learning and classifier training, as we have used two independent SVM classifiers in the process, one for discrimination term and the other for final classification. However, it did not hurt much the performance either, proving that our proposed algorithm RD_SROC can robustly cooperate with different discrimination terms without changing the algorithm itself.

Detailed analysis Our proposed sparse coding method SFSC needs some criterion as stopping rule. It can be either the number of atoms used for sparse decomposition or the decrease of the objective function value. We have chosen the first criterion for its simplicity and the fact that it can avoid the case where many atoms have been used but only with very small coefficients thus yielding a non-sparse representation, which may probably happen with the second criterion. However, determining the optimal number of atoms used still remains an open question. In our experimentation, we have tested three typical values (30, 60, 100) for all three

Table 3.3: Classification Rate (CR) of Fisher for visual object categorization on SIMPLIcity using RD_SROC.

CR	1st part	2nd part	3rd part	4th part	Average
C1	92%	96%	80%	84%	88%
C2	72%	72%	80%	88%	78%
C3	88%	80%	84%	76%	82%
C4	100%	100%	96%	88%	96%
C5	100%	100%	100%	100%	100%
C6	84%	96%	88%	64%	83%
C7	100%	100%	100%	88%	97%
C8	88%	96%	96%	96%	94%
C9	88%	84%	88%	68%	82%
C10	84%	100%	96%	84%	91%
Average	89.6%	92.4%	90.8%	83.6%	89.1%

Table 3.4: Classification Rate (CR) for visual object categorization on SIMPLIcity using R_SROC (4-fold cross-validation).

CR	1st part	2nd part	3rd part	4th part	Average
C1	88%	96%	92%	84%	90%
C2	72%	76%	72%	72%	73%
C3	84%	76%	72%	80%	78%
C4	96%	100%	96%	96%	97%
C5	100%	100%	100%	100%	100%
C6	84%	100%	92%	68%	86%
C7	100%	100%	100%	80%	95%
C8	96%	96%	100%	100%	98%
C9	80%	76%	68%	68%	73%
C10	72%	100%	88%	80%	85%
Average	87.2%	92.0%	88.0%	82.8%	87.5%

Table 3.5: Classification Rate (CR) of SVM_RBF for visual object categorization on SIMPLIcity using RD_SROC.

CR	1st part	2nd part	3rd part	4th part	Average
C1	92%	88%	80%	84%	86%
C2	64%	68%	84%	88%	76%
C3	88%	76%	80%	72%	79%
C4	96%	96%	96%	92%	95%
C5	100%	100%	100%	100%	100%
C6	88%	88%	88%	76%	85%
C7	100%	100%	100%	88%	97%
C8	84%	96%	100%	96%	94%
C9	92%	76%	72%	68%	77%
C10	72%	100%	92%	84%	87%
Average	87.6%	88.8%	89.2%	84.8%	87.6%

Table 3.6: Classification Rate (CR) of SVM_Linear for visual object categorization on SIMPLIcity using RD_SROC.

CR	1st part	2nd part	3rd part	4th part	Average
C1	96%	84%	80%	80%	85%
C2	64%	64%	80%	88%	74%
C3	88%	72%	72%	76%	77%
C4	100%	100%	100%	88%	97%
C5	100%	100%	100%	100%	100%
C6	80%	92%	96%	60%	82%
C7	100%	100%	100%	88%	97%
C8	88%	96%	100%	96%	95%
C9	88%	80%	72%	72%	78%
C10	88%	100%	96%	80%	91%
Average	89.2%	88.8%	89.6%	82.8%	87.6%

Table 3.7: Classification Rate (CR) of Fisher for visual object categorization on SIMPLIcity using RD_SROC with 30 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	88%	88%	80%	80%	84%
C2	60%	64%	68%	88%	70%
C3	84%	72%	80%	56%	73%
C4	100%	100%	92%	76%	92%
C5	100%	100%	100%	100%	100%
C6	80%	96%	80%	56%	78%
C7	100%	100%	100%	88%	97%
C8	80%	96%	100%	96%	93%
C9	88%	72%	60%	60%	70%
C10	76%	100%	88%	88%	88%
Average	85.6%	88.8%	84.8%	78.8%	84.5%

experiments, namely Fisher, SVM_RBF and SVM_Linear, corresponding to (6%, 12%, 20%) of the total number of atoms. Besides the results presented above for 60 atoms used, the results of Fisher with 30 and 100 atoms used are presented in Table 3.7 and Table 3.8 respectively. Similarly the results of SVM_RBF with 30 and 100 atoms used are given in Table 3.9 and Table 3.10. Finally, the results of SVM_Linear with 30 and 100 atoms used are presented in Table 3.11 and Table 3.12.

From these tables of results, we can clearly see that the classification rates with 60 atoms and 100 atoms are much higher than that with 30 atoms, presenting an improvement of 4% in average. However, there is not much difference between the results with 60 atoms and 100 atoms, the results with 60 atoms being a little bit better than those with 100 atoms. This suggests that using 60 atoms is a good choice for space coding with SFSC and using more atoms may not be helpful to improve the performance.

As the authors of [Huang & Aviyente 2006] have also proposed a sparse coding method in their work, a supplemental experiment has been done by using their method instead of SFSC in RD_SROC. The same 3 numbers of atoms used have

Table 3.8: Classification Rate (CR) of Fisher for visual object categorization on SIMPLIcity using RD_SROC with 100 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	88%	92%	88%	92%	90%
C2	68%	76%	84%	96%	81%
C3	84%	80%	80%	72%	79%
C4	96%	100%	96%	88%	95%
C5	100%	100%	100%	100%	100%
C6	88%	96%	92%	40%	79%
C7	100%	100%	100%	84%	96%
C8	92%	96%	100%	96%	96%
C9	88%	84%	80%	68%	80%
C10	92%	100%	92%	92%	94%
Average	89.6%	92.4%	91.2%	82.8%	89.0%

Table 3.9: Classification Rate (CR) of SVM_RBF for visual object categorization on SIMPLIcity using RD_SROC with 30 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	88%	96%	76%	76%	84%
C2	56%	64%	64%	84%	67%
C3	84%	60%	76%	60%	70%
C4	96%	100%	92%	80%	92%
C5	100%	100%	100%	100%	100%
C6	80%	96%	88%	56%	80%
C7	100%	100%	100%	80%	95%
C8	76%	96%	92%	96%	90%
C9	72%	76%	72%	68%	72%
C10	84%	100%	92%	76%	88%
Average	83.6%	88.8%	85.2%	77.6%	83.8%

Table 3.10: Classification Rate (CR) of SVM_RBF for visual object categorization on SIMPLIcity using RD_SROC with 100 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	92%	96%	84%	80%	88%
C2	68%	76%	72%	88%	76%
C3	84%	68%	72%	80%	76%
C4	100%	100%	100%	84%	96%
C5	100%	100%	100%	100%	100%
C6	84%	96%	84%	56%	80%
C7	100%	100%	96%	76%	93%
C8	88%	96%	100%	96%	95%
C9	96%	76%	72%	60%	76%
C10	88%	96%	96%	88%	92%
Average	90.0%	90.4%	87.6%	80.8%	87.2%

Table 3.11: Classification Rate (CR) of SVM_Linear for visual object categorization on SIMPLIcity using RD_SROC with 30 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	92%	88%	76%	76%	83%
C2	56%	56%	76%	88%	69%
C3	84%	84%	64%	60%	73%
C4	96%	100%	92%	84%	93%
C5	100%	100%	100%	100%	100%
C6	80%	84%	88%	68%	80%
C7	100%	100%	100%	92%	98%
C8	84%	96%	92%	96%	92%
C9	76%	72%	64%	72%	71%
C10	76%	100%	92%	80%	87%
Average	84.4%	88.0%	84.4%	81.6%	84.6%

Table 3.12: Classification Rate (CR) of SVM_Linear for visual object categorization on SIMPLIcity using RD_SROC with 100 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	92%	92%	76%	88%	87%
C2	72%	68%	76%	88%	76%
C3	84%	76%	76%	76%	78%
C4	100%	100%	96%	80%	94%
C5	100%	100%	100%	100%	100%
C6	84%	100%	88%	56%	82%
C7	100%	100%	100%	92%	98%
C8	92%	92%	100%	96%	95%
C9	88%	76%	76%	64%	76%
C10	84%	100%	96%	72%	88%
Average	89.6%	90.4%	88.4%	81.2%	87.4%

been considered, namely (30, 60, 100), and their results are listed respectively in Table 3.13, 3.14 and 3.15. A severe degradation of performance has been observed compared to our approach, the decrease being of 5%-8% for the classification rate. Among the three numbers of atoms used, we can see that 60 is still a good choice which balances the performance and the computational burden.

3.4.2.2 Results on Caltech101 dataset

Caltech101 [L. Fei-Fei & Perona 2004] is a dataset which contains 101 categories of objects and one extra background category, thus having a total of 102 categories. Most categories contain about 50 images while some of them may contain only 30 images or up to 800 images. Some sample images are presented in Figure 3.3.

A traditional experimental configuration is used, i.e. 15 images chosen randomly from each category for training and another 15 images chosen in the same way for test. So we have totally 1530 training images and the same number of test images. Concerning the feature set, SIFT, CSIFT, OSIFT, LBP and HOG have been employed (see 2.2.1 for more details). Here one problem is that the number of SIFT-like features extracted from an image can vary from one image to another, while our

Table 3.13: Classification Rate (CR) for visual object categorization on SIMPLIcity using RD_SROC with the sparse coding method of [Huang & Aviyente 2006] and 30 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	84%	76%	64%	72%	74%
C2	68%	60%	60%	68%	64%
C3	80%	68%	76%	48%	68%
C4	92%	84%	88%	84%	87%
C5	100%	100%	100%	100%	100%
C6	80%	88%	64%	64%	74%
C7	100%	100%	100%	72%	93%
C8	68%	100%	84%	100%	88%
C9	80%	64%	64%	60%	67%
C10	60%	100%	80%	76%	79%
Average	81.2%	84.0%	78.0%	74.4%	79.4%

Table 3.14: Classification Rate (CR) for visual object categorization on SIMPLIcity using RD_SROC with the sparse coding method of [Huang & Aviyente 2006] and 60 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	96%	80%	64%	76%	79%
C2	60%	56%	60%	68%	61%
C3	76%	80%	76%	60%	73%
C4	96%	100%	88%	84%	92%
C5	100%	100%	100%	100%	100%
C6	84%	88%	76%	52%	75%
C7	100%	96%	100%	76%	93%
C8	72%	88%	96%	100%	89%
C9	76%	64%	60%	68%	67%
C10	68%	96%	80%	76%	80%
Average	82.8%	84.8%	80.0%	76.0%	80.9%

Table 3.15: Classification Rate (CR) for visual object categorization on SIMPLIcity using RD_SROC with the sparse coding method of [Huang & Aviyente 2006] and 100 atoms.

CR	1st part	2nd part	3rd part	4th part	Average
C1	92%	84%	64%	80%	80%
C2	60%	60%	56%	76%	63%
C3	92%	80%	68%	64%	76%
C4	96%	100%	76%	84%	89%
C5	100%	100%	100%	100%	100%
C6	84%	92%	80%	60%	79%
C7	100%	96%	100%	76%	93%
C8	60%	96%	92%	100%	87%
C9	84%	60%	68%	56%	67%
C10	64%	96%	80%	68%	77%
Average	83.2%	86.4%	78.4%	76.4%	81.1%

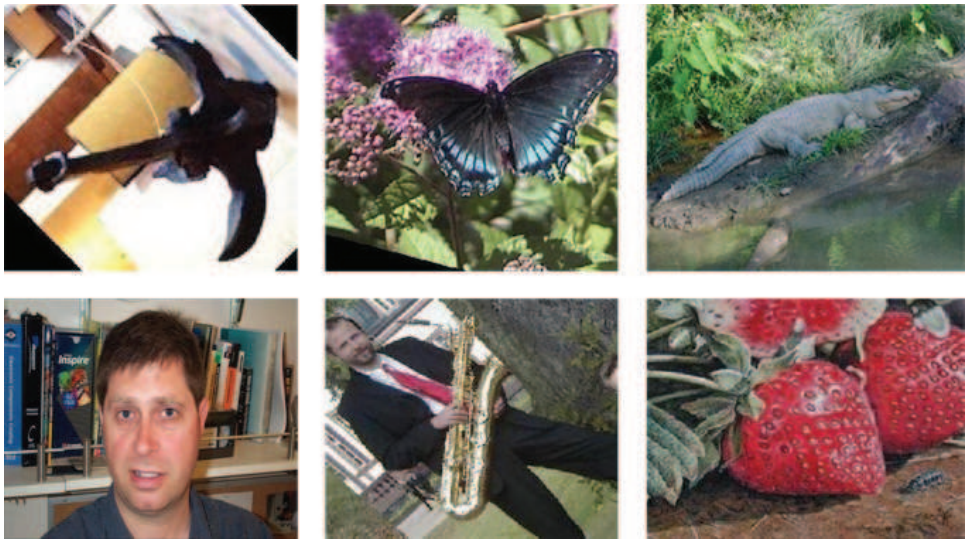

Figure 3.3: Some sample images from Caltech101 dataset (from top to bottom, from left to right, they belong to anchor, butterfly, crocodile, face, saxophone and strawberry).

RD_SROC algorithm requires one feature vector with the same dimension to represent an image. Thus, the classical "Bag of Features" (BoF) approach is chosen for this purpose as its effectiveness has been demonstrated in [Everingham *et al.* 2007]. Finally, a spatial pyramid as explained by Figure 2.8 is also applied, thus yielding 8 histograms corresponding to the whole image and 7 subregions for one image, noted as sp1 to sp8.

The classification process is as follows. First a reconstructive and discriminative dictionary is trained on BoF feature vectors sp1 using RD_SROC with 500 atoms in the dictionary and 60 atoms used for sparse coding, which corresponds to the best configuration evaluated previously during our experiments on SIMPLIcity. Then a sparse coding using SFSC is performed for all BoF features vectors sp1 to sp8 to obtain the corresponding Sparse Representation Coefficients (SRC) sp1 to sp8. After that, kernel matrices of SRC are computed for each spatial pyramid level and merged together using the proportion as merging weights, which is the ratio of areas of its corresponding subregion compared to the whole image, to form the final entire kernel matrix (see section 2.4.1.3 for details). In this step, 3 normalization methods of kernel matrices are considered before their fusion: 1, no normalization; 2, mean normalization in which each kernel matrix is divided by its mean value; 3 std normalization in which each kernel matrix is normalized to have mean 0 and standard deviation 1. Finally, one-against-one SVM classifier is used to perform the classification task for every type of feature, with 2 popular kernels evaluated, namely linear kernel and RBF kernel (the parameter γ in RBF kernel is optimized using 4-fold cross-validation on training sets of SRC sp1). The ultimate decision of category for a test image is made based on its maximal probability after the average fusion of probabilities given by SVM for each type of feature.

The detailed results in terms of classification rate are presented in Table 3.16, where we also show the results obtained when using directly BoF feature vectors to feed SVM classifier after kernel computation and fusion for comparison purpose. We can note that the type of normalization method has almost no impact on the performance and all of them provide almost the same classification rate. This might mean that the normalization step before the fusion of kernel matrices is not necessary.

Table 3.16: Classification Rate (CR) for visual object categorization on Caltech101. SRC means the results obtained using coefficients from RD_SROC and BoF means the results obtained using BoF directly.

CR	non normalization		mean normalization		std normalization	
Type of results	SRC	BoF	SRC	BoF	SRC	BoF
Linear kernel	46.3%	21.8%	46.7%	29.4%	46.5%	30.7%
RBF kernel	45.6%	24.8%	45.6%	32.6%	45.8%	28.2%

Moreover, our RD_SROC is robust to both of linear kernel and RBF kernel with a little bit favor for linear kernel. This is a very interesting property because linear kernel is much more computational efficient, especially in the case where we should find a good γ for RBF kernel to insure a better performance. However, the obtained results are worse than other results reported in the literature, which can reach classification rates around 60% to 75%. This has led us to do a comparative experiment, whose results are in the same table, in order to know whether this degradation came from RD_SROC itself or other components of the whole classification process.

We can see that using BoF directly has provided even worse results compared to SRC, which proves that the image representation SRC obtained through RD_SROC has gained indeed more discrimination ability, making it more suitable for classification task. But why the overall result is not so good? We guess that a deeper research in the future on many steps in the whole classification process would be interesting and useful to improve the performance as the goal is to find a best adapted classification method to sparse representation based image representation. These steps include the task dependent parameter regularization for RD_SROC, the fusion of different spatial pyramid levels, the intelligent way to combine the results of different features (for example, we can replace SVM by MKL to realize an automatic feature combination in the kernel level), the design of novel kernels to best fit the properties of our sparse representation base image representation etc.

3.4.2.3 Results on Pascal 2007 dataset

We have also evaluated our RD_SROC on 5 representative categories of Pascal 2007 dataset [Everingham *et al.* 2007], namely aeroplane, bicycle, bus, horse and person. As the images in this dataset may have several labels, i.e. one image may belong to several categories, it is necessary to build one classifier per category using one against all strategy. This kind of classification configuration offers us the possibility to propose an innovation compared to the previous experiments, considering not only the reconstructive and discriminative dictionary but also the adapted purely reconstructive dictionary for one category. Inspired by [Perronnin *et al.* 2006], the basic idea is very simple: we first train a reconstructive and discriminative dictionary based on both the positive training images and negative training images using RD_SROC, and then a purely reconstructive dictionary can be adapted from the previously obtained dictionary, using the images from that category only through the combination of OMP and K-SVD. The final dictionary is the combination of them. Thus we can expect that an image is better approximated by the atoms in the adapted dictionary of a certain category if it belongs to this category and otherwise it would rather be described by the atoms in the reconstructive and discriminative dictionary.

Most of the experimental configuration is the same as that of Caltech101, except that we have a final dictionary with the size of 1000, which is the result of combination of two dictionaries with the size of 500. Therefore, the number of atoms used for sparse coding is correspondingly changed to 120. As we have already shown that the type of kernels and normalization methods has almost no impact on the performance of the experiments on Caltech101, we have just used linear kernel without normalization before kernel fusion on Pascal 2007, with the purpose of reducing the computational cost. The results are given in Table 3.17. Morever, Table 3.18 presents the results using BoF directly, without sparse representation.

Average fusion in the tables means that the results generated by different features are equally fused to form the final result while val fusion takes the normalized average precision on validation data as weighs to fuse the results of different features.

Table 3.17: Average precision (AP) for visual object categorization on Pascal 2007 using SRC.

AP for SRC	average fusion	val fusion
aeroplane	0.615	0.605
bicycle	0.281	0.287
bus	0.301	0.301
horse	0.640	0.639
person	0.709	0.710

Table 3.18: Average precision (AP) for visual object categorization on Pascal 2007 using BoF directly.

AP for BoF	average fusion	val fusion
aeroplane	0.517	0.522
bicycle	0.114	0.114
bus	0.148	0.148
horse	0.447	0.465
person	0.610	0.610

Actually, they exhibited more or less the same performance. By comparing these two tables, we can see that our approach performed better than using BoF directly, with large superiority for all the 5 categories. But the overall performance even if it remains in an acceptable level, is still distant to the best reported in the challenge (see Table 2.10 for details). Moreover, if we compare the results in Table 3.17 to the best results obtained using SMIR in Table 2.9, we find that the performance increases significantly for some categories, such as "aeroplane" and "horse", whereas for other categories, the performance regrettably decreases. All phenomena reveal again that further research in the future on the points mentioned in the case of Caltech101 would be useful to exploit the potential discrimination ability of our sparse representation based image representation.

3.5 Conclusion

Sparse representation is originally used in signal processing as a powerful tool for acquiring, representing and compressing high-dimensional signals. Motivated by the great successes it has achieved, recently it has become a hot research topic in the domain of computer vision and pattern recognition. In this chapter, we have proposed two approaches for visual object categorization via sparse representation, including a reconstructive method (R_SROC) as well as a reconstructive and discriminative one (RD_SROC). Based on the intuitive hypothesis that an image can be represented by a linear combination of training images from the same category, R_SROC approach first computes the sparse representation of images through solving the ℓ^1 (or ℓ^0) norm minimization problem and then uses them as new feature vectors for images to be classified by traditional classifiers such as SVM in our case. To improve the discrimination ability of the sparse representation, we have proposed RD_SROC which includes a discrimination term, such as Fisher discrimination measure or the output of a SVM classifier, to the standard sparse representation objective function in order to learn a reconstructive and discriminative dictionary.

Experiments carried out on the SIMPLIcity dataset have clearly revealed that our reconstructive approach has gained an obvious improvement of the classification accuracy compared to standard SVM using image features as input. Moreover, our reconstructive and discriminative approach has obtained better results than pure reconstructive one which shows that adding a discrimination term for constructing the sparse representation is more suitable for the classification task.

Experiments on Caltech101 and Pascal 2007 datasets, have revealed that our approach has indeed gained more discrimination ability compared to the traditional "bag of features" representation. However, even if the overall performance remains in an acceptable level, it is still lower than some of state of the art methods.

Thus, we believe that sparse representation can greatly help for designing efficient approaches for VOC purpose. We have proposed in this chapter two innovative and promising methods but since it is a rather precursory work, many directions still need to be investigated, including the way to identify optimal regularization pa-

rameters for RD_SROC, the way different spatial pyramid levels should be fused, the way to combine the results from different features (for example, we can replace SVM by MKL to realize an automatic feature combination in the kernel level), the design of novel kernels to best fit the properties of our sparse image representation.

Chapter 4

Conclusion and future works

Contents

This thesis addresses the active research topic of generic visual object categorization (VOC) which consists in labeling a real world image according to the objects it contains given a set of categories under consideration.

Without imposing any restriction on the processed images, we are faced with image content that may be heterogeneous, ambiguous, and also acquired under poor conditions. Moreover, we have to deal with problems inherent to object categories like the wide variations of shape and appearance of objects inside a category, and due to the representation of an object in an image, such as various scales and orientations, as well as illumination and occlusion problems. To all these difficulties, we also need to add the one induced by the large number of real world object types that need to be discriminated.

Despite many efforts and much progress that have been made during the past years, VOC remains an open and very challenging problem. In this context, we have proposed in this thesis our contributions, especially concerning the two main components of the methods addressing this problem, namely features selection and image representation. In the following, we will first summarize our contributions and then propose some perspectives which would be interesting for future work.

4.1 Contributions

- Firstly, we have proposed an Embedded Sequential Forward feature Selection algorithm (ESFS) for VOC. Its goal is to select the most important and non-redundant features for obtaining a good performance for the categorization. It is mainly based on the commonly used sub-optimal search method Sequential Forward Selection (SFS), which relies on the simple principle to add incrementally most relevant features. However, ESFS not only adds incrementally most relevant features in each step but also merges them in an embedded way thanks to the concept of combined mass functions from the evidence theory which also offers the benefit of obtaining a computational cost much lower than the one of original SFS. Experiments have shown that used as a filter selection method, ESFS performs better than widely used state of the art approaches such as Fisher and PCA for the filter methods and SFS, SFFS and OS for the wrapper approaches applied to the visual object categorization task. Moreover, ESFS can be used not only as a feature selection method, but also directly as a classifier.

- Secondly, we have proposed novel image representations through polynomial interpolation and statistical measures, called PMIR and SMIR respectively, to model the visual content of an image. They allow to overcome the main drawback of the popular "bag of features" method which is the difficulty to fix the optimal size of the visual vocabulary. Moreover, when a GMM is used for a soft assignment, we can avoid the inaccurate assumption of the Gaussian distribution of features which is not always the case in the different applications. Finally, our representations are also able to cope with a smaller number of feature vectors per image, a situation that we often encounter. We have tested PMIR and SMIR on a subset of Pascal 2007 dataset along with our proposed region based features and SIFT. Two different fusion strategies, early and late, have also been considered to merge information from different "channels" represented by the different types of features. Results of PMIR have shown that a good performance can be achieved with our approach and

that our segment features carry information which is complementary to the one of SIFT features. A deeper evaluation of SMIR combined with several dimensionality reduction methods and classifier construction techniques facing unbalanced dataset has then been carried out and drawn the same conclusion as in the case of PMIR, that is a good classification accuracy, which is comparable to the best results reported in the Pascal challenge, can be achieved and our region based features and SIFT are complementary to each other.

- Thirdly, we have proposed two approaches for VOC via sparse representation, including a reconstructive method (R_SROC) as well as a reconstructive and discriminative one (RD_SROC). Indeed, sparse representation model of signals has received a lot of attentions and has been a very active research area in recent years. Recently, techniques from sparse signal representation have significantly impacted the domain of computer vision and pattern recognition. This has motivated us to propose approaches adapting these principles to the problem of VOC.

  Based on the intuitive hypothesis that an image can be represented by a linear combination of training images from the same category, R_SROC approach first computes the sparse representation of images through solving the ℓ^1 (or ℓ^0) norm minimization problem and then uses them as new feature vectors for images to be classified by traditional classifiers such as SVM in our case. To improve the discrimination ability of the sparse representation to better fit the classification problem, we have also proposed RD_SROC which includes a discrimination term, such as Fisher discrimination measure or the output of a SVM classifier, to the standard sparse representation objective function in order to learn a reconstructive and discriminative dictionary. Moreover, we have also proposed to combine the reconstructive and discriminative dictionary and the adapted pure reconstructive dictionary for a given category so that the discrimination power can further be increased.

  Experiments carried out on the SIMPLIcity dataset have clearly revealed that our reconstructive approach has gained an obvious improvement of the classi-

fication accuracy compared to standard SVM using image features as input. Moreover, our reconstructive and discriminative approach has obtained better results than pure reconstructive one which shows that adding a discrimination term for constructing the sparse representation is more suitable for the classification task.

Supplemental experiments on Caltech101 and Pascal 2007 datasets, have revealed that our approach has gained more discrimination ability compared to the traditional "bag of features" representations. However, even if the overall performance remains in an acceptable level, it is still lower than some of state of the art methods, which suggests that the promising sparse image representation may be improved to better fit VOC properties.

4.2 Perspectives for future works

Extensions of this work that we envisage, not only concerning feature selection but also image representation, are presented in the following paragraphs.

- We plan to investigate alternative solutions for building mass functions associated to each single feature within ESFS. Indeed, for the moment, masses are distributed on single classes for a given feature. However, the evidence theory allows the reasoning on union of classes, which may be more accurate. Moreover, an interesting issue would be to integrate into the feature selection process the conflict information that can be obtained from combined mass functions and which may allow to avoid combining features that give contradictory informations. Indeed, even if several fusion operators we considered integrate the notion of conflict, such as the one of Dempster and Yager, their performance has not been significantly improved compared to the performance of TBM which does not handle the conflict. Therefore further research is needed in order to integrate the conflict information in a more efficient way.

- Since features of different natures extracted from an image often carry different image informations which can contribute respectively to the final image clas-

sification from different aspects, the fusion of them is considered to be able to effectively improve the classification performance. This point of view has been confirmed in both of experiments using PMIR and SMIR. However, actually we have only evaluated in this work two fusion strategies, namely early fusion and late fusion, and we think that it might be meaningful to consider other numerous intermediate strategies which may consist in generating intermediate classes from different sources and to take a final decision based on these intermediate classes. The objective would be to find a fusion strategy which allows to best exploit the complementarity between features while eliminating as much as possible their contradictory part.

- As an extension of SVM, MKL allows to use a combination of kernels instead of a single one in SVM. Each basis kernel in the combination can either be different kernels with different parameter configurations or use different types of features. This characteristic offers more freedom to incorporate more features combined with different kernels to improve the performance, since MKL performs an automatic feature fusion and feature selection during the training procedure. Therefore, we think that it would be interesting to evaluate MKL for the classification using our approaches, instead of the current SVM.

- We believe that sparse representation can greatly help for designing efficient approaches for VOC purpose. We have proposed two innovative and promising methods, R_SROC and RD_SROC, but since it is a rather precursory work, many directions still need to be investigated. In particular, parameter regularization is an important aspect for these methods, especially the weights attributed to the reconstructive term, discriminative term and sparsity in the objective function. Actually, we have empirically used equal weighting for all these 3 terms. However, it might not be the optimal choice for achieving the best performance. Exploiting intelligent ways for its automatic determination depending on a concrete object categorization task would be another interesting direction for future improvement. Moreover, the way to combine the results from different features is another important point. We plan to

replace SVM by MKL to perform an automatic feature combination at the kernel level using novel kernels to better fit the properties of our sparse image representation. Finally, although SMIR has obtained comparable results to that reported in the Pascal challenge, it is still less effective than the best method in the challenge which mainly relies on BoF. This is the reason why we have chosen, in case of local image features, to make use of BoF to compute the image representation further used for sparse representation. However, we envisage to evaluate the efficiency of our SMIR representation instead of BoF within our R_SROC and RD_SROC sparse image representations.

Publications

During this thesis, 5 papers have been published in international conferences, 1 paper has been submitted to IEEE Transaction on Knowledge and Data Engineering. There is another paper published in international journal in the domain of virtual reality, with collaboration of my ancient colleague.

International Conferences:

1. H. Fu, A. Pujol, E. Dellandréa, L. Chen: Image modeling using statistical measures for visual object categorization, International Conference on Image Processing Theory, Tools and Applications (IPTA'10), pp. 319-324, April 2010.

2. C. Zhu, H. Fu, C.E. Bichot, E. Dellandréa, L. Chen: Visual object recognition using local binary patterns and segment-based feature, International Conference on Image Processing Theory, Tools and Applications (IPTA'10), pp. 426-431, April 2010.

3. H. Fu, C. Zhu, E. Dellandréa, C.E. Bichot, L. Chen: Visual object categorization via sparse representation, International Conference on Image and Graphics (ICIG'09), pp. 943-948, June 2009.

4. H. Fu, Z. Xiao, E. Dellandréa, W. Dou, L. Chen: Image categorization using ESFS: a new embedded feature selection method based on SFS, Advanced Concepts for Intelligent Vision Systems (Acivs 2009), pp. 288-299, April 2009.

5. H. Fu, A. Pujol, E. Dellandréa, L. Chen: Region based visual object categorization using segment features and polynomial modeling, IAPR International Workshops on Structural, Syntactic and Statistical Pattern Recognition (S+SSPR 2008), in conjunction with ICPR 2008, pp. 277-286, April 2008.

Submission to an International Journal:

1. H. Fu, Z. Xiao, E. Dellandréa, W. Dou, L. Chen: A new embedded sequential feature selection method for categorization of image and audio, submitted to IEEE Transaction on Knowledge and Data Engineering, 2010.

International Journal:

1. L. Ma, W. Zhang, H. Fu, Y. Guo, D. Chablat, F. Bennis: A framework for interactive work design based on motion tracking, simulation, and analysis, International Journal of Human Factors and Ergonomics in Manufacturing, Volume 20, Issue 4, pp. 339-352, June 2009.

Bibliography

[Agarwal & Roth 2002] S. Agarwal and D. Roth. *Learning a sparse representation for object detection.* In Proceedings of the European Conference on Computer Vision, volume 4, pages 113–130, 2002. 19

[Aharon *et al.* 2006] M. Aharon, M. Elad and A. Bruckstein. *K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation.* IEEE Transactions on Signal Processing, vol. 54, no. 11, pages 4311–4322, 2006. 5, 86, 87

[Almuallim & Dietterich 1991] H. Almuallim and T.G. Dietterich. *Learning with many irrelevant features.* In Proceedings of the National Conference on Artificial Intelligence, pages 547–552, 1991. 31

[Arauzo-Azofra *et al.* 2004] A. Arauzo-Azofra, J.M. Benitez and J.L. Castro. *A feature set measure based on relief.* In Proceedings of the International Conference on Recent Advances in Soft Computing, pages 104–109, 2004. 31

[Ardabilian & Chen 2001] M. Ardabilian and L. Chen. *A new line extraction algorithm: Fast connective hough transform.* In Proceedings of the International Conference on Pattern Recognition and Information Processing, pages 127–134, 2001. 61

[Ayache *et al.* 2007a] S. Ayache, G. Quénot and J. Gensel. *Classifier fusion for SVM-based multimedia semantic indexing.* In Proceedings of the European Conference on Information Retrieval, pages 494–504, 2007. 28

[Ayache *et al.* 2007b] S. Ayache, G. Quénot and J. Gensel. *Image and video indexing using networks of operators.* EURASIP Journal on Image and Video Processing, vol. 2007, no. 4, pages 1–13, 2007. 29

[Bach *et al.* 2004] F.R. Bach, G.R.G. Lanckriet and M.I. Jordan. *Multiple kernel learning, conic duality, and the SMO algorithm.* In Proceedings of the International Conference on Machine Learning, 2004. 26

[Barnard *et al.* 2003] K. Barnard, P. Duygulu, R. Guru, P. Gabbur and D.A. Forsyth. *The effects of segmentation and feature choice in a translation model of object recognition.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pages 675–682, 2003. 57, 58

[Bellman 1961] R. Bellman. Adaptive control processes: A guided tour. Princeton University Press, 1961. 4, 30, 63, 69

[Bishop 2007] C.M. Bishop. Pattern recognition and machine learning. Springer, 2007. 21, 22, 97

[Blumensath & Davies 2007] T. Blumensath and M.E. Davies. *On the difference between orthogonal matching pursuit and orthogonal least squares.* Technical report, IDCOM & Joint Research Institute for Signal and Image Processing, Edinburgh University, 2007. 85

[Boiman *et al.* 2008] O. Boiman, E. Shechtman and M. Irani. *In defense of nearst-neighbor based image classification.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, 2008. 52

[Bouchard & Triggs 2004] G. Bouchard and B. Triggs. *The trade-off between generative and discriminative classifiers.* In Proceedings of the IASC Symposium in Computational Statistics, pages 721–728, 2004. 23

[Breiman *et al.* 1984] L. Breiman, J.H. Friedman, R. Olshen and C.J. Stone. Classification and regression trees. Chapman & Hall/CRC, 1984. 32

[Bruckstein *et al.* 2009] A.M. Bruckstein, D.L. Donoho and M. Elad. *From sparse solutions of systems of equations to sparse modeling of signals and images.* Society for Industrial and Applied Mathematics Review, vol. 51, no. 1, pages 34–81, 2009. 82

[Candès 2006] E.J. Candès. *Compressive sampling.* In Proceedings of the International Congress of Mathematicians, 2006. 88

[Canny 1986] J. Canny. *A computational approach to edge detection.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pages 679–698, 1986. 61

[Chang & Lin 2001] C.C. Chang and C.J. Lin. *A library for support vector machines.* Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm, 2001. 25, 73, 95

[Chen *et al.* 1998] S.S. Chen, D.L. Donoho and M.A. Saunders. *Atomic decomposition by basis pursuit.* SIAM Journal on Scientic Computing, vol. 20, no. 1, pages 33–61, 1998. 84

[Combarro *et al.* 2005] E.F. Combarro, E. Montanes, I. Diaz, J. Ranilla and R. Mones. *Introducing a family of linear measures for feature selection in text categorization.* IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 9, pages 1223–1232, 2005. 30

[Cortes & Vapnik 1995] C. Cortes and V. Vapnik. *Support vector networks.* Machine learning, vol. 20, no. 3, pages 273–297, 1995. 24

[Cortes & Vapnik 2005] C. Cortes and V. Vapnik. *Support vector networks.* Machine Learning, vol. 20, pages 273–297, 2005. 24

[Cristianini & Shawe-Taylor 2000] N. Cristianini and J. Shawe-Taylor. An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press, 2000. 24

[Dalal & Triggs 2005] N. Dalal and B. Triggs. *Histograms of oriented gradients for human detection*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 1, pages 886–893, 2005. 15

[Dance *et al.* 2004] C. Dance, J. Willamowski, L. Fan, C. Bray and G. Csurka. *Visual categorization with bags of keypoints*. In ECCV International Workshop on Statistical Learning in Computer Vision, 2004. 20, 49

[Daubechies *et al.* 2004] I. Daubechies, M. Defrise and C. De-Mol. *An iterative thresholding algorithm for linear inverse problems with a sparsity constraint*. Communications on Pure and Applied Mathematics, pages 1413–1457, 2004. 85

[Daugman 1985] J.G. Daugman. *Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by twodimensional visual cortical filters*. Journal of The Optical Society of America, vol. 2, no. 7, pages 1160–1169, 1985. 15

[Dellaert 2002] F. Dellaert. *The expectation maximization algorithm*. Technical report, College of Computing, Georgia Institute of Technology, 2002. 21

[Dempster 1968] A.P. Dempster. *A generalization of bayesian inference*. Journal of the Royal Statistical Society, Series B, vol. 30, 1968. 35

[Deng *et al.* 2001] Y. Deng, B.S. Manjunath, C. Kenney, M.S. Moore and H. Shin. *An efficient color representation for image retrieval*. IEEE Transactions on Image Processing, vol. 10, no. 1, pages 140–147, 2001. 61

[Desolneux *et al.* 2008] A. Desolneux, L. Moisan and J.M. Morel. From gestalt theory to image analysis: A probabilistic approach. Springer, 2008. 57

[Dikmen & Huang 2008] M. Dikmen and T. Huang. *Robust estimation of foreground in surveillance video by sparse error estimation*. In Proceedings of the International Conference on Image Processing, pages 1–4, 2008. 83

[Doak 1992] J. Doak. *An evaluation of feature selection methods and their application to computer security*. CSE Technical report 92-18, University of California at Davis, 1992. 32

[Donoho & Huo 2001] D. Donoho and X. Huo. *Uncertainty principles and ideal atomic decomposition*. IEEE Transaction on Information Theory, vol. 47, no. 7, pages 2845–2862, 2001. 84

[Donoho 2004] D.L. Donoho. *For most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution*. Communications on Pure and Applied Mathematics, vol. 59, pages 797–829, 2004. 84

[Duarte-Carvajalino & Sapiro 2009] J.M. Duarte-Carvajalino and G. Sapiro. *Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization.* IEEE Transactions on Image Processing, vol. 18, no. 7, pages 1395–1408, 2009. 88

[Elad & Aharon 2006] M. Elad and M. Aharon. *Image denoising via learned dictionaries and sparse representation.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 1, pages 895–900, 2006. 82

[Engan *et al.* 1999] K. Engan, S.O. Aase and J.H. Hakon-Husoy. *Method of optimal directions for frame design.* In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, volume 5, pages 2443–2446, 1999. 86

[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/, 2007. v, 16, 17, 20, 49, 64, 74, 76, 101, 112, 114

[Everingham *et al.* 2008] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn and A. Zisserman. *The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results.* http://www.pascal-network.org/challenges/VOC/voc2008/workshop/, 2008. vii, 20, 49, 54, 55

[Farquhar *et al.* 2005] J.D.R. Farquhar, S. Szedmak, H. Meng and J. Shawe-Taylor. *Improving "bag-of-keypoints" image categorisation: Generative models and PDF-kernels.* Technical report, University of Southampton, 2005. 50, 51, 52

[Fei-Fei & Perona 2005] L. Fei-Fei and P. Perona. *A Bayesian hierarchical model for learning natural scene categories.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pages 524–531, 2005. 10, 20

[Felzenszwalb & Huttenlocher 2005] P.F. Felzenszwalb and D.P. Huttenlocher. *Pictorial structures for object recognition.* International Journal of Computer Vision, vol. 61, no. 1, pages 55–79, 2005. 19

[Figueiredo *et al.* 2007] M. Figueiredo, R.D. Nowak and S.J. Wright. *Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems.* IEEE Journal of Selected Topics in Signal Processing, vol. 1, no. 4, pages 586–597, 2007. 85

[Fisher 1936] R.A. Fisher. *The use of multiple measurements in taxonomic problems.* Annals of Eugenics, vol. 7, pages 179–188, 1936. 63, 71

[Freund & Schapire 1997] Y. Freund and R.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. 28, 71

[Freund & Schapire 1999] Y. Freund and R. Schapire. *A short introduction to boosting.* Journal of Japanese Society for Artificial Intelligence, vol. 14, no. 5, pages 771–780, 1999. 29, 71

[Fu *et al.* 2008] H. Fu, A. Pujol, E. Dellandréa and L. Chen. *Region based visual object categorization using segment features and polynomial modeling.* In IAPR International Workshops on Structural, Syntactic and Statistical Pattern Recognition, in conjunction with ICPR 2008, pages 277–286, 2008. 58, 63

[Fu *et al.* 2009a] H. Fu, Z. Xiao, E. Dellandréa, W. Dou and L. Chen. *Image categorization using ESFS: A new embedded feature selection method based on SFS.* In Proceedings of the Advanced Concepts for Intelligent Vision Systems, pages 288–299, 2009. 34

[Fu *et al.* 2009b] H. Fu, C. Zhu, E. Dellandréa, C.E. Bichot and L. Chen. *Visual object categorization via sparse representation.* In Proceedings of the International Conference on Image and Graphics, pages 943–948, 2009. 86, 91

[Fu *et al.* 2010] H. Fu, A. Pujol, E. Dellandréa and L. Chen. *Image modeling using statistical measures for visual object categorization.* In Proceedings of the International Conference on Image Processing Theory, Tools and Applications, pages 319–324, 2010. 68

[Gersho & Gray 1991] A. Gersho and R.M. Gray. Vector quantization and signal compression. Kluwer Academic, 1991. 26

[Gorodnitsky & Rao 1997] I.F. Gorodnitsky and B.D. Rao. *Sparse signal reconstruction from limited data using FOCUSS: A re-weighted norm minimization algorithm.* IEEE Transactions on Signal Processing, vol. 45, no. 3, pages 600–616, 1997. 85

[Grauman & Darrell 2005] K. Grauman and T. Darrell. *Pyramid match kernels: Discriminative classification with sets of image features.* In Proceedings of the International Conference on Computer Vision, pages 1458–1465, 2005. 27, 54

[Guo *et al.* 2010] Y. Guo, S. Ruan, J. Landré and J.M. Constans. *A sparse representation method for magnetic resonance spectroscopy quantification.* IEEE Transactions on Biomedical Engineering, vol. 57, no. 7, pages 1620–1627, 2010. 82

[Guyon & Elisseff 2003] I. Guyon and A. Elisseff. *An introduction to variable and feature selection.* Journal of Machine Learning Research, vol. 3, pages 1157–1182, 2003. 30

[Guyon *et al.* 2002] I. Guyon, J. Weston, S. Barnhill and V. Vapnik. *Gene selection for cancer classification using support vector machines.* Machine Learning, vol. 46, no. 1-3, pages 389–422, 2002. 32

[Hall & Smith 1997] M.A. Hall and L.A. Smith. *Feature subset selection: A correlation based filter approach.* In Proceedings of the International Conference on Neural Information Processing and Intelligent Information Systems, pages 855–858, 1997. 30, 31

[Harris & Stephens 1988] C. Harris and M. Stephens. *A combined corner and edge detector.* In Proceedings of the Alvey Vision Conference, pages 147–152, 1988. 10

[Hofmann 1998] T. Hofmann. *Probabilistic latent semantic indexing.* In Proceedings of the International SIGIR Conference on Research and Development in Information Retrieval, pages 50–57, 1998. 26

[Huang & Aviyente 2006] K. Huang and S. Aviyente. *Sparse representation for signal classification.* In Proceedings of the Advances in Neural Information Processing Systems, volume 19, pages 609–616, 2006. vi, 90, 97, 106, 110, 111

[Huang *et al.* 1997] J. Huang, S.R. Kumar, M. Mitra, W. Zhu and R. Zabih. *Image indexing using color correlograms.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 762–768, 1997. 11, 45

[Huang *et al.* 2007] J.J. Huang, Y.Z. Cai and X.M. Xu. *A hybrid genetic algorithm for feature selection wrapper based on mutual information.* Pattern Recognition Letters, vol. 28, no. 13, pages 1825–1844, 2007. 31

[Jain & Zongker 1997] A.K. Jain and D. Zongker. *Feature selection: Evaluation, application, and small sample performance.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, pages 153–158, 1997. 33

[Jolliffe 2002] I.T. Jolliffe. *Principal component analysis.* Springer series in statistics, 2002. 43, 71

[Jurie & Triggs 2005] F. Jurie and B. Triggs. *Creating efficient codebooks for visual recognition.* In Proceedings of the International Conference on Computer Vision, volume 1, pages 604–610, 2005. 50

[Kaniza 1997] G. Kaniza. *Grammatica del vedere.* Il Mulino, 1997. 56

[Keerthi & Lin 2003] S.S. Keerthi and C.J. Lin. *Asymptotic behaviours of support vector machines with Gaussian kernel.* Neural Computation, vol. 15, no. 7, pages 1667–1689, 2003. 73

[Kim *et al.* 2007] S.-J. Kim, K. Koh, M. Lustig, S. Boyd and D. Gorinevsky. *An interior-point method for large-scale ℓ_1-regularized least squares.* IEEE Journal of Selected Topics in Signal Processing, vol. 1, no. 4, pages 606–617, 2007. 85

[Kohavi & John 1997] R. Kohavi and G.H. John. *Wrappers for feature subset selection.* Artificial Intelligence, vol. 97, no. 1-2, pages 273–324, 1997. 30, 31

[Kojadinovic & Wottka 2000] I. Kojadinovic and T. Wottka. *Comparison between a filter and a wrapper approach to variable subset selection in regression problems.* In Proceedings of the European Symposium on Intelligent Techniques, pages 311–321, 2000. 31

[L. Fei-Fei & Perona 2004] R. Fergus L. Fei-Fei and P. Perona. *Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories.* In Workshop on Generative-Model Based Vision, IEEE Conference on Computer Vision and Pattern Recognition, 2004. 101, 109

[Lanckriet *et al.* 2004] G.R.G. Lanckriet, T.D. Bie, N. Cristianini, M. Jordan and W. Noble. *A statistical framework for genomic data fusion. Bioinformatics.* Bioinformatics, vol. 20, no. 16, pages 2626–2635, 2004. 25

[Lazebnik *et al.* 2006] S. Lazebnik, C. Schmid and J. Ponce. *Beyond bags of features: Spatial pyramid matching for recognition natural scene categories.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pages 2169–2178, 2006. vii, 49, 54, 55

[Leung & Malik 2001] T. Leung and J. Malik. *Representing and recognizing the visual appearance of materials using three-dimensional textons.* International Journal of Computer Vision, vol. 43, no. 1, pages 29–44, 2001. 20

[Lew *et al.* 2006] M.S. Lew, N. Sebe, C. Djeraba and R. Jain. *Content-based multimedia information retrieval: State of the art and challenges.* ACM Transactions on Multimedia Computing Communication and Applications, vol. 2, no. 1, pages 1–19, 2006. 2

[Lin & Lin 2003] H.T. Lin and C.J. Lin. *A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods.* Technical report, Department of Computer Science, National Taiwan University, 2003. 73

[Lindeberg 1998] T. Lindeberg. *Feature detection with automatic scale selection.* International Journal of Computer Vision, vol. 30, no. 2, pages 79–116, 1998. 10

[Liu & Yu 2005] H. Liu and L. Yu. *Toward integrating feature selection algorithms for classification and clustering.* IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 4, pages 491–502, 2005. 30

[Liu *et al.* 2009] J. Liu, Y. Yang and M. Shah. *Learning semantic visual vocabularies using diffusion distance.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 461–468, 2009. 51

[Logothetis & Sheinberg. 1996] N.K. Logothetis and D.L Sheinberg. *Visual object recognition*. Annual Review of Neuroscience, vol. 19, pages 577–621, 1996. 19

[Lowe 2004] David G. Lowe. *Distinctive image features from scale-invariant keypoints*. International Journal of Computer Vision, vol. 60, no. 2, pages 91–110, 2004. 16, 20

[Mairal *et al.* 2008a] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman. *Discriminative learned dictionaries for local image analysis*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, 2008. 5, 82, 83, 90

[Mairal *et al.* 2008b] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman. *Supervised dictionary learning*. In Proceedings of the Advances in Neural Information Processing Systems, 2008. 90

[Mairal *et al.* 2008c] J. Mairal, G. Sapiro and M. Elad. *Learning multiscale sparse representations for image and video restoration*. SIAM Multiscale Modeling & Simulation, vol. 7, no. 1, pages 214–241, 2008. 82

[Malioutov *et al.* 2005] D.M. Malioutov, M. Cetin and A.S. Willsky. *Homotopy continuation for sparse signal representation*. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 5, pages 733–736, 2005. 84

[Mallat & Zhang 1993] S.G. Mallat and Z.F. Zhang. *Matching pursuits with time-frequency dictionaries*. IEEE Transaction on Signal Processing, vol. 41, no. 12, pages 3397–3415, 1993. 82, 84

[Manjunath & Ma 1996] B.S. Manjunath and W.Y. Ma. *Texture features for browsing and retrieval of large image data*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, no. 8, pages 837–842, 1996. 14

[Mao 2004] K.Z. Mao. *Orthogonal forward selection and backward elimination algorithms for feature subset selection*. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 34, no. 1, pages 629–634, 2004. 31

[Marée *et al.* 2005] R. Marée, P. Geurts, J. Piater and L. Wehenkel. *Random subwindows for robust image classification*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 1, pages 34–40, 2005. 10

[Marszalek & Schmid 2006] M. Marszalek and C. Schmid. *Spatial weighting for bag-of-features*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2118–2125, 2006. 54

[Martinetz & Schulten 1991] T. Martinetz and K. Schulten. *A "Neural-Gas" network learns topologies*. Artificial Neural Networks, vol. I, pages 397–402, 1991. 59

[McCallum & Nigam 1998] A. McCallum and K. Nigam. *A comparison of event models for Naive Bayes text classification.* In AAAI-98 Workshop On Learning For Text Categorization, pages 41–48, 1998. 49

[Mikolajczyk & Schmid 2001] K. Mikolajczyk and C. Schmid. *Indexing based on scale invariant interest points.* In Proceedings of the International Conference on Computer Vision, volume 1, pages 525–531, 2001. 10

[Mikolajczyk & Schmid 2004] K. Mikolajczyk and C. Schmid. *Scale & affine invariant interest point detectors.* International Journal of Computer Vision, vol. 60, no. 1, pages 63–86, 2004. 20

[Mikolajczyk *et al.* 2005] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Gool. *A comparison of affine region detectors.* International Journal of Computer Vision, vol. 65, no. 1, pages 43–72, 2005. 10

[Mohan *et al.* 2001] A. Mohan, C. Papageorgiou and T. Poggio. *Example-based object detection in images by components.* IEEE Transactions On Pattern Analysis And Machine Intelligence, vol. 23, no. 4, pages 349–361, 2001. 19

[Moosmann *et al.* 2007] F. Moosmann, B. Triggs and F. Jurie. *Fast discriminative visual codebooks using randomized clustering forests.* In Proceedings of the Advances in Neural Information Processing Systems, pages 985–992, 2007. 50

[Narendra & Fukunaga 1977] P.M. Narendra and K. Fukunaga. *A branch and bound algorithm for feature selection.* IEEE Transactions on Computers, vol. 26, no. 9, pages 917–922, 1977. 31, 32, 43

[Navon 1977] D. Navon. *Forest before trees: The precedence of global features in visual perception.* Cognitive Psychology, vol. 9, no. 3, pages 353–383, 1977. 57

[Niblack *et al.* 1993] W. Niblack, R. Barber, W. Equitz, M. Fickner, E. Glasman, D. Petkovic and P. Yanker. *The QBIC project: Querying images by content using color, texture and shape.* volume 1908, pages 173–187, 1993. 18

[Nowak *et al.* 2006] E. Nowak, F. Jurie and B. Triggs. *Sampling strategies for bag-of-features image classification.* In Proceedings of the European Conference on Computer Vision, pages 490–503, 2006. 10

[Nowozin 2005] S. Nowozin. *Libsift - Scale-Invariant Feature Transform implementation.* http://user.cs.tu-berlin.de/ nowozin/libsift/, 2005. 65

[Olshausen & Field 1996] B.A. Olshausen and B.J. Field. *Emergence of simple-cell receptive field properties by learning a sparse code for natural images.* Nature, vol. 381, no. 6583, pages 607–609, 1996. 82

[Olshausen & Field 1997] B.A. Olshausen and B.J. Field. *Sparse coding with an overcomplete basis set: A strategy employed by V1?* Vision Research, vol. 37, no. 23, pages 3311–3325, 1997. 82

[Olshausen *et al.* 2001] B.A. Olshausen, P. Sallee and M.S. Lewicki. *Learning sparse image codes using a wavelet pyramid architecture.* In Proceedings of the Advances in Neural Information Processing Systems, volume 13, pages 887–893, 2001. 82

[Palmer 1977] S.E. Palmer. *Hierarchical structure in perceptual representation.* Cognitive Psychology, vol. 9, no. 4, pages 441–474, 1977. 19

[Papageorgiou & Poggio 2000] C. Papageorgiou and T. Poggio. *A trainable system for object detection.* International Journal of Computer Vision, vol. 38, no. 1, pages 15–33, 2000. 18

[Pass & R. Zabih 1997] G. Pass and J. Miller R. Zabih. *Comparing images using color coherence vectors.* In Proceedings of the ACM international conference on Multimedia, pages 65–73, 1997. 11, 45

[Pati *et al.* 1993] Y.C. Pati, R. Rezaiifar and P.S. Krishnaprasad. *Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition.* In Proceedings of the Asilomar Conference on Signals, Systems and Computers, volume 1, pages 40–44, 1993. 84, 95

[Pearson 1901] K. Pearson. *On lines and planes of closest fit to systems of points in space.* Philosophical Magazine, vol. 2, no. 6, pages 559–572, 1901. 71

[Perronnin & Dance 2007] F. Perronnin and C. Dance. *Fisher kernels on visual vocabularies for image categorization.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, 2007. 50

[Perronnin *et al.* 2006] F. Perronnin, C. Dance, G. Csurka and M. Bressan. *Adapted vocabularies for generic visual categorization.* In Proceedings of the European Conference on Computer Vision, volume 4, pages 464–475, 2006. 5, 50, 51, 52, 114

[Pudil *et al.* 1994a] P. Pudil, F.J. Ferri, J. Novovičová and J. Kittler. *Floating search methods for feature selection with nonmonotonic criterion functions.* In Proceedings of the International Conference on Pattern Recognition, pages 279–283, 1994. 33

[Pudil *et al.* 1994b] P. Pudil, J. Novovičová and J. Kittler. *Floating search methods in feature selection.* Pattern Recognition Letters, vol. 15, no. 11, pages 1119–1125, 1994. 31, 33, 43

[Pudil *et al.* 2002] P. Pudil, J. Novovičová and P. Somol. *Feature selection toolbox software package.* Pattern Recognition Letters, vol. 23, no. 4, pages 487–492, 2002. 32

[Pujol & Chen 2007] A. Pujol and L. Chen. *Line segment based edge feature using hough transform.* In Proceedings of the IASTED International Conference on Visualization, Imaging, and Image Processing, 2007. 61

[Quinlan 1986] J.R. Quinlan. *Induction of decision trees.* Machine Learning, vol. 1, no. 1, pages 81–106, 1986. 28, 32

[Quinlan 1993] J.R. Quinlan. C4.5: Programs for machine learning. Morgan Kaufmann Publishers, 1993. 28, 32

[Quinlan 1996] J.R. Quinlan. *Improved use of continuous attributes in C4.5.* Journal of Artificial Intelligence Research, vol. 4, pages 77–90, 1996. 32

[Raina *et al.* 2007] R. Raina, A. Battle, H. Lee, B. Packer and A.Y. Ng. *Self-taught learning: Transfer learning from unlabeled data.* In Proceedings of the International Conference on Machine Learning, volume 227, pages 759–766, 2007. 89

[Rakotomalala 2005] R. Rakotomalala. *TANAGRA : a free software for the education and the research.* Actes des 5ème journées Extraction et Gestion des Connaissances (EGC 2005), Revue des Nouvelles Technologies de l'Information (RNTI-E-3), vol. 2, pages 697–702, 2005. 44

[Rakotomamonjy *et al.* 2008] A. Rakotomamonjy, F.R. Bach, S. Canu and Y. Grandvalet. *SimpleMKL.* Journal of Machine Learning Research, vol. 9, pages 2491–2521, 2008. 26

[Rakotomamonjy 2003] A. Rakotomamonjy. *Variable selection using svm-based criteria.* Journal of Machine Learning Research, vol. 3, pages 1357–1370, 2003. 32

[Rao *et al.* 2008] S. Rao, R. Tron, R. Vidal and Y. Ma. *Motion segmentation via robust subspace separation in the presence of outlying, incomplete, and corrupted trajectories.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, 2008. 83

[Rodriguez & Sapiro 2007] F. Rodriguez and G. Sapiro. *Sparse representations for image classification: Learning discriminative and reconstructive non-parametric dictionaries.* Technical report, University of Minnesota, 2007. 90

[Rosenblatt 1962] F. Rosenblatt. Principles of neurodynamics: perceptrons and the theory of brain mechanisms. Spartan Books, 1962. 27

[Rothganger *et al.* 2006] F. Rothganger, Svetlana Lazebnik, Cordelia Schmid and Jean Ponce. *3D Object modeling and recognition using local affine-invariant image descriptors and multi-view spatial contraints.* International Journal of Computer Vision, vol. 66, no. 3, pages 231–259, 2006. 20

[Ruan *et al.* 2010] S. Ruan, N. Zhang, S. Lebonvallet, Q. Liao and Y. Zhu. *Fusion and classification of multi-source images by SVM with selected features in a kernel space.* In Proceedings of the International Conference on Image Processing Theory, Tools and Applications, 2010. 24

[Rubner *et al.* 2000] Y. Rubner, C. Tomasi and L.J. Guibas. *The earth mover's distance as a metric for image retrieval.* International Journal of Computer Vision, vol. 40, no. 2, pages 99–121, 2000. 51

[Saeys *et al.* 2007] Y. Saeys, I. Inza and P. Larranaga. A review of feature selection techniques in bioinformatics. Oxford University Press, 2007. 63

[Salton & McGill 1983] G. Salton and M.J. McGill. Introduction to modern information retrieval. McGraw-Hill, 1983. 20, 49

[Sayad *et al.* 2010] I. El Sayad, J. Martinet, T. Urruty and C. Djeraba. *Toward a higher-level visual representation for content-based image retrieval.* Journal of Multimedia Tools and Applications, pages 1–28, 2010. 2

[Schiele & Crowley 2000] B. Schiele and J. Crowley. *Recognition without correspondence using multidimensional receptive field histograms.* International Journal of Computer Vision, vol. 36, no. 1, pages 31–50, 2000. 18

[Schweizer & Sklar 1983] B. Schweizer and A. Sklar. Probabilistic metric spaces. North Holland, New York, 1983. 43

[Shafer 1976] G. Shafer. A mathematical theory of evidence. Princeton University Press, 1976. 4, 35

[Shakhnarovich *et al.* 2005] G. Shakhnarovich, T. Darrell and P. Indyk. Nearest-neighbor methods in learning and vision: theory and practice. MIT press, 2005. 28

[Shen & Bai 2004] L.L. Shen and L. Bai. *Adaboost gabor feature selection for classification.* Proceeding of Image and Vision Computing Conference NewZealand, pages 77–83, 2004. 71

[Snoek *et al.* 2005] Cees G. M. Snoek, Marcel Worring and Arnold W. M. Smeulders. *Early versus late fusion in semantic video analysis.* In Proceedings of the ACM International Conference on Multimedia, pages 399–402, 2005. 28, 29, 66

[Somol & Pudil 2000] P. Somol and P. Pudil. *Oscillating search algorithms for feature selection.* In Proceedings of the International Conference on Pattern Recognition, pages 406–409, 2000. 31, 34, 43

[Spence & Sajda 1998] C. Spence and P. Sajda. *The role of feature selection in building pattern recognizers for computer-aided diagnosis.* In Proceedings of SPIE, Vol. 3338, Medical Imaging 1998: Image Processing, pages 1434–1441, 1998. 31

[Starck *et al.* 2005] J. Starck, M. Elad and D. Donoho. *Image decomposition via the combination of sparse representation and a variational approach.* IEEE Transaction on Image Processing, vol. 14, no. 10, pages 1570–1582, 2005. 82

[Stearns 1976] S.D. Stearns. *On selecting features for pattern classifiers.* In Proceedings of the International Conference on Pattern Recognition, pages 71–75, 1976. 31, 33

[Stricker & Orengo 1995a] M.A. Stricker and M. Orengo. *Similarity of color images.* In Proceedings of the Storage and Retrieval for Image and Video Databases III, volume 2, pages 381–392, 1995. 12, 45

[Stricker & Orengo 1995b] M.A. Stricker and M. Orengo. *Similarity of color images.* In Proceedings of the Storage and Retrieval for Image and Video Databases (SPIE), pages 381–392, 1995. 61

[Swain & Ballard 1991] M.J. Swain and D.H. Ballard. *Color indexing.* International Journal of Computer Vision, vol. 7, no. 1, pages 11–32, 1991. 10, 94

[Takala *et al.* 2005] V. Takala, T. Ahonen and M. Pietikainen. *Block-based methods for image retrieval using local binary patterns.* In Proceedings of the Scandinavian Conference on Image Analysis, pages 882–891, 2005. 14, 94

[Tibshirani 1996] R. Tibshirani. *Regression shrinkage and selection via the lasso.* Journal of the Royal Statistical Society: Series B, vol. 58, no. 1, pages 267–288, 1996. 84

[Tishby *et al.* 1999] N. Tishby, F. Pereira and W. Bialek. *The information bottleneck method.* In Proceedings of the Allerton Conference on Communication, Control and Computing, pages 368–377, 1999. 50

[Trémeau *et al.* 2004] A. Trémeau, C. Fernandez-Maloigne and P. Bonton. Digital image processing, from acquisition to processing (in french). Dunod, 2004. 61

[Tuceryan & Jain 1993] M. Tuceryan and A.K. Jain. *Texture analysis.* In The Handbook of Pattern Recognition and Computer Vision (2nd Edition), pages 207–248, 1993. 12, 13, 45

[Ullman *et al.* 2001] S. Ullman, E. Sali and M. Vidal-Naquet. *A fragment-based approach to object representation and classification.* Proceedings of the International Workshop on Visual Form, pages 85–100, 2001. 19

[van de Sande *et al.* 2008] K.E.A. van de Sande, T. Gevers and C.G.M. Snoek. *Evaluation of color descriptors for object and scene recognition.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, 2008. 17

[van Gemert *et al.* 2008] J.C. van Gemert, J.M. Geusebroek, C.J. Veenman and A.W.M. Smeulders. *Kernel codebooks for scene categorization.* In Proceedings of the European Conference on Computer Vision, pages 696–709, 2008. vii, 52, 53

[van Gemert *et al.* 2010] J.C. van Gemert, C.J. Veenman, A.W.M. Smeulders and J.M. Geusebroek. *Visual word ambiguity.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 7, pages 1271–1283, 2010. 52

[Vapnik 1995] V. Vapnik. The nature of statistical learning theory. Springer News York Inc., 1995. 24

[Viola & Jones 2001] P. Viola and M. Jones. *Robust real-time object detection.* International Journal of Computer Vision, vol. 57, no. 2, pages 137–154, 2001. 18

[Vogel & Schiele 2004] J. Vogel and B. Schiele. *Natural scene retrieval based on a semantic modeling step.* In Proceedings of the International Conference on Image and Video Retrieval, 2004. 51

[Wang *et al.* 2001a] J.Z. Wang, J. Li and G. Wiederhold. *SIMPLIcity: semantics-sensitive integrated matching for picture libraries.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 9, pages 947–963, 2001. 44, 101

[Wang *et al.* 2001b] J.Z. Wang, J. Li and G. Wiederhold. *SIMPLIcity: Semantics-sensitive integrated matching for picture libraries.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 9, pages 947–963, 2001. 94, 101

[Weber *et al.* 2000] M. Weber, M. Welling and P. Perona. *Unsupervised learning of models for recognition.* In Proceedings of the European Conference on Computer Vision, pages 18–32, 2000. 19

[Wertheirmer 1923] M. Wertheirmer. *Untersuchungen zur lehre der gestalt II.* Psychologische Forschung, vol. 4, pages 301–350, 1923. 56

[Whitney 1971] A.W. Whitney. *A direct method of nonparametric measurement selection.* IEEE Transactions on Computers, vol. 20, no. 9, pages 1100–1103, 1971. 4, 30, 31, 43

[Winn *et al.* 2005] J. Winn, A. Criminisi and T. Minka. *Object categorization by learned universal visual dictionary.* In Proceedings of the International Conference on Computer Vision, volume 2, pages 1800–1807, 2005. 10, 50

[Won 2004] C.S. Won. *Feature extraction and evaluation using edge histogram descriptor in MPEG-7.* In Proceedings of the Advances in Multimedia Information Processing¨CPCM, volume 3333, pages 583–590, 2004. 15, 45

[Wright *et al.* 2009a] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. Huang and S. Yan. *Sparse representation for computer vision and pattern recognition.* In Proceedings of IEEE, volume 98, pages 1031–1044, 2009. 5, 82, 86

[Wright *et al.* 2009b] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry and Y. Ma. *Robust face recognition via sparse representation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 2, pages 210–227, 2009. 5, 82, 83, 86, 88, 91

[Yang & Honavar 1998] J.H. Yang and V. Honavar. *Feature subset selection using a genetic algorithm.* IEEE Intelligent Systems, vol. 13, no. 2, pages 44–49, 1998. 31, 33

[Yang *et al.* 2008a] J. Yang, J. Wright, T. Huang and Y. Ma. *Image super-resolution as sparse representation of raw image patches.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, 2008. 83

[Yang *et al.* 2008b] L. Yang, R. Jin, R. Sukthankar and F. Jurie. *Unifying discriminative visual codebook generation with classifier training for object category recognition.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8, 2008. 51

[Yang *et al.* 2009a] J. Yang, Y. Li, Y. Tian, L. Duan and W. Gao. *Group-sensitive multiple kernel learning for object categorization.* In Proceedings of the International Conference on Computer Vision, pages 436–443, 2009. 26

[Yang *et al.* 2009b] J. Yang, K. Yu, Y. Gong and T. Huang. *Linear spatial pyramid matching using sparse coding for image classification.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1794–1801, 2009. 89

[Zhang *et al.* 2000] D. Zhang, A. Wong, M. Indrawan and G. Lu. *Content-based image retrieval using gabor texture features.* In Proceedings of the IEEE Pacific-Rim Conference on Multimedia, pages 1139–1142, 2000. 14

[Zhang *et al.* 2007] J. Zhang, M. Marszalek, S. Lazebnik and C. Schmid. *Local features and kernels for classifcation of texture and object categories: A comprehensive study.* International Journal of Computer Vision, vol. 73, no. 2, pages 213–238, 2007. vii, 11, 50

[Zhu & Yuille 1996] S.C. Zhu and A. Yuille. *Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, no. 9, pages 884–900, 1996. 60

[Zhu *et al.* 2010] C. Zhu, H. Fu, C.E. Bichot, E. Dellandréa and L. Chen. *Visual object recognition using local binary patterns and segment-based feature.* In Proceedings of the International Conference on Image Processing Theory, Tools and Applications, pages 426–431, 2010. 94