UNIVERSIDADE TÉCNICA DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Object component models using Gabor filters for visual recognition

Plinio Moreno López
(Mestre)

Dissertação para a obtenção do Grau de Doutor em Engenharia Electrotécnica e de Computadores

Orientador: Doutor José Alberto Rosado dos Santos Victor
Co-orientador: Doutor Alexandre José Malheiro Bernardino

Júri

Presidente: Reitor da Universidade Técnica de Lisboa
Vogais:
Doutor Alberto Sanfeliu Cortés
Doutor Aurélio Joaquim de Castro Campilho
Doutor José Alberto Rosado dos Santos Victor
Doutor Mário Alexandre Teles de Figueiredo
Doutor Fabrizio Smeraldi
Doutor Alexandre José Malheiro Bernardino

Setembro de 2008

Abstract

The primate visual system is extremely successful and efficient in the challenging task of recognizing objects in complex scenes. A key component of the primate visual system is a massive utilization of neuronal circuits located in the low-level visual cortex areas, with responses similar to Gabor functions. These functions have important properties for image analysis such as selectivity to orientation, scale and frequency and being specially suited to characterize image texture.

In this thesis we explore the properties of Gabor functions in the context of component-based object recognition. Current component-based object recognition approaches represent objects as constellation of sub-parts, dividing the problem in three stages: interest region selection, image region description and, eventually, the recognition step. We introduce novel methods using Gabor filters for interest point selection and image region description. Performance is evaluated with state-of-the-art object recognition architectures.

Regarding the selection of interest points, we define a new top-down saliency function. We encode the appearance of object components in terms of Gabor filter responses to build the saliency function. This saliency function computes a wavelength profile for every component, being effective in filtering out clutter and noisy features. The aim of this function is to reduce the number of candidates for posterior analysis, but maintaining high recall rates.

Once the points of interest have been detected, we propose region descriptors with rich and efficient matching representations that explore the full set of parameters of Gabor filters. Local maxima of the filter energy response is the criterion to define two types of descriptors: a feature vector formed by Gabor filter responses that are chosen specifically for each object component and an alternative way to compute the SIFT descriptor.

We perform extensive tests in real scenarios, to show experimentally that our models for interest point selection and local descriptor computation are well suited for component-based object recognition. Results show that approaches based on Gabor filter responses outperform state-of-the-art approaches in several aspects of the object recognition problem.

Keywords: object recognition, Gabor filters, top-down saliency, component recognition, local descriptor, parameter selection

Resumo

O sistema visual dos primatas desempenha, com sucesso e rapidamente, tarefas complexas como o reconhecimento de milhares de objetos. Uma componente chave do sistema visual dos primatas é a utilização maciça de circuitos neuronais localizados em áreas corticais de baixo nível, com respostas semelhantes às funções de Gabor. As funções de Gabor têm propriedades importantes na análise de imagem, devido à sua selectividade à orientação, escala, e frequência, sendo particularmente adequadas à caracterização de texturas.

Nesta tese iremos utilizar as propriedades das funções de Gabor para abordar o problema do reconhecimento de objectos baseado em componentes. Os métodos clássicos para abordar este problema representam objectos como constelações de componentes e separam o problema em três fases: selecção das regiões de interesse, descrição das regiões de interesse, e finalmente a fase de reconhecimento. Nesta tese introduzimos métodos inovadores para os problemas da selecção de pontos de interesse e a descrição das regiões da imagem.

Relativamente à selecção de pontos de interesse, definimos uma nova função de saliência para cada componente de um objecto. Propõe-se uma representação da aparência dos componentes dos objectos baseada na resposta de filtros de Gabor para calcular a função de saliência. Estas funções discriminativas calculam um perfil do comprimento de onda para cada componente, e podem ser usadas para reduzir o efeito do ruído ou objectos estranhos, e para reduzir o número de candidatos, sem rejeitar os componentes do objecto que se pretendem reconhecer.

Estando os pontos de interesse detectados, a fase seguinte é o cálculo de descritores para cada região. É explorada a selecção automática dos parâmetros de filtros de Gabor, usando o critério de máximos locais da resposta da energia do filtro. Este critério é aplicado, a fim definir dois tipos de descritores: (i) Um vector de características composto pelas respostas de filtros de Gabor, escolhidos especificamente para cada componente do objecto, e (ii) uma maneira alternativa para calcular o descritor SIFT.

São feitas experiências de reconhecimento de objectos em cenários reais, que demonstram a boa aplicabilidade dos modelos propostos nesta tese para a selecção de pontos de interesse e cálculo de descritores locais. Os resultados demonstram um melhor desempenho das abor-

dagens baseadas nos filtros de Gabor quando comparadas com métodos dentro do estado da arte no reconhecimento de objectos.

Palavras chave: : Reconhecimento de objectos, filtros de Gabor, saliência, reconhecimento de componentes de objectos, descritor local, selecção de parâmetros

Agradecimentos

O primeiro agradecimento vai para o meu orientador científico José Santos-Victor. A forma clara e rigorosa de abordar os problemas, discussões, e sugestões, fazem que a productividade do Vislab aumente todos os dias. O segundo agradecimento é para o meu co-orientador científico Alexandre Bernardino. A forma de analisar problemas e propor novas ideas foi fundamental para a conclusão deste trabalho.

Agradeço ao Instituto de Sistemas e Robótica, pelas condições adequadas para realizar e conseguir trabalho científico de alta qualidade, e em particular aos directores durante a realização desta tese, João Sentiero, Maria Isabel Ribeiro e Vítor Barroso.

Nestes últimos cinco anos passaram pelo Vislab muitas pessoas que duma ou outra forma colaboraram no desfecho desta tese. Um ambiente de trabalho de alta qualidade é fundamental para conseguir resultados, razão pela qual agradeço sinceramente a todos os alunos, professores e visitantes do laboratório.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The human visual system has the astonishing capability of recognizing and categorizing thousands of objects in real time using just 2D image information [8]. This outstanding performance is likely to result mainly from the union of two capabilities: (i) parallel processing of a huge amount of low-level features, and (ii) developmental cognition processes. By contrast, the performance of computer vision approaches for object recognition is still not comparable with their biological analogies in these respects: real-time response, level of performance, and the ability to handle thousands of objects. Nevertheless, recent work in visual object recognition has made significant advances towards these goals.

Initial approaches to object recognition from images [100, 106, 92] proposed the use of appearance to model the image of an object as a whole. While this might seem a good idea to solve the object recognition problem, it is very susceptible to background clutter and object occlusions. These reasons led researchers to propose alternative approaches where appearance is computed only at selected (local) regions of the image. This so-called "component-based object recognition" approach has delivered various successful results [69, 107, 81, 121]. All these works exploit the idea of splitting the object in a group of components. Using this approach, the usual procedure consists of two steps: (i) selecting object components and representing the appearance of each individual component and (ii) combining the appearance models of multiple object parts to build the overall object model. Once the model is built, we are able to recognize objects by matching the model against novel images.

In this thesis we adopt the component-based approach for object recognition. We will use local filters inspired by the human visual system (Gabor functions) [20] to propose new methods in object component selection and local appearance representation.

1.1 What is component-based object recognition?

In order to illustrate the idea, Figure 1.1 shows some snapshots of the component-based object recognition steps. To recognize the woman in the Mona Lisa[1] oil painting, we would need to build a model, as follows:

1. Select the woman's most relevant points (i.e. salient points, interest points), plotted in red in the second image of Figure 1.1.

2. Select a neighborhood around each interest point, in order to define object components. We observe in the third image of Figure 1.1 shows the object components selected. Then, represent the appearance of each component, using a local descriptor (i.e. feature vector).

3. Build the object model by collecting the appearances of the components (parts). In addition to appearances, the model can use shape information (relative position between components). Appearance-only information allows the detection of the presence/absence of the object, but it is very difficult to retrieve its location unless shape information is also considered. In the rightmost image of Figure 1.1 we sketch Mona Lisa's model, which includes the individual components and spatial configuration of the parts, represented with lines between local regions.

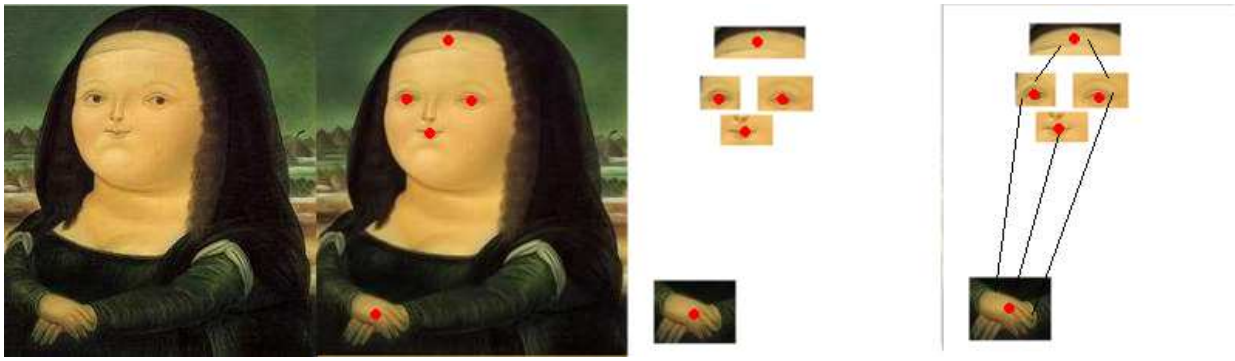


Figure 1.1: Component-based object recognition illustration

Considering the steps described, we need to answer the following questions:

- Which points in the object should be used (interest point selection)?

- How to select the interest point's neighborhood and model its appearance?

[1] by the Colombian artist Fernando Botero, 1977

- How to model the entire object, either considering the appearances alone or a joint description of shape and appearances?

We shall make a survey of the state of the art for these problems, referring to recent approaches that have had a significant impact in the computer vision field. First, we will look at the interest point selection, followed by the local appearance representation approaches and the global modeling frameworks for component-based object recognition.

1.1.1 Interest point selection

Macaques and humans have an impressive performance in real time interpretation of complex scenes. It seems that intermediate and high level processes in the visual system select just a few regions in the visual field to perform further processing, thus reducing the complexity of scene analysis. This interest point selection, often referred to as "salient point detection," can be driven in a bottom-up or top-down fashion [54].

Bottom-up approaches for saliency search for variability of low-level image attributes, such as contrast, color and texture, in order to find salient points. Thus, bottom-up approaches can be seen as "image-oriented" processes. In difference, top-down approaches for saliency search for regions in the images that are attached to a specific target object. For instance, when searching for faces in images we can guide the process, defining a saliency function specific to eye detection. Thus, top-down approaches can be labeled as "object-oriented" processes.

Although the presence of saliency mechanisms in the human visual system is clear, quantitative measures of saliency for different tasks remain unclear. One can design various kinds of saliency functions, based on what we want to "see." For example, if we want to have good regions for tracking we can try to minimize the matching error [81, 116], whilst if we want to find blobs at several scales, searching for local maxima of the Laplacian of Gaussian [69] is more appropriate. Saliency functions have been proposed to detect image features like corners [44], edges [11], ridges [66], and textured regions [50]. Other examples of saliency functions are local maxima of scale-Shannon entropy [51], and the "conspicuity map" [49] that combines color, intensity and orientation. All these examples of saliency functions exploit the bottom-up paradigm to select image interest points.

Although recent works consider mostly bottom-up saliency mechanisms, there is evidence of the interaction between bottom-up and top-down processes in nearly every search model in the human visual system [14]. The visual search of object components during the recognition process can be boosted with some prior (top-down) information about the regions we are searching. A recent work considers top-down saliency, defining the discriminant saliency

[39] concept, where salient points are extracted from the image features that enable best discrimination between one class (object) versus all the other classes. Thus, we have groups of salient points that are specific for each object.

Salient point detection is followed by appearance representation, also referred to as "local descriptor computation" in the computer vision literature.

1.1.2 Local image descriptors

Local image descriptors are vectors of features that characterize the vicinity of particular points of the image. Such descriptors are used to distinguish between different image patterns and (ideally) should be invariant with respect to a set of image transformations. Several types of local descriptors have been proposed in the literature: gradient magnitude and orientation maps [69], Gaussian derivatives [107, 81], rectangular features [118], differential invariants [57], steerable filters [36], Gabor features [113, 59, 88], cortex-like (HMAX-based) features [103], and the Scale Invariant Feature Transform (SIFT) [69], amongst others.

Local descriptors represent meaningful information of the neighborhood of an interest point, where "meaningful" varies according to the goals. For object recognition, the most critical requirement is that of allowing correct matches between corresponding regions of object images, irrespective of the transformations applied to the original image (e.g. affine, illumination, compression). In order to handle invariance to image transformations, three main approaches have been widely adopted: (i) constructing a descriptor using features whose response is invariant to image transformations [59, 107], (ii) conceiving interest point detectors that provide additional parameters (e.g. affine) that can be utilized to normalize image regions [81, 70, 83] and (iii) performing an exhaustive matching that considers a set of possible transformations [116, 109]. The first approach corresponds to a truly invariant local descriptor, the second approach assigns the invariance problem to the salient point detection, and the third approach assigns invariance to the matching procedure.

In terms of the operator applied to compute the representation, there are two main groups of descriptors: *filter-based* [36, 57, 113, 59, 103] and *histogram-based* descriptors [69, 83, 4, 55]. Filter-based approaches compute responses of operators (e.g. Gaussian derivatives, Gabor filters, HMAX, differential invariants) in order to build the local descriptors. Histogram-based operators (e.g. SIFT, shape context) compute spatial statistics of gradient responses to build the descriptors. Although histogram-based descriptors have been reported to outperform the filter-based counterparts in a matching experiment [83], the HMAX descriptor based on Gabor filter bank responses has been recently successfully used in recognition problems [109].

Another aspect that may have an important impact in the required computational and memory resources is the sampling method applied to compute the local descriptor. Existing

works have adopted either dense or sparse sampling approaches. Dense descriptors [68, 109] sample exhaustively the parameters of the operator used (e.g. scale, orientation) and pixels in the image neighbourhood under consideration. By contrast, sparse representations [113, 107, 47, 70] select particular parameter values and pixels (usually only at the interest point) to sample the response of the operators. The sampling criterion chosen is directly related to the descriptor size, a critical aspect in storage capabilities and matching time requirements.

We have seen how to detect salient points in an image, which are putative candidates for object components. Then, we described each region around the detected points for later recognition. It remains to see how to combine the descriptors of detected salient points, i.e. candidates for components, in order to recognize objects in images.

1.1.3 Models for object recognition

Component-based object models aim to combine the appearance of several components in order to represent each object as an entity. Probabilistic models for object recognition are the state-of-the-art techniques to tackle object detection and localization. The probabilistic formulation has an important property, the compatibility with machine learning algorithms. That property allows to handle object recognition in two stages: model estimation (i.e. training, learning) and model classification (i.e. recognition). The goal of the model estimation process is to compute the parameters' values that describes relevant information about the object class. Then, we match the model in new images, classifying possible new model instances as positive or negative.

Several probabilistic models have been proposed in the literature [19, 96, 10, 114, 70]. While some of these models use the local descriptors exclusively (i.e. appearance-only), there are some others that explicitly incorporate the pose between object components (i.e. shape-and-appearance). Intuitively, one cannot expect appearance information alone to allow the correct detection of objects. However, several works have reported very good performance [119, 19, 96, 109] using appearance-only models in cluttered images, due to the representation of the object and its "context" (background). The advantages of these methods include robustness to occlusions and non-rigid transformations of the object. A major drawback of appearance-only approaches is the difficulty of object localization in the image, although for some classes it is possible to estimate their positions with additional assumptions.

Shape-and-appearance models were originally proposed by Fischler et. al. as the "parts and structure model" [33]. The object consists of a set of templates (i.e. parts, components) arranged in some geometric configuration (i.e. structure, shape). Recent works have adopted this idea, considering several aspects of this approach, for example the constellation model [10], efficient matching with pictorial structures [47], and probabilistic Hough voting

[62], amongst others. Shape-and-appearance models are in general less robust to non-rigid transformations, but the shape is able to provide an estimate of the location of the object.

Regarding the number of objects that the model is able to consider in each class, the models are divided in two groups: (i) single object models, and (ii) category object models. Single object recognition approaches [86, 107, 70, 4] are designed for recognizing a single object instance, having very little or no intra-class variability. Instead, approaches for object categorization [47, 30, 121, 28] aim at grouping all similar objects in the same class and must handle larger in-class variability.

Additionally, the model estimation process can be carried out either in a supervised or in an unsupervised manner, depending on the amount of information available to the algorithm. In the case of single object recognition, a single sample can be used to compute the model, but for object categorization a large amount of segmented and/or labeled images is usually needed. There are several supervised approaches for object categorization that attain good performance [10, 95, 79, 121] at the cost of requiring a huge number of samples. Recently, weakly supervised approaches were proposed, where only image labeling is needed [30, 17]. Completely unsupervised approaches to image category detection [112, 9, 29, 25, 26] are preferable because they avoid both image segmentation and labeling.

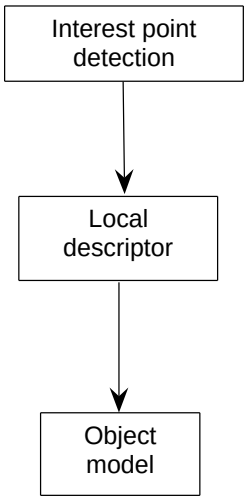


Figure 1.2: Main steps of component-based object recognition.

Figure 1.2 shows the steps required for component-based object recognition approaches that we have described. The main advantages of these approaches are: (i) invariance to rigid

image transformations [69, 29, 12], (ii) tolerance to objects occlusion in the image [69, 29], (iii) some degree of robustness to non-rigid image transformations [52, 81, 47, 29], and (iv) excellent performance in single-object recognition [69, 81] and categorization [29, 25, 69].

This thesis proposes ideas, methods and techniques for interest point detection and computation of local descriptors, as will be detailed in the next sections.

1.2 Approach of this thesis

In this thesis we focus on models and techniques for interest point detection and local descriptor computation for component-based object recognition that are largely based on Gabor filters. The choice of Gabor functions to perform computer vision and image processing tasks has been motivated by biological findings in the low-level areas of the primate visual cortex [20] and more recently by simulations of the primate/human visual system [49, 23].

A 2D Gabor function is formed by the product of a 2D Gaussian and a complex exponential function. Gabor functions act as low-level, oriented edge and texture discriminators that are sensitive to different frequencies and scale information. In an information theoretical sense, Gabor [38] has discovered that Gaussian-modulated complex exponentials provide the best trade-off between spatial and frequency resolution, allowing simultaneously good spatial localization and description of signal structures. Other interesting properties of the Gabor response are the invariance to changes in image contrast and robustness with respect to small translations of the image pattern in consideration.

Gabor filters have been widely used in numerous applications such as image compression [101], optical flow computation [45], disparity estimation [94, 99], texture segmentation [50, 24], human iris recognition [22], face recognition [122, 75, 68], and object recognition [5, 109]. Finally, the recent proposal of fast methods for Gabor filtering [7, 6] have further enhanced the feasibility of Gabor based recognition.

In this thesis, we use Gabor functions to build models for component-based object recognition for two main reasons:

- The first two steps of component-based object recognition (salient point detection and local descriptor computation) are low-level processes, where an analogy between Gabor filters and the low-level areas of the primate visual system can be established.
- Gabor filters have several degrees of freedom (i.e. function parameters) that have not been fully explored yet and can lead to simpler or more powerful models.

In the next sections we explain briefly our approaches for interest point selection and local descriptor computation using Gabor filters. We propose models to detect and describe object

components using Gabor filters. Then, we utilize these models to perform component-based object recognition.

1.2.1 Interest point selection

We start from the salient point detection, a problem that has been primarily addressed in a bottom-up way [69, 116, 81, 49, 52]. However, when searching the image for specific objects, it is convenient to incorporate object-related knowledge as early as possible in the recognition process, either to reduce the amount of possible candidates or to improve the recognition performance [39].

This thesis proposes an approach where saliency computation is biased to favor object related points, eliminating bottom-up salient points very different from the object related points and having very few misses of object points. This type of top-down saliency works as a refinement stage after the bottom-up interest point selection. Therefore, we manage to improve the efficiency in the subsequent steps of recognition by reducing the number of bottom-up interest point candidates for matching an object component.

The top-down saliency operator relies on the Gabor wavelength parameter that captures the texture information of an object's interest point. For every wavelength, the operator sums the contribution of all Gabor filter responses that were computed at that particular wavelength. Thus, the operator encodes the "wavelength signature" of an interest point, a coarse representation of an object component. The addition of the saliency model during the early stages of object recognition increases the efficiency of the entire process, reducing the number of component candidates for matching.

Additionally, the saliency operator is able to estimate the intrinsic scale of object components. The method proposed computes a very good approximation of the scaling factor between regions, having properties similar to those of the Laplacian of Gaussian, with added versatility to compute the intrinsic scale in ridge features.

The addition of the top-down saliency module modifies the architecture scheme during recognition, as can be seen in Figure 1.3.

1.2.2 Local descriptors

After detecting salient points, there is the problem of designing suitable local image descriptions. This aspect has been addressed in several recent works [69, 107, 81, 121, 4, 36], shifting the global matching problem to local matching.

We exploit the Gabor filter properties to define a filter-based local descriptor and a histogram-based descriptor. In both descriptors we explore the automatic parameter selection

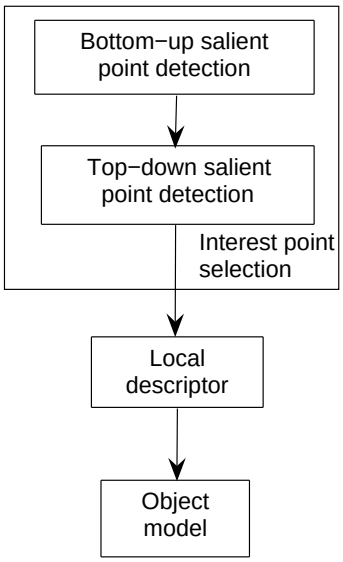


Figure 1.3: Architecture of our component-based object recognition approach

approach. The idea is to select the most adequate Gabor filters, using their response to define several criteria.

Filter-based local descriptor

The design of a local descriptor must choose the size of a feature vector in accordance to the requirements of a particular recognition problem. Dense descriptors may lead to very large feature vectors, possibly leading to prohibitive computational and storage needs. The HMAX [103] is an example of a state-of-the-art descriptor of this type, composed by a set of Gabor filter-bank responses computed at every pixel in a neighborhood across an exhaustive set of scales and orientations.

Instead, sparse descriptors sample the image responses computed in the object component image in a particular way, usually choosing responses only at the interest point. Sparse descriptors have shown good recognition rates in several applications [113, 122, 5, 47] along with very efficient matching methods. These properties lead us to adopt the sparse sampling approach for the Gabor filter-based descriptor.

A widely used, straightforward approach is to build local descriptors using Gabor filters responses, where the filter parameters are fixed [113, 24, 5, 76, 122, 109]. The adaptation of filter parameters to particular object components was first exploited in [53]. They propose to select Gabor function parameters in a semi-automatic fashion, using the local maxima of the "Information Diagram". The Information Diagram plots the magnitude of the Gabor

response for several scales and orientations at an interest point. We expand the concept of the Information Diagram to the Extended Information Diagram function, by adding the frequency parameter to the function. The set of parameters (scale, frequency and orientation) that best represent an object component corresponds to local maxima in the Extended Information Diagram function.

We model an object component by means of a feature vector formed from Gabor filter responses evaluated at the interest point of the object component. In order to compute Gabor responses, we use the parameters provided by the Extended Information Diagram. Thus, we utilize a filter bank matched to each object component in the sense of using the filter bank yielding the largest energy. This procedure automatically selects the adequate parameters of the Gabor filter to model a particular object component. We shall now see how to apply the parameter selection approach with histogram-based descriptors.

Histogram-based descriptors

Additionally, we explore the Gabor filter parameter selection to improve one of the most successful local descriptors presented in the literature, the SIFT [70] descriptor. Due to its particular way of extracting information from a local neighborhood, the SIFT descriptor has the best distinctiveness when compared to state-of-the-art local descriptors [83]. SIFT is a histogram-based descriptor that encodes local appearance using the image gradient in the neighborhood of an interest point. The local neighborhood is divided according to a cartesian grid and the histogram of the gradient orientation, weighted by its magnitude, is computed for each subimage.

SIFT uses the pixel differences to estimate the image gradient, a procedure sensitive to noise and other artifacts. Instead, we propose to use the Gabor filter parameter selection, picking the filter yielding the largest energy at every point in the neighborhood. In general, the use of odd Gabor filters instead of pixel differences to approximate first order image derivatives allows us to improve the distinctiveness of the SIFT local descriptor.

So far we have presented a top-down saliency computation method that is able to introduce specific information of the object components and local descriptors that represent component appearances. The remaining step is to choose an adequate object model for object recognition.

1.2.3 Object recognition tests

In this thesis we have proposed models for interest point selection and local descriptor computation. These models fit in the first two steps of component-based object recognition.

The remaining step for performing full object recognition is the choice of the model. In order to evaluate the performance of the proposed component models in different full object recognition problems, we consider both appearance-only and shape-and-appearance models.

All appearance-only object models share the same properties and drawbacks. However, a recent model based on cortex-like mechanisms [109] has demonstrated very good performance and versatility in various kinds of visual tasks. This appearance-only architecture uses a dense Gabor-filter-based representation for local descriptors, allowing us to compare the state-of-the-art filter-based descriptors against the histogram-based representations. Thus, within this framework, we compare SIFT [70], HMAX [103], and the SIFT improvement described in Section 1.2.2.

Regarding shape-and-appearance models, there are several approaches in the literature, so it is harder to compare methods in a qualitative manner. Nonetheless, the Pictorial Structure [47] includes various state-of-the-art properties: (i) joint estimation of shape and appearance, (ii) availability of efficient methods for matching, and (iii) robustness to partial occlusions of the object. We assess the following models presented in this thesis: (i) the top-down discriminant saliency, (ii) the filter-based, and (iii) the histogram-based descriptors.

The experiments in cluttered scenes show the capabilities of:

- the top-down saliency model, bringing efficiency to the subsequent steps of object recognition,
- the improved SIFT descriptor, increasing the matching capabilities of SIFT, and
- the HMAX-based descriptor and matching procedure, demonstrating that Gabor-based approaches are feasible in the object recognition context.

1.3 Contributions

The general contribution of this thesis is the construction of new models for component-based approaches to object recognition. These models are general in the sense that they can capture most of the interest regions that may appear in everyday images. More specifically, the contributions of this thesis are:

- A top-down saliency model that extracts low-level wavelength information of object components, reducing the computational complexity in the subsequent steps of object recognition.

- A method to define the intrinsic scale of an object component. This method relies on the wavelength profile function and is able to estimate the intrinsic scale in different image structures.
- A clear way to explore the parameter selection paradigm for Gabor filter functions, with the ability to construct two types of local descriptors, suitable for representing the appearance of object components.
- A filter-based local descriptor built with Gabor responses that selects the best parameters according to each image region to form an adaptive Gabor filter bank.
- An improved SIFT local descriptor using Gabor filter parameter selection to determine the best filter to compute first order image derivatives.

1.4 Thesis organization

This thesis is organized as follows: In Chapter 2 we tackle the interest point selection problem, introducing a top-down saliency point detection procedure that uses a frequency profile function. We also present a new way to compute the intrinsic scale of an image region, using this novel frequency profile function.

In Chapter 3 we introduce the parameter selection paradigm in order to build a filter-based local descriptor. We explain first how to build a good descriptor and we then add scale and rotation invariance to that descriptor.

In Chapter 4 we use the parameter selection paradigm in order to improve the SIFT local descriptor. We explain how to select the Gabor filters to compute first order image derivatives, the initial step of SIFT computation.

In Chapter 5 we perform an object category detection experiment using an appearance-only model to evaluate the improved SIFT descriptor, followed by an object detection and localization experiment that uses a shape-and-appearance model [47] in order to evaluate the top-down saliency model and the adaptive Gabor bank and SIFT descriptors.

In Chapter 6 we draw the thesis conclusions and establish directions of future work.

Chapter 2

Interest point selection

Component-based approaches for object recognition represent objects as collections of their parts. When searching for learnt objects, the selection of candidates for object components in new images is very important. Only "promising" points should be evaluated in the image; otherwise, in the case of unseen cluttered scenes, matching can be a very computationally expensive procedure. In order to avoid an exhaustive search, several authors utilize saliency operators that act like attentional mechanisms, concentrating computational resources on a few, highly promising points. The procedure to detect interest points can be oriented bottom-up or top-down. Bottom-up methods extract interest points using only image data criteria, while top-down methods also introduce task and context related information.

Most of the saliency functions proposed in the literature are bottom-up processes. They capture the variability of low-level signal attributes, like contrast, color, orientation or texture. This detection process does not rely on the information about the type of object to be recognized (the task).

Instead, top-down saliency methods are based on the task/goal description to guide the search process towards image regions that are likely to be parts of the sought objects. They are object or task-specific and require an initial learning phase, where the saliency filters are designed based on a number of samples of the object regions.

We propose a top-down saliency mechanism that operates over bottom-up interest points to vastly reduce the amount of candidates for matching/recognition. We design a novel saliency operator, conceived to encode object component information, which is based on the isotropic wavelength (texture) characteristics of the object component to detect. We explain how to compute and match the top-down saliency model for an object component and show how the proposed method is able to reduce significantly the number of candidates for recognition.

The properties of the saliency function are also exploited for the definition of a novel

method to compute the intrinsic scale of object components. This procedure provides scale invariance to the saliency model during the learning and detection processes. In the final part of this chapter we evaluate the performance of the top-down saliency method in a facial landmark candidate selection task.

2.1 Related work

We start with a brief revision of the most relevant approaches for interest point detection, considering both bottom-up and top-down approaches.

2.1.1 Bottom-up interest point selection

Initial approaches for bottom-up interest point selection focused on locating image features like edges and corners [44, 11]. The scale of the features was rarely addressed in these works and although spatial support was considered, it usually had a fixed value during interest point detection.

The spatial support of the feature is an important parameter that one must take into account. As pointed out by Crowley et al. [18], scale (i.e. spatial support) should be used as an additional parameter to build the shape model of an object. In that work, the scale-space structure of a particular image was utilized to build a graph that represents the shape of an object. The scale-space structure comprises location of peaks and ridges in the Difference of Gaussians [18] pyramid of the object, using both 2D (space) and 3D (scale-space) peaks.

However, it was only after the introduction of scale-space as the solution to the diffusion equation by Koenderink and Van Doorn [56, 58] that it was possible to define scale-invariant operators. Koenderink and Van Doorn introduced the Gaussian function and its derivatives of order n as solutions of the diffusion equation applied to images. Later, Lindeberg used the scale-space formulation in order to introduce a method for locating image features in scale-space [66]. Lindeberg's method provides (x_i, y_i, s_i) points (where x, y stand for position and s for scale) by computing local extrema of scale-normalized operators applied to the image $I(x, y)$. Lindeberg defines scale-normalized operators for blobs, junctions, edges, ridges and local frequency estimation [66]. An appropriate identification of the scale of the image features is essential to match object components correctly.

In practical terms, the scale invariance of any interest point detector does not guarantee flawless matching of object components. In order to assess matching capabilities, Mikolajczyk and Schmid propose the repeatability criterion [80]. The idea of the criterion is that once interest points are detected, the same points should be detected in any other image of the same

object, up to occluded regions. They evaluated the robustness to scale changes of several operators: the Difference of Gaussians, scale-space versions of image gradient, Laplacian, and Harris [44] (*cornerness*) function. The best repeatability was achieved by the scale-normalized Laplacian followed by the Difference of Gaussians (DoG) operator.

Interest point detection should cope with viewpoint transformations as well. Mikolajczyk and Schmid proposed in [82] an algorithm to provide local affine invariance properties to the scale-adapted Harris detector. The affine shape of a point neighborhood is estimated based on the windowed second-order moment matrix that computes an average of first order derivatives over the vicinity of an image point. The proposed algorithm uses the eigenvalues of the windowed second-order moment matrix as the initial values to search for the local affine parameters, such that two image patches related by an affine transformation become identical. They show experimentally that affine-covariant[1] regions can be matched in images with severe viewpoint transformations. In a more recent work, Mikolajczyk et al. [84] compare the repeatability of several affine region detectors, concluding that Maximally Stable Extremal Region (MSER) [78] and Hessian-affine [84] have better repeatability in average. MSER [78] are connected components of a thresholded image, whose pixels have either larger or smaller intensity than all pixels on its outer boundary. The Hessian-affine [84] detector selects local maxima of the Hessian matrix determinant and estimates the shape adaptation matrix in the points selected by the Hessian matrix. The Hessian-affine detector locates blobs and ridges covariant to affine transformations up to a rotation factor.

All previous methods rely on Gaussian derivatives to detect salient points in a bottom-up fashion. There are other bottom-up techniques based on different functions to detect interest points. Kadir and Brady [51] define salient points based on local maxima of Shannon entropy along several scales. In the case of pixel intensities, the Shannon entropy has small values in constant intensity regions, while it has larger values in image regions with high intensity variations. Later, Kadir et al. [52] added affine invariance to the salient point detection, showing better repeatability and matching results than Difference of Gaussians and Harris-affine detectors.

Itti et al. [49] propose a salient point detection for visual attention applications. They propose a biologically plausible architecture that builds a saliency map by applying "center-surround" filters sensitive to multiple scales, in color and intensity images and a Gabor filter bank sensitive to several scales and orientations. Then the information of all filters is combined across scales, building three "conspicuity maps" for intensity, color and orientation. The conspicuity maps are normalized and summed into the final saliency map. In the last step, they simulate visual attention by shifting the focus of attention to the most salient

[1] Corresponding regions in the two images are called covariant.

image location (i.e peak in the saliency map). After attending the current focus of attention, the location is inhibited in the saliency map to force an attention shift to the consecutive most salient point. This attention-based saliency mechanism has shown performance similar to the primate visual system for saliency-driven focal visual attention.

We have briefly presented bottom-up salient point detection approaches, applied to a variety of tasks. Bottom-up approaches are "general" salient point detection methods that can be shared across several computer vision tasks, performing the same computations regardless of the system's goal. Points selected by bottom-up techniques can be used in any high-level visual task, like shape recognition [85], object representation [70, 107, 81], visual attention [49, 120], wide baseline stereo matching [78], and mobile robot navigation [108]. Now we revise approaches where salient point detection is guided in a top-down manner by the task and context dependent criteria.

2.1.2 Top-down interest point selection

Top-down approaches for salient point detection use task-specific information to select the points of interest that are relevant to the task, neglecting the others. For instance, if we are searching for oranges, we should reject any points not having a strong enough red value in their color information.

Several top-down interest point selection methods have been proposed in the context of feature tracking, one of the most important tasks in computer vision [72, 35, 111, 41, 116]. The points selected for tracking are the local minima of the template matching cost function, which minimizes the error between image patches and a template, under a set of possible transformations applied to the template. Initial approaches by Förstner [35] and Harris [44] consider translation transformations of the patches. Later, Triggs [116] considers a wide range of patch transformations, including affine deformation and illumination changes. Triggs defines a reduced scatter matrix (reduced in the sense that it considers affine transformations only) that evaluates the self-matching properties of an interest point. Interest point selection relies on the minimum eigenvalue of the scatter matrix that reflects the maximum permissible errors in translation, rotation, and scale. We consider this interest point selection oriented top-down because interest point locations depend on the image transformation (i.e. motion) model. It is important to remark that the Harris corner detector [44] has been referred to as a bottom-up interest point detector in most of the works, but Triggs [116] presents an approach where the Harris detector is a particular case of a top-down salient point detector.

A recent work considers the idea of "discriminant saliency" [39], where the salient points are extracted from the features that enable best discrimination between one class and all other classes. Gao and Vasconcelos [39] compute a saliency map for all images in the pos-

itive class to select the most salient locations, their strength, and scales. The discriminant saliency map is a weighted sum of feature responses at every pixel. In order to consider discrimination power between classes, the weight of each feature is the marginal diversity [117]. Vasconcelos introduced the marginal diversity as a feature selection method, which under some assumptions selects a subset of the feature space that maximizes the mutual information of class labels and features. Thus, weights enforce features that provide the best recognition performance of a specific class. This top-down saliency procedure selects the best features (filter, location, and scale) for each class and can be viewed as a weakly supervised method to perform image segmentation of an image class.

While Gao and Vasconcelos' discriminant saliency selects salient features for each class, we propose an approach where saliency is guided by object components. We use the component's appearance to define a saliency operator that captures texture related low-level properties of an object part, creating a coarse component model. This saliency model selects only a limited number of image pixels as candidates for object components to recognize, thus discarding irrelevant information. This approach is also addressed in our previous publications [87, 89].

2.2 Using texture for component-based saliency

When searching for an object component, we propose to use its specific local texture characteristics as the main discriminant feature for selecting candidate points. Obviously, this does not prevent the use of other important feature dimensions (e.g. color), but here we are only considering gray-scale information. Gabor filters are among the most successful methodologies to extract texture information. After convolving an object component patch with a particular Gabor filter, we obtain a filter response that represents the amount of overlap between the texture represented by the filter and the texture in the object component. We will exploit the properties of Gabor Filters to represent texture and introduce top-down information to select object component candidates from a set of interest points.

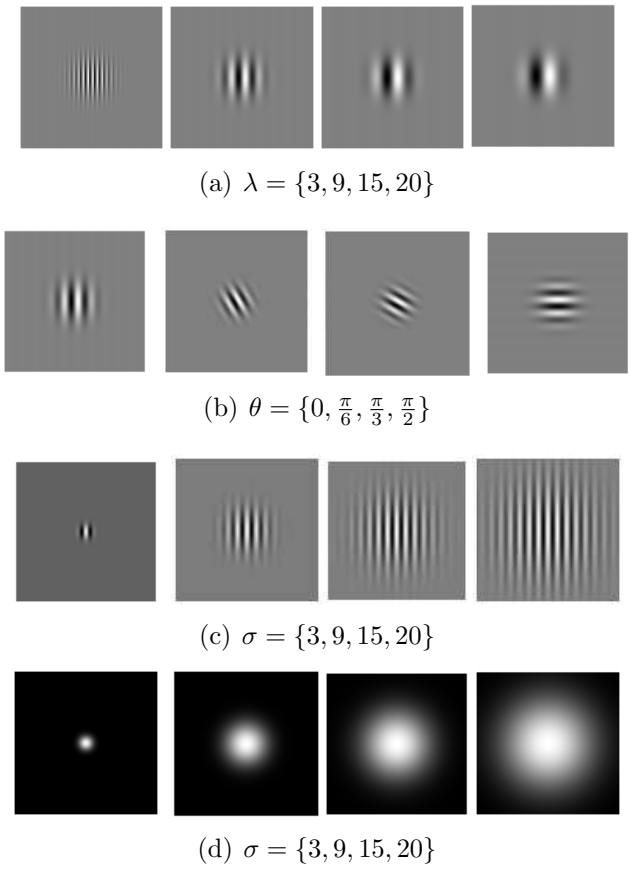
2.2.1 Gabor functions

The 2D zero mean isotropic Gabor function is expressed as:

$$g_{\theta,\lambda,\sigma}(x,y) = \frac{e^{-\frac{x^2+y^2}{2\sigma^2}}}{2\pi\sigma^2}\left(e^{\frac{j2\pi}{\lambda}(x\cos(\theta)+y\sin(\theta))} - e^{-\frac{2\sigma^2\pi^2}{\lambda^2}}\right), \tag{2.1}$$

where the parameters λ, θ, and σ are the wavelength (inverse of spatial frequency), orientation, and width (spatial support) of the Gabor function. Figure 2.1 shows the appearance of

some Gabor kernels as a function of σ, θ, and λ.



(a) $\lambda = \{3, 9, 15, 20\}$

(b) $\theta = \{0, \frac{\pi}{6}, \frac{\pi}{3}, \frac{\pi}{2}\}$

(c) $\sigma = \{3, 9, 15, 20\}$

(d) $\sigma = \{3, 9, 15, 20\}$

Figure 2.1: Examples of Gabor functions. Each row shows the real part of the Gabor function of Equation (2.1) for different values of λ, θ, and σ. The last row shows the magnitude of the filter for several widths. All images have equal size.

This conventional form of representing Gabor functions does not provide a simple representation of the filter scale. In fact, a scale change of the Gabor function must consider the recalculation of two parameters: the width and the frequency. We observe in Figure 2.1(c) that the visual aspect of the Gabor function changes severely by changing only the filter width value (σ). In order to localize scaled versions of a reference wavelet in the time-frequency plane, the wavelet theory [74, 73] defines a ratio that includes the scaling parameter and the center frequency, as follows:

$$\xi_n = \frac{\xi_0}{s}. \tag{2.2}$$

The Equation (2.2) expresses the center frequency of the scaled wavelet (ξ_n) as the ratio between the center frequency of the reference wavelet (ξ_0) and the scale parameter (s). We follow this reasoning and introduce the ratio between wavelength (multiplicative inverse of

the frequency) and width as a new parameter

$$\tilde{\lambda} = \frac{\lambda}{\sigma}, \tag{2.3}$$

proportional to the number of wave periods within the filter width. $\tilde{\lambda}$ is a scale invariant wavelength parameter. Substituting the expression for λ from Equation (2.3) into Equation (2.1), the Gabor function is reparametrized as

$$g_{\theta,\tilde{\lambda},\sigma}(x,y) = \frac{e^{-\frac{x^2+y^2}{2\sigma^2}}}{2\pi\sigma^2}\left(e^{\frac{j\pi}{\tilde{\lambda}\sigma}(x\cos(\theta)+y\sin(\theta))} - e^{-\frac{2\pi^2}{\tilde{\lambda}^2}}\right). \tag{2.4}$$

Figure 2.2 shows examples of Gabor functions of Equation (2.4) for a constant $\tilde{\lambda}$ value. In

(a) $\sigma = 5$ (b) $\sigma = 10$ (c) $\sigma = 15$

Figure 2.2: Examples of Gabor functions of Equation (2.4), using $\tilde{\lambda} = 1/3$

that figure we note that fixing $\tilde{\lambda}$ keeps the shape of Gabor functions constant, so that the number of wave periods within the filter width is 1.5 regardless of the filter width. Thus, by keeping the value of $\tilde{\lambda}$ constant, the filter appearance is maintained for different σ.

The Fourier analysis techniques denote Gabor functions as time-frequency atoms [73] (time switches to space on images), due to the concentration of their energy in Heisenberg boxes. A Heisenberg box (given by its center, time spread and frequency spread) provides the resolution of a Gabor function in the time-frequency plane and is defined by the parameters of the function. Figure 2.3 illustrates the Heisenberg boxes of a 1D Gabor function and its scaled version. The center frequencies of those boxes are related as presented in Equation (2.2).

The Heisenberg box with parameters ξ_0 and σ of Figure 2.3 is analogously related to the isotropic 2D Gabor function with parameters $\lambda = 2\pi/\xi_0$ and σ in Equation (2.1). A time-frequency Gabor atom extract the energy of a well localized part of the spectrum that are particular textured patterns parametrized by λ and σ. In addition, the texture orientation is provided by the angle θ.

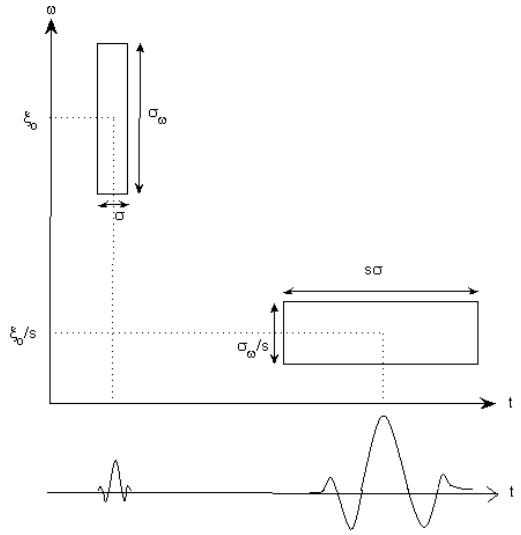


Figure 2.3: Example of the time-frequency localization of the Heisenberg boxes of 1-D Gabor atoms. The reference Gabor function (bottom left) with its correspondent Heisenberg box (top left) and the scaled Gabor (bottom right) with its respective box. σ_ω is the frequency width, σ is the time width, ξ_0 is the center frequency of the reference wavelet, and s is the scale factor.

2.2.2 Representing texture of object components

The convolution of the Gabor function with the image $I(x, y)$ allows us to extract the energy of a well localized part of the spectrum. The Gabor response at object component location (x_c, y_c) is

$$(g_{\theta,\lambda,\sigma} * I)(x_c, y_c) = \int\int I(x, y) g_{\theta,\lambda,\sigma}(x_c - x, y_c - y) dx dy. \tag{2.5}$$

The parameters σ, λ, and θ characterize the dominant texture of the object component c. One approach to select the texture that characterizes that object component would be to compute the response of several Gabor filters, tuned to different orientations, wavelengths, and widths and retain the parameters corresponding to the maximum response:

$$(\hat{\sigma}_c, \hat{\lambda}_c, \hat{\theta}_c) = \arg\max_{\sigma,\lambda,\theta} |(g_{\theta,\lambda,\sigma} * I)(x_c, y_c)|. \tag{2.6}$$

The set of parameters provided by the Equation (2.6) define a particular Gabor function that captures the object component appearance as an oriented texture. However, if we apply a geometric transformation to the component, we obtain a different set of parameters. Thus, the obtained texture description is not invariant to the orientation and spatial support (analysis window) of the object component. In the next sections we introduce a Gabor-based

texture representation invariant to common 2D geometric transformations.

2.3 Component invariant texture: the λ-signature

The parameters of the oriented texture selected by Equation (2.6) will change whenever the image is subjected to geometric transformations. In order to obtain invariance to rotation and size of the analysis window, we proceed as follows:

1. Integrate the response of the Gabor filters for all orientations and spatial supports:

$$S_w(x_c, y_c, \lambda) = \int_0^{\infty} \int_{-\pi}^{\pi} (g_{\theta,\lambda,\sigma} * I)(x_c, y_c) d\theta d\sigma, \tag{2.7}$$

   where S_w stands for Gabor wavelength saliency function. At component point (x_c, y_c), this is a function of the wavelength only and for each value of λ, S_w sums the contribution of all Gabor filter responses, which were computed with that particular wavelength λ. Thus, the S_w function can be viewed as the λ-*signature* of the component under analysis. This function will give us the "energy" of the object component for any wavelength of interest, independently of its orientation and spatial support.

2. The λ-signature of an object component is independent of the orientation and extent of the analysis window, but it is not scale invariant. If we compute the λ-signature in a rescaled version of the image, the signature amplitude and location in λ axis will change. To overcome this problem we need to compute the intrinsic scale of the object component and use this parameter to normalize the λ-signature. Finally, we map λ-signature to scale invariant values, using the scale invariant wavelength $\tilde{\lambda}$ as given in Equation (2.3).

We could have selected a standard isotropic filter like the Laplacian of Gaussian (i.e. Mexican Hat) to extract the texture characteristics. However, the wavelength saliency function of Equation 2.7 is built on very well localized time-frequency Gabor atoms, so the S_w function extracts a very well localized energy spectrum in the time-frequency plane for all wavelength values. In difference, the energy spectrum of the Laplacian of Gaussian has a large overlap even at small wavelengths (high frequencies), thus extracting similar information at that range. To illustrate the difference in overlap, Figure 2.4 shows the Fourier transform of two LoG functions and two S_w functions.

In the next sections we will explain in detail the two steps briefly described above.

(a) Laplacian of Gaussian functions (b) S_w functions

Figure 2.4: Magnitude of the Fourier transform of two LoG functions (left side) and two S_w functions (right side). The center frequency of both type of functions is the same, so the S_w functions have a better selectivity of the energy spectrum.

2.3.1 The "Gabor wavelength saliency" operator

To compute the λ-signature of an object component one could use the direct implementation of Equation (2.7). However this would require a significant amount of computation. To overcome this problem we define an equivalent kernel that filters the image just once for each wavelength. The equivalent kernel is obtained by summing the Gabor kernels for all spatial supports and orientations and is denoted "Gabor wavelength Saliency" kernel,

$$w(x_c, y_c, \lambda) = \int_0^{\infty} \int_{-\pi}^{\pi} g_{\sigma,\theta,\lambda}(x_c, y_c) d\theta d\sigma. \tag{2.8}$$

The closed form expression for the wavelength-space kernel is the following:

$$w(r, \lambda) = \frac{\sqrt{\pi/2}}{r} \left(-e^{\frac{-2\pi r}{\lambda}} + J_0\left(\frac{2\pi r}{\lambda}\right)\right). \tag{2.9}$$

where, $r = \sqrt{x_c^2 + y_c^2}$, and $J_0(z)$ is the Bessel function of the first kind. Looking at Equation (2.9), the equivalent kernel is an exponentially decreasing 2D Bessel function and it is rotationally invariant because it is explicitly expressed in terms of r.

The kernel computation from Equation 2.8 and Equation 2.9 assume spatial support values not present in discrete images (e.g. 0, ∞). Considering the resolution limits in discrete images, the lower and upper limits of the spatial support (σ) in Eq.(2.8) cannot cover the whole interval $[0, \infty)$. We use image resolution constraints to define the adequate σ integral limits in Eq. 2.8. In the case of the lower σ limit of the integral in Eq. (2.8), we consider

that the Gabor wavelength should not be greater than the Gaussian envelope ($\lambda \leq 6\sigma$), so $\sigma \geq \frac{\lambda}{6}$, otherwise no significant texture information is provided. In the case of the upper σ limit of the integral, we first find an appropriate minimum $\tilde{\lambda}$ using the expression

$$\tilde{\lambda} = \lambda/\sigma.$$

Due to the discrete nature of images, the wavelength value is provided by the Nyquist sampling ($\lambda = 2$). To sample adequately a Gabor filter with $\lambda = 2$, the filter width must be greater than 1 ($\sigma > 1$). We choose $\sigma = 2$ and replacing both $\sigma = 2$ and $\lambda = 2$ values yields $\tilde{\lambda} = 1$. Having the minimum $\tilde{\lambda}$, the upper integral limit is $\sigma = \lambda$. Recomputing the Equation (2.8) with the new integral limits, the equivalent kernel for the wavelength signature is:

$$w_d(x, y, \lambda) = \int_{\lambda/6}^{\lambda} \int_{-\pi}^{\pi} g_{\sigma,\theta,\lambda}(x, y) d\theta d\sigma, \tag{2.10}$$

where w_d stands for the equivalent kernel for discrete images. The closed form expression

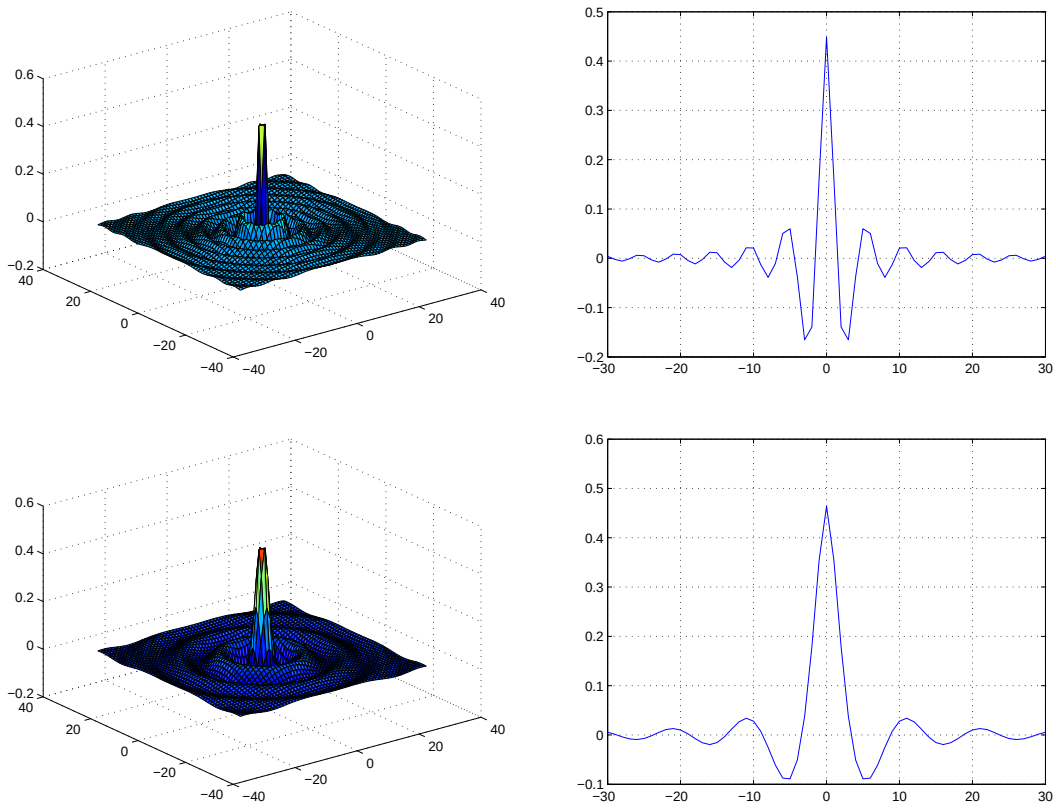


Figure 2.5: Example of λ-signature equivalent kernel. Top figures, 3D plot and 1D slice of $w_d(x, y, 5)$. Bottom figures, 3D plot and 1D slice of $w_d(x, y, 10)$

of Equation (2.10) is presented in appendix A.1 and Figure 2.5 shows an example of w_d. Therefore, the computation of the λ-signature of the object component located at point (x_c, y_c) can be efficiently performed by:

$$S_{w_d}(x_c, y_c, \lambda) = I * w_d. \tag{2.11}$$

We utilize Equation (2.11) to compute the signature of object components in discrete images. Even though the λ-signature is invariant to image rotations, image scale changes will cause the λ-signature to translate in the wavelength axis, and its amplitude will change linearly with the scale factor. Now we tackle the problem of ensuring scale invariance for the λ-signature in Equation (2.11).

2.4 Providing scale invariance for the λ-signature

We analyze first the behavior of the λ-signature amplitude under scale changes. To obtain a coefficient that performs scale-normalization of the amplitude of the λ-signature, we follow Lindeberg's idea to provide a scale normalization for features [66]. The rationale is to find a normalization factor specific for each feature, proportional to the width of the operator. The second step to normalize the λ-signature consists in mapping λ values to the scale invariant wavelength parameter $\tilde{\lambda}$.

2.4.1 Amplitude normalization

In order to normalize signature amplitude, let us consider two images: the initial image $I(x, y)$ and an homogeneously scaled version of the initial image, $I_s(x, y)$. The new image is scaled by a factor a: $I_s(x, y) = I(ax, ay)$. The λ-signature at $I_s(x_c, y_c)$ is:

$$\begin{aligned} S_w^{I_s}(x_c, y_c, \lambda) &= I_s * w = w * I_s \\ &= \int\int w(x, y, \lambda) I_s(x_c - x, y_c - y) dxdy \\ &= \int\int w(x, y, \lambda) I(ax_c - ax, ay_c - ay) dxdy \end{aligned}$$

Now let $\breve{x} = ax$, $\breve{y} = ay$, and $\breve{\lambda} = \lambda a$. Then $dx = d\breve{x}/a$, and $dy = d\breve{y}/a$. By making substitutions in the equivalent kernel of Equation (2.9),

$$\begin{aligned} S_w^{I_s}(x_c, y_c, \lambda) &= (I * S_w)(ax_c, ay_c, \lambda a) \\ &= \frac{\lambda}{\breve{\lambda}} S_w^I(ax_c, ay_c, \lambda a) \end{aligned} \tag{2.12}$$

finally yielding:

$$\frac{1}{\lambda} S_w^{I_s}(x_c, y_c, \lambda) = \frac{1}{\breve{\lambda}} S_w^I(ax_c, ay_c, \lambda a). \tag{2.13}$$

From Equation (2.13) we see that if we multiply the response of the kernel by the inverse of the wavelength, the λ-signature amplitude becomes normalized with respect to scale changes. Thus, the amplitude normalized λ-signature at point (x_c, y_c), is:

$$S_w^{norm}(x_c, y_c, \lambda) = \frac{1}{\lambda}(I * w_d). \tag{2.14}$$

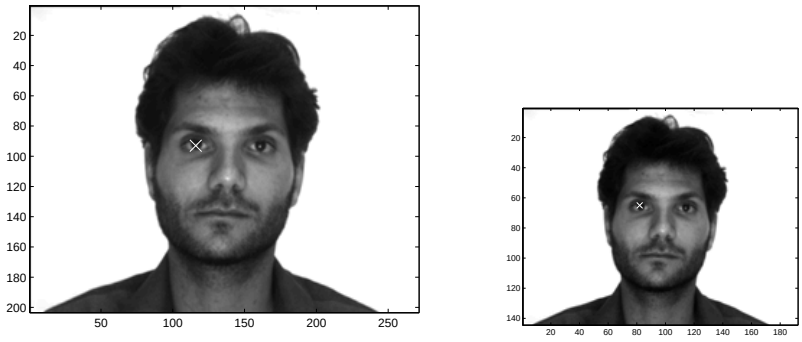
Equation (2.14) introduces the scaling factor $1/\lambda$, which guarantees theoretically the same amplitude of the λ-signature for two images with different scales. But in real images, the amplitudes will have very similar values and not the same value due to the discretization effects of image subsampling on high frequencies. Figure 2.6 illustrates the effect of the scale normalization of S_w amplitude by plotting both S_w and S_w^{norm} at an eye's center point of scaled images. We observe in Figure 2.6(b) a larger difference between the response of S_w in scaled images, while in Figure 2.6(c) the normalized response of S_w^{norm} is very similar between scaled images. However, if we want to match the signatures plotted in Figure 2.6, it is necessary to warp one of the signatures before matching. To overcome this problem we need to compute the intrinsic scale of the interest point and use this parameter to map λ-signature to scale invariant values.

2.4.2 Scale normalization

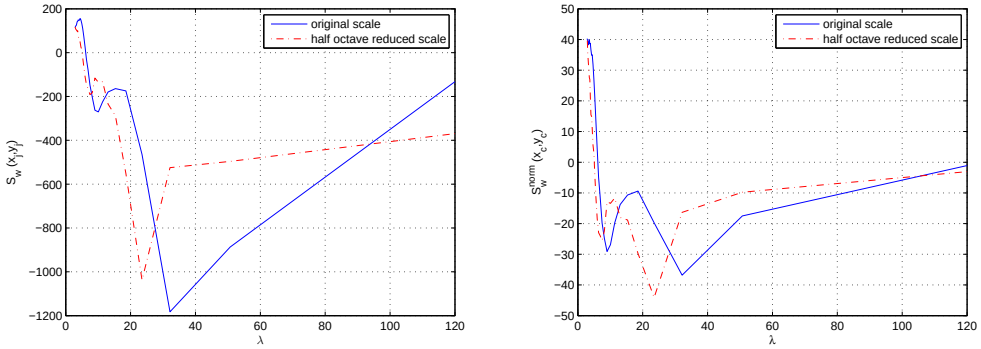
Finally, to obtain a scale invariant signature, we have to map the λ-signature function using the intrinsic scale ${\sigma_{\text{int}}}^2$, as the normalization parameter:

$$\tilde{\lambda} = \frac{\lambda}{\sigma_{\text{int}}}. \tag{2.15}$$

[2]The intrinsic scale computation will be explained in the next section



(a) Left column, original image. Right column, scaled image



(b) S_w at left eye point of images in (a) (c) S_w^{norm} at left eye point of images in (a)

Figure 2.6: Example of S_w and S_w^{norm} for an eye's center point

In Equation (2.3) we presented $\tilde{\lambda}$ as the scale invariant wavelength parameter of the Gabor function. We apply the same definition, but replacing the filter width σ with the intrinsic scale σ_{int} in order to map λ values to the scale invariant $\tilde{\lambda}$ values.

To determine the signature for a given interest point, we should consider a range of wavelength values, $\Lambda = \{\lambda_1, \ldots, \lambda_k, \ldots, \lambda_K\}$. This set can now be normalized with respect to scale changes according to the procedure we described:

$$\tilde{\Lambda} = \{\tilde{\lambda}_1, \ldots, \tilde{\lambda}_k, \ldots, \tilde{\lambda}_K\} = \{\lambda_1/\sigma_{\text{int}}, \ldots, \lambda_k/\sigma_{\text{int}}, \ldots, \lambda_K/\sigma_{\text{int}}\}.$$

Finally, the $\tilde{\lambda}$-signature (top-down saliency model) of the component located at point (x_c, y_c), encompassing this set of wavelength values, is denoted as $\tilde{\Lambda}S$ and defined according to:

$$\tilde{\Lambda}S_{x_c,y_c}(\tilde{\lambda}_k) = S_w^{norm}(x_c, y_c, \lambda_k/\sigma_{\text{int}}), \tilde{\lambda}_k \in \tilde{\Lambda}. \tag{2.16}$$

To illustrate the scale normalization procedure presented in this section, we show in Figure

2.7: (i) the $\lambda-$signature, (ii) the amplitude-normalized $\lambda-$signature, and (iii) the fully scale-normalized $\lambda-$signature. The small difference of the signatures in Figure 2.7(d) is caused by the discretization effects of the scaling procedure. There is a remaining issue to consider in order to match the signatures of components: the intrinsic scale computation. This is the topic of the following section.

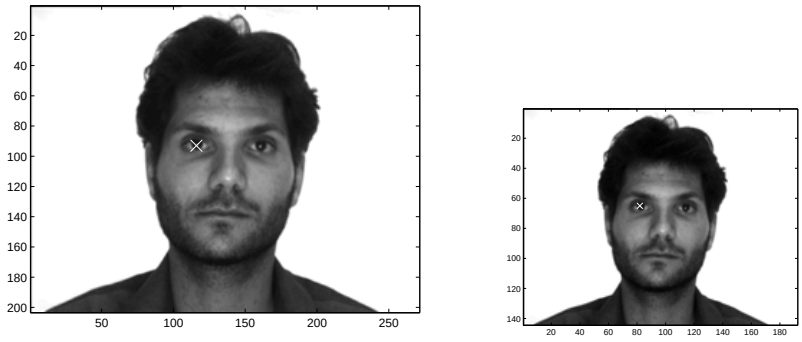
2.5 Intrinsic scale from the λ-signature

The work described in [66, 65] proposed an automatic selection of scale for several image features: blobs, junctions, edges, and ridges. The procedure is to look for local extrema in scale-space, using scale-normalized operators to find features' intrinsic scale, without any prior knowledge of feature size. The intrinsic scale is characteristic of a given texture and changes (proportionally) when the image scale is modified. As such, it allows us to track scaling modifications of a textured pattern through adequate normalization. We could use any of the operators proposed in [66, 65], but we noticed experimentally that the zero crossings of the λ-signature function closest to the global maxima are very stable under image scale changes and are directly proportional to the scale factor. Thus, we compute the intrinsic scale σ_{int} at object component point (x_c, y_c) as:

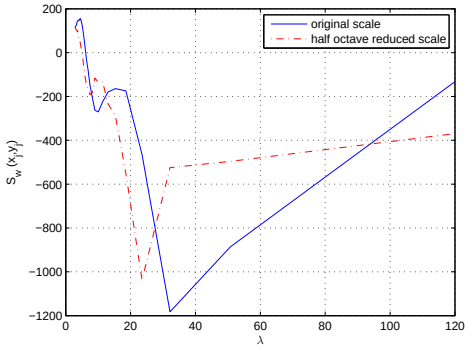
$$\sigma_{\text{int}} = \arg\min_{\lambda\in\lambda_0} |\lambda - \hat{\lambda}|; \quad \lambda_0 = \{\lambda : S_w^{norm}(x_c, y_c, \lambda) = 0\}; \quad \hat{\lambda} = \arg\max_{\lambda} |S_w^{norm}| \tag{2.17}$$

Figure 2.8 illustrates the similarity between the λ-signature kernel w and the Laplacian of Gaussian (LoG) kernel defined by Equation (2.18). Notice that apart from the magnitude and sign inversion, the two functions are very similar, but the λ-signature kernel is sharper at the origin. The extrema points of the LoG response provide the location of blobs in images. With the addition of a scale-normalization factor (σ^2 in Equation 2.19), the extrema points of the LoG response compute the intrinsic scale of blob-like image structures (Equation 2.19).

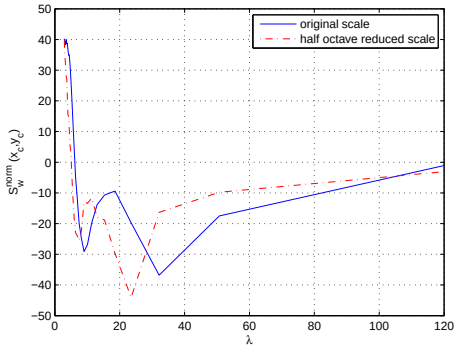
$$\begin{aligned}
G_w(x, y, \sigma) &= \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \\
LoG_w(x, y, \sigma) &= \nabla^2 G_w = \frac{\partial^2 G_w}{\partial x^2} + \frac{\partial^2 G_w}{\partial y^2} = -\frac{1}{\pi\sigma^4}\left(1 - \frac{x^2+y^2}{2\sigma^2}\right) e^{-\frac{x^2+y^2}{2\sigma^2}}, \qquad (2.18) \\
LoG^{norm}(x, y, \sigma) &= \sigma^2 I * LoG_w(x, y, \sigma), \\
\sigma_{\text{int}}^{LoG} &= \arg\max_{\sigma} |LoG_{norm}|. \qquad (2.19)
\end{aligned}$$



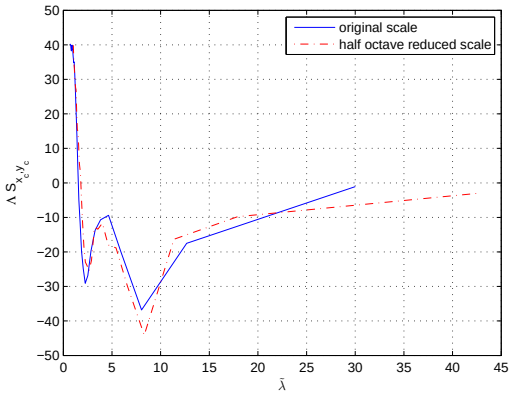
(a) Left column, original image. Right column, scaled image



(b) S_w at left eye point of images in (a)



(c) S_w^{norm} at left eye point of images in (a)



(d) $\tilde{\Lambda}S$ of left eye

Figure 2.7: Example of scale invariant signature, $\tilde{\Lambda}S$.

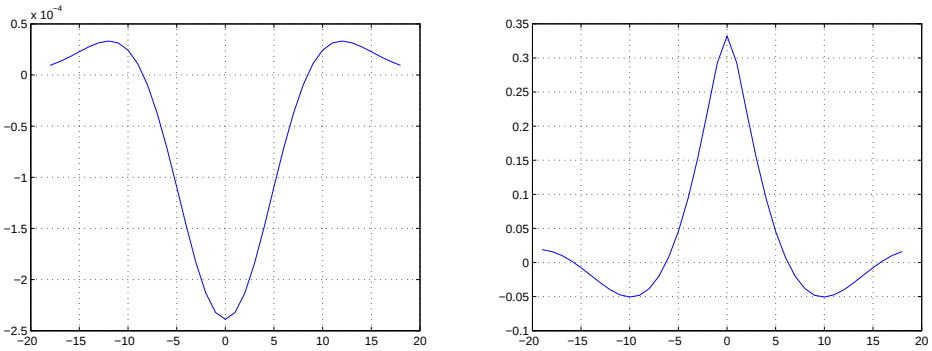


Figure 2.8: In left side we plot the 1D cut $LoG(0, y, 6)$, in the right side we plot the 1D cut $S_w(0, y, 18)$

2.5.1 Evaluation in synthetic images

We proceed now with a comparison between the proposed and classical forms of intrinsic scale computation, respectively S_w^{norm} and LoG_{norm}. We compute the intrinsic scale in the center of a circle image and in the center of a ridge image, which we show in Figure 2.9. The intrinsic scale is computed by using the S_w^{norm} zero crossing in Equation (2.17) and the global maximum of $|LoG_{norm}|$ in Equation (2.19). Each image is subjected to a scale change of a factor of 2. A correct intrinsic scale computation method should be able to obtain the scale factor between images, by computing the ratio between intrinsic scales.

(a) $r = 6$ (b) $r = 12$ (c) width= 8 (d) width= 16

Figure 2.9: Circle and ridge synthetic images. Parameter values are in pixels.

In order to illustrate the intrinsic scale computation, we present in Figures 2.10 and 2.11 the S_w^{norm} and LoG_{norm} curves at the center point of the synthetic images, and the respective intrinsic scale values in Table 2.1. As expected, LoG_{norm} computes a correct intrinsic scale ratio in circle images because this operator was conceived to detect blobs. Even though the S_w^{norm} scale ratio is not exact, it is a very good approximation to the correct scale ratio. In the case of ridge images, we see in Table 2.2 that S_w^{norm} intrinsic scale ratio is closer to the real scale factor, while LoG_{norm} scale ratio is farther from the real scale factor. The reason

	Intrinsic scale S_w^{norm}	Intrinsic scale LoG_{norm}
circle $r = 6$ pixels	9.15	4
circle $r = 12$ pixels	18.01	8
intrinsic scale ratio	1.97	2

Table 2.1: Intrinsic scale at center point of circle images in Figure 2.9

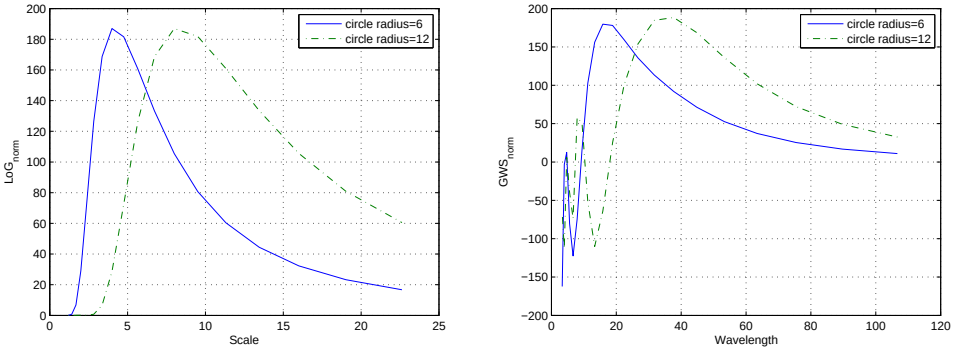


Figure 2.10: LoG_{norm} (left side) and S_w^{norm} (right side) for the circle images in Figure 2.9.

for this is that LoG_{norm} has a good behavior only with blobs. On the other hand, S_w^{norm} computes adequate intrinsic scales for both circles and ridges.

	Intrinsic scale S_w^{norm}	Intrinsic scale LoG_{norm}
ridge width= 8 pixels	8.42	$4 \cdot 2^{1/4}$
ridge width= 16 pixels	15.67	8
intrinsic scale ratio	1.86	1.68

Table 2.2: Intrinsic scale at center point of ridge images in Figure 2.9

To summarize, we have presented a synthetic image test to see experimentally the advantages of the intrinsic scale computation by searching for the zero cross closest to the global maximum of $|S_w^{norm}|$, presented in Equation (2.17). We propose a method that provides a good approximation to the correct scale factor between images, with higher versatility than LoG_{norm}. Using the intrinsic scale σ_{int} of Equation (2.17), we can now map the λ values to $\tilde{\lambda}$ values, using the Equation (2.15). A more thorough analysis of the proposed intrinsic scale computation method is provided in Section 2.7 and includes the use of real images.

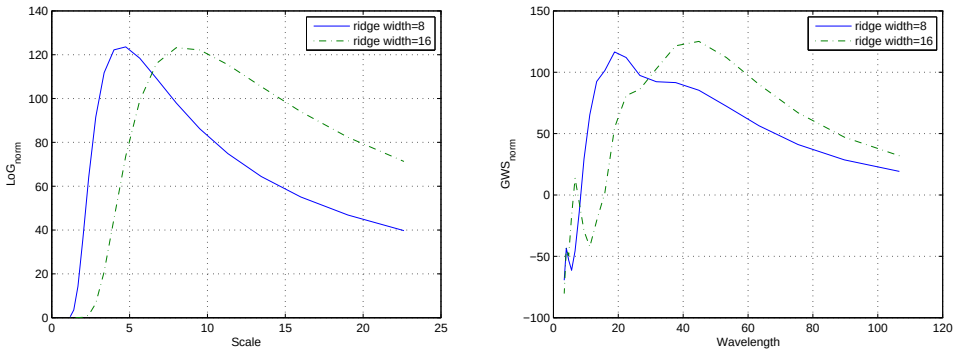


Figure 2.11: LoG_{norm} (left side) and S_w^{norm} (right side) for the ridge images in Figure 2.9.

2.6 Top-down saliency model with the λ-signature

So far we have proposed a rotation and scale invariant signature computed at object component point (x_c, y_c). In order to estimate the top-down saliency model, we can use a single example of the scale invariant signature $\tilde{\Lambda}S$, or if training examples are available, a statistical description of the data set can be computed (i.e. the mean). The top-down saliency model of component c is:

$$SM_c(\tilde{\lambda}_k) = \begin{cases} \overline{\tilde{\Lambda}S}_c(\tilde{\lambda}_k), \tilde{\lambda}_k \in \tilde{\Lambda}, & \text{mean signature} \\ \tilde{\Lambda}S(\tilde{\lambda}_k), \tilde{\lambda}_k \in \tilde{\Lambda} & \text{single sample signature,} \end{cases} \tag{2.20}$$

where $\overline{\tilde{\Lambda}S}_c$ denotes the mean value of $\tilde{\Lambda}S_{x_n^c, y_n^c}$ at locations of the object component c in the training set, $\{(x_1^c, y_1^c), \ldots, (x_n^c, y_n^c), \ldots, (x_N^c, y_N^c)\}$. After having learnt an object component model SM_c, we can analyze novel images and select only those interest points with $\tilde{\lambda}$-signatures conforming to the model. The rejection of bad candidates is performed by matching the $\tilde{\lambda}$-signature of the interest point $\tilde{\Lambda}S_{x_c,y_c}$ with the saliency model SM_c, computing the euclidean distance between signatures. We reject $\tilde{\Lambda}S_{x_c,y_c}$ if the euclidean distance is greater than the threshold learnt in the training set.

The top-down saliency model SM_c defined in Equation (2.20) computes a wavelength profile that captures the texture information of object component c. The steps to obtain the invariant wavelength profile are as follows:

- Computation of the amplitude normalized signature, S_w^{norm}.
- Computation of the object component intrinsic scale, σ_{int}.
- Computation of $\tilde{\Lambda}S_{x_c,y_c}$ by mapping λ to $\tilde{\lambda}$ values.

The model can now be used to filter out interest points very unlikely to be object component c in novel images. To assess the properties of the saliency model SM_c, we present some tests in the context of facial feature detection.

2.7 Tests

In the first test we assess the properties of the intrinsic scale computation using the λ-signature. In a second group of tests we perform interest point selection applied to face components using the scale normalized λ-signature.

2.7.1 Variance of intrinsic scale

We present results that illustrate the low variance of the intrinsic scale computation using the λ-signature, when compared to LoG_{norm}. The test comprises intrinsic scale computation of facial landmarks in AR face database [77], using ground truth points provided by [16]. We select 82 subjects without glasses and compute mean and variance of the intrinsic scale at several facial landmarks. In Figure 2.12 we observe the facial landmarks selected to compute intrinsic scale.

Figure 2.12: Facial landmarks

Table 2.3 shows the results of mean and variance of the intrinsic scale for the S_w^{norm}, LoG_{norm} and the eyes ground truth. Considering only the eyes, we remark the very similar values of the variances between the ground truth and the intrinsic scale from S_w^{norm}. The

	S_w^{norm}		LoG_{norm}		$\frac{\sigma^2_{LoG_{norm}}}{\sigma^2_{S_w^{norm}}}$	Ground truth	
Facial Point	μ	σ^2	μ	σ^2		μ	σ^2
Left Eye center	6.1	0.4	4.4	1.5	3.75	3.81	0.33
Right Eye center	5.9	0.4	4.4	1.8	4.5	4.15	0.47
Left Eye corner	5	0.7	3.5	3.8	5.43	-	-
Right Eye corner	4.4	0.8	6.7	5.8	7.25	-	-
Nose	4.8	0.3	3.8	0.3	1	-	-
Left Nostril	4.9	1	4.9	7.6	7.6	-	-
Right Nostril	5	0.2	4.1	7	35	-	-
Left Mouth corner	7.7	35.3	5.8	18.1	0.51	-	-
Right Mouth corner	6.1	36	5.3	11.8	0.33	-	-

Table 2.3: Mean μ, variance σ^2 of intrinsic scale, and variance ratio between intrinsic scales. The last two columns shows the Mean μ, variance σ^2 of the ground truth obtained from the pupils radii, computed from user-clicked points. For the rest of facial landmarks but the eyes, is very difficult to define the spatial extent to have an adequate ground truth measurement.

S_w^{norm} intrinsic scale variance is in general lower than LoG_{norm} intrinsic scale variance for the eyes and nose facial landmarks, while in the case of mouth landmarks LoG_{norm} intrinsic scale has a lower variance. Nevertheless, in the case of mouth landmarks, the variance of the intrinsic scale is very large in both cases because mouth landmarks have a greater variability (e.g. thin and thick lips, beard presence/absence), which questions the use of such landmarks for facial analysis.

In order to measure quantitatively the variance relation between S_w^{norm} intrinsic scale and LoG_{norm} intrinsic scale, we compute the variance ratio

$$\frac{\sigma^2_{LoG_{norm}}}{\sigma^2_{S_w^{norm}}}.$$

In most of the cases the ratio is greater than 1, meaning that the intrinsic scale from S_w^{norm} has smaller variance than the intrinsic scale from LoG_{norm}. We must remark that the intrinsic scale from S_w^{norm} has smaller variance even in the case of blob-like facial landmarks like eyes and nostrils. The reason for this behavior is that in the real images the eyes have two blobs: one caused by the reflection of light on the pupil and the pupil blob itself. The intrinsic scale from S_w^{norm} is less sensitive to the presence of two blobs in the eye's interest point. In the case of nostrils, their shape is elliptical instead of circular and the variation of nostril size across subjects lead to intrinsic scale errors. Thus, in real images the intrinsic scale from S_w^{norm} outperforms the intrinsic scale from LoG_{norm}. Additionally, in the specific application of intrinsic scale to normalize the λ-signature, low variance values help to compute a model

with higher precision.

2.7.2 Interest point selection of facial components

The goal of the tests in this section is to assess experimentally the most important properties of the top-down saliency model of Equation (2.20): (i) removal of points very different from the model, having very few rejections of model points, and (ii) scale invariance of the $\tilde{\lambda}$-signature. The data uses 82 subjects from the AR face database[77], where half of them are used for learning the saliency model and the remaining half is used for testing our methods. The saliency model is learnt in a supervised manner, using ground truth component points in order to compute the model SM_c.

We select candidates for facial components from a set of interest points. Four facial components are modeled by the top-down saliency model SM_c of Equation (2.20): eye, nose, nostril, and mouth corner. To estimate the saliency model SM_c, we use ground truth points of the face components in the training set. The test stage is conducted as follows:

1. We define a set of scales $\sigma_1 = 1, \sigma_2 = 2, \sigma_3 = 4, \sigma_4 = 8, \sigma_5 = 16$

2. For a given scale σ_i, the set of interest points IP_i is provided by local maxima of the amplitude of the Laplacian of Gaussian response applied at σ_i,

$$IP_i = \arg\max_{(x,y)} |I * LoG_w(x, y, \sigma_i)|, \tag{2.21}$$

   where LoG_w is the Laplacian of Gaussian kernel presented in Equation (2.18).

3. The entire set of bottom-up interest points results from the union of the points detected at all scales:

$$IP = \bigcup_i IP_i \tag{2.22}$$

4. Every point in the set IP is matched against the facial component saliency model SM_c. The matching procedure rejects interest point locations with low similarity.

To evaluate the performance of the method, we compute the recall rate of facial component detection,

$$recall = \frac{\#correct\ matches}{\#positive\ examples}. \tag{2.23}$$

We consider a match as correct if at least one of the interest points selected by the saliency model SM_c is located in the proximity of the ground truth facial component location. Proximity is defined as a circular region around the component's interest point of radius 5 pixels, so points inside the circle are marked as correct matches.

In order to evaluate the impact in computational complexity reduction, we compute the removal rate of interest points not conforming with the model,

$$removal = 1 - \frac{\#points\ selected\ by\ SM_c}{|IP|}. \tag{2.24}$$

Facial Component	Recall	removal (%)
Eye	100	79.64
Nose	100	80.03
Nostril	100	77.94
Mouth corner	90.2	89.9

Table 2.4: Results of top-down guiding search of facial landmarks

Table 2.4 shows that excluding the mouth corner point, we obtain perfect matching performance, while removing interest points that are very different from the facial component we are looking for. In average we remove 79.21% of the initial Laplacian of Gaussian salient points. The few missing points by the model are the corner mouth interest points. The reason of this behaviour is the high visual variability of corner mouth components, leading to a very unstable saliency model. As shown in Section 2.7.1, the mouth corner point is an unstable landmark and should not be used as an object component.

These tests show experimentally how the proposed method succeeds in selecting object components in a top-down manner. In the following set of tests, we check the scale invariance of the saliency model.

Scale invariance of saliency model

Using the saliency model learnt in the previous section, we now analyze the performance of the method in selecting candidates for facial components in scaled versions of the images. We compute the mean recall rate for three facial components: eye, nose, and nostril. In Table 2.5 we observe that the saliency model learnt at a fixed scale is highly tolerant to scale changes up to ± 0.5 octaves. The method is not fully invariant because of the very small size of the nostrils in the lowest resolution images. However, the results in Table 2.5 shows the suitability of the intrinsic scale method to provide scale invariance to the λ-signature.

scale change(octaves)	Performance(%)
-0.5	93.49
-0.25	100
0	100
0.25	100
0.5	100

Table 2.5: Results of scale invariance test of the saliency model

2.8 Discussion

In this chapter we have exploited Gabor filter parameters to represent texture. We characterize an object component by its dominant texture, encoded in a new top-down saliency operator, the $\lambda-$signature. We showed suitability of the signature for modeling object components, specifically in: (i) building a coarse appearance model of components, and (ii) computing the intrinsic scale of the components.

The signature is utilized for selecting interest points conforming to a particular object component and relies on a scale and rotation invariant wavelength signature. The rationale behind the signature is to gather texture information in the object components, in order to build a coarse appearance model suitable for interest point selection. The proposed appearance based saliency function is characterized by the following properties:

- Successfully removes points that are very different from the object component.
- Has very few rejections of true positives.
- Invariance to position, orientation and scale of the object component being searched.

These properties are adequate to include the saliency model proposed during the early stages of the object recognition process in order to reduce the number of interest point candidates for every component, decreasing significantly the number of computations during object matching process.

As a second application of the wavelength saliency operator, we describe a method to compute the intrinsic scale of an interest point. The method proposed is able to compute a very good approximation of the scaling factor between scaled image regions, having a similar behavior to the scale-normalized Laplacian of Gaussian (LoG). The intrinsic scale σ_{int} from the λ-signature is characterized by the following properties:

- Higher versatility than the intrinsic scale $\sigma_{\text{int}}^{LoG}$ from $|LoG_{norm}|$, supported by the correct behavior of σ_{int} in both blob and ridge features.

- Smaller variance than $\sigma_{\text{int}}^{LoG}$ in the case of facial components of similar size.

In summary, we have shown how to apply the properties of Gabor filters in the first step of the component-based object recognition: interest point selection. In the next chapters, we will explore Gabor filter properties in the second step of object recognition: local descriptor computation.

Chapter 3

Filter-based descriptors

So far we have explored the use of the wavelength (inverse of frequency) parameter of the Gabor function to select interest points in a top-down manner, preselecting candidates for object components from a set of interest points. In this chapter we show how to represent object components in a much richer way, in order to perform component matching in a robust manner under common image luminance and geometric deformations. Object components will be represented by local descriptors. The local descriptors proposed in the literature can be divided in two main classes: filter-based and histogram-based. Filter-based descriptors rely on collecting and processing the response of linear filters in the interest point neighborhood, like Gaussian derivatives [58], Gabor filters [38, 21], and steerable filters [36]. Histogram-based descriptors [4, 70, 3] instead compute the statistical distribution of the image gradient in image patches around the interest point.

In this chapter we focus on filter-based descriptors, while histogram-based descriptors will be addressed in Chapter 4. The motivation behind the usage of filter-based descriptors is two-fold:

- Biological findings of neuron responses in low-level visual cortex areas [46, 20] show that the neurons' response pattern can be characterized as filter responses (i.e. receptive field responses).
- Linear filters have been studied extensively in signal processing domain, and formal methods are available to tune their parameters in order to capture particular properties of the image region under analysis.

From this class of descriptors, the ones used in [68, 109] in the HMAX architecture have shown very good results in recognition performance. They are based on a very dense sampling of the filter-bank parameters (orientations and scales). Furthermore, they collect the filter-bank responses at all points in the image patch of interest. Usually these descriptors have a

very high dimension, requiring large storage capabilities and long computation times in the matching procedure. We denote this class of representation as *dense descriptors*.

Despite the descriptors of the HMAX architecture being state-of-the-art, their computation time is too long for the real-time demands of some applications. In this chapter we will look at faster alternatives, tolerating a possible decrease in performance, but allowing decisions to be taken in short time, depending on the application requirements. Several works have adopted such a parsimonious representation paradigm to build local image descriptors [113, 107, 47, 70] and they have shown good recognition rates in several applications. These descriptors have a smaller dimension because they store the response of the filter-bank only at the interest point and have a sparser sampling of the parameter space. Given their smaller dimension, the matching process is very fast. This class of representation we denote as *sparse descriptors*.

In this chapter we adopt a sparse descriptor approach to object component representation. Contrary to previous approaches, where the filter-bank parameters are fixed regardless of the object component to be described, we propose an adaptive computation of the filter-bank parameters depending on the particular image information. We use the Gabor filter response to select the most adequate parameters for every object component. The selected parameters are then used to compute the feature vector. For the same vector dimension, an adaptive filter descriptor with appropriately selected parameters will, in general, overcome fixed parameter descriptors in recognition performance. It will eventually approach the performance of dense representations, but with lower computational cost.

In the previous chapter we have shown that wavelength is an important Gabor parameter for object component preselection. Likewise, we include all Gabor filter parameters (scale, orientation, and wavelength) in the adaptive descriptor. To perform the selection, we look for local extrema in the parameter space, of the response of Gabor filters at the interest point location [88].

Usually, approaches that use Gabor filters as local descriptors are not fully invariant to image transformations. To evaluate the robustness of the descriptor we analyze how it changes under image rotations and scalings. Then, we introduce methods to achieve rotation and scale invariance of the adaptive Gabor bank for object component modeling [89]. In summary, we introduce a method that explores the richness of Gabor filter parameters by selecting the filter parameters that best represent each object component, while being invariant to rigid transformations.

3.1 Related work

Here we review the most important filter-based descriptors that have been applied to represent image patches. One of the most paradigmatic works on filter based descriptors [58] uses Gaussian derivative filters. Gaussian derivatives of order n capture the local structure around an image pixel and can be viewed as the n order coefficients of the Taylor series expansion of the image at that pixel. The parameters of Gaussian derivative filters are the derivative order and scale. In order to build Gaussian derivative filter-bank descriptors, common approaches compute responses for selected derivative orders and scales. Then, responses are stacked in a feature vector (i.e. local descriptor). Huttenlocher and Felzenszwalb [47] represent the appearance of object components by convolving the object part with a set of Gaussian derivatives. Schiele and Crowley [106] compute histograms of scale-normalized Gaussian derivative responses of object components. In the case of color images, Geusebroek et.al. [40] introduced a Gaussian color model, utilized by Hall et.al. [42] to compute Gaussian derivatives of color receptive fields. Hall et. al. used the color differential structure provided by Gaussian derivatives to compute local descriptors in color images. Local descriptors based on Gaussian derivatives are able to represent object components, but in their standard form, they are not invariant to common image transformations like rotations and scalings.

Koenderink and Van Doorn introduced the differential invariants [57], combinations of Gaussian derivatives of different orders invariant to 2-dimensional rigid transformations. Schmid and Mohr [107] use these differential invariant responses to compute local descriptors (i.e. "local jets").

Freeman and Adelson proposed another approach to achieve orientation invariance, consisting of the use of steerable filters [36] to form a set of "basis functions." This procedure allows to "steer" a filter to any orientation. Rao and Ballard [102] apply a bank of steerable filters using Gaussian derivatives as "basis functions" in order to build descriptors of object components.

Gabor filters have a richer set of parameters than Gaussian derivatives and have been used in some works as local descriptors [60, 122, 113]. Lades et.al. [60] and Wiskott et.al [122] use Gabor filter response to describe regions around nodes of a graph, computing a "Gabor jet" (i.e. filter bank). Smeraldi and Bigun [113] design a bank of Gabor filters, whose energy is spread in the frequency domain in a log-polar configuration. The design criterion guarantees that relevant frequencies are captured by the filter bank. Then, the filter bank is utilized to represent facial landmarks.

The biologically inspired HMAX features [103] also use Gabor filter bank responses to compute local descriptors, but belong to the type of dense representations. The initial step

is the computation of Gabor filter responses using a fixed set of scales and orientations, whose parameters are tuned similarly to V1 simple cells in the visual cortex of monkeys [20]. Then, local maxima over position and scale of the Gabor responses generate the so called C1 features, a dense and redundant representation of the local image appearance. The C1 representation has recently been used in object detection in cluttered images [109].

Our proposal belongs to the sparse type of Gabor filter-based representations. One of the most recent works in object recognition using Gabor filters [59] proposed a sparse Gabor feature representation, being invariant to scale, rotation, translation, and illumination image transformations. This work introduces a feature matrix, where the (i, j) component is the Gabor filter response at fixed scale σ_i and orientation θ_j. The rationale behind the matrix is to cope with image rotations and scalings. If the matrix is computed from objects in a standard pose, it is possible to introduce column-wise or/and row-wise shifts to match the object. In order to handle linear illumination changes, the matrix is normalized by the summation of all its components. An application of this idea is presented in [43], in a face detection task using high resolution images.

In this thesis we go a step further by exploring all Gabor function parameters to represent an object component. The adaptation of feature parameters to particular object parts was first exploited in [53]. They propose to select the Gabor function scale and orientation in a semi-automatic fashion, using the "Information Diagram" concept. The Information Diagram represents the Gabor filter response at an image point, as a function of the filter orientation and scale. We extend the Information Diagram concept to consider all Gabor filter parameters (scale, orientation, and wavelength), thus resulting in a 3-dimensional function.

3.2 Dense vs. sparse Gabor filter-based descriptors

In this section we concentrate on descriptor models using Gabor filters. We will start with a brief description the HMAX model [109]. Given its reported state-of-the-art performance, in Chapter 5 this model will be considered as a benchmark for comparison purposes.

3.2.1 The HMAX descriptor

The biologically inspired HMAX model was firstly proposed by Riesenhuber and Poggio [103] and recently revised by Serre et al. [109]. The HMAX architecture considers all phases of an object recognition architecture, including feature extraction, local image description, descriptor matching and object model learning and classification. We start by describing the way to represent an object component. Later, in Chapter 5, we will focus on object model

learning and classification.

In Figure 3.1 we can see a graphical description of the HMAX feature extraction steps (from [109]). The feature extraction, descriptor definition, and matching steps are described in the following algorithm:

1. S1 maps: First, images are analyzed with a Gabor filter bank. The parameters of the filter bank are tuned for several scales (σ) and orientations (θ), similarly to V1 simple cells in the visual cortex of monkeys [20]. The maps created are denoted by $\mathbf{S1}(x, y; \sigma, \theta)$, with x, y the image spatial coordinates.

2. C1 maps: Pairs of scale adjacent S1 maps are subsampled and combined into *bands*, by computing the local maximum across scale. Thus, a pixel of the C1 map has the strongest response between pixels at the same location in two adjacent scales. This process is done for each orientation and pair of adjacent scales independently. These maps are represented by $\mathbf{C1}(x, y; b, \theta)$, where b is the *band* index. Figure 3.2 shows the C1 maps of all orientations in the first band.

The two steps just explained (S1 and C1 maps) compute, for each pixel (x_i, y_i) in the object component, a vector containing the local maximum in adjacent scales of the Gabor filters responses for the different orientations. Therefore, they provide a dense Gabor filter-based representation as the component model that we refer to as $\mathbf{u}(b, \theta)$. In order to match the descriptor in new images, we do as follows:

1. Compute the C1 map of the new image, $\mathbf{X}(x, y; b, \theta)$.

2. S2 maps: Compute the exponential mapping of the Euclidean distance between the descriptor centered at all image points $\mathbf{X}(x, y; b, \theta)$ and $\mathbf{u}(b, \theta)$.

$$\mathbf{S2}(x, y; b, \theta) = \exp(-\gamma \|\mathbf{X}(x, y; b, \theta) - \mathbf{u}(b, \theta)\|^2), \tag{3.1}$$

   where γ is a tunable parameter.

3. C2 features: compute the maximum over all positions, bands and orientations at the S2 map, obtaining a single value C2 for the object component

$$\mathrm{C2} = \max_{x,y,b,\theta} \mathbf{S2}(x, y; b, \theta) \tag{3.2}$$

Thus, a C2 value describes the strength of the most similar point in one image with respect to the descriptor $\mathbf{u}$ of a particular object component. This value can be used to train a binary classifier with positive and negative examples of the component.

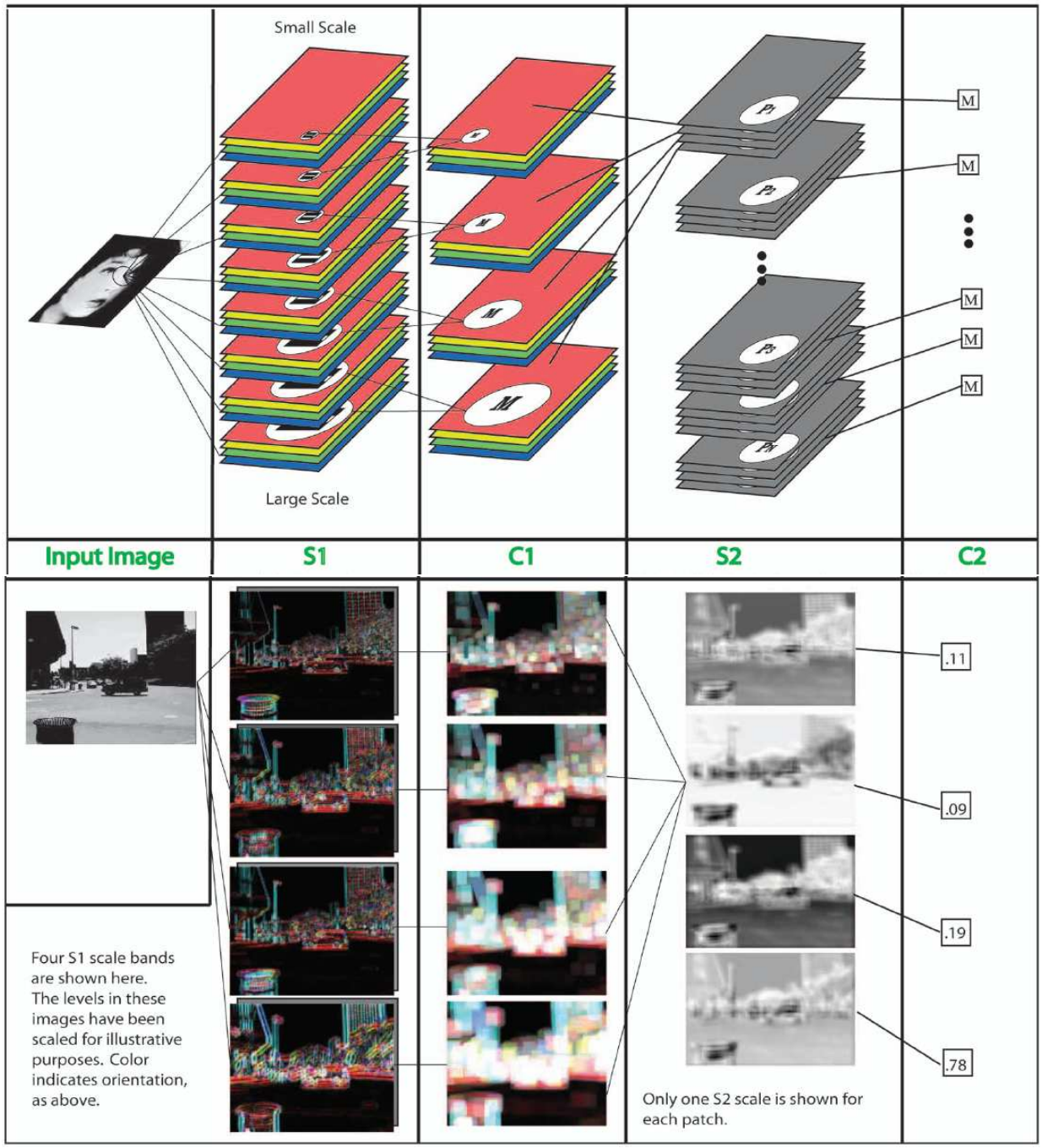
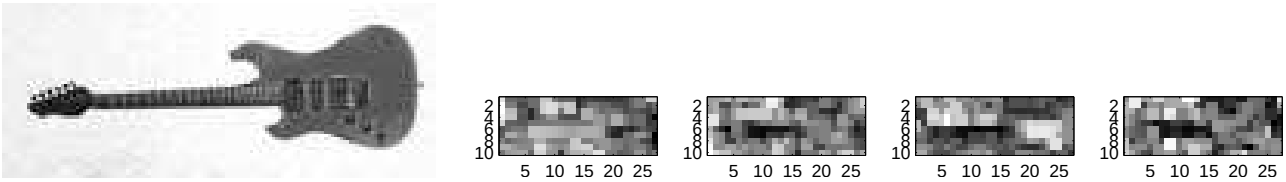


Figure 3.1: Overview of HMAX feature extraction, extracted from [110]



(a) (b)

Figure 3.2: Sample C1-HMAX representation. (a) Original image. (b) C1-HMAX representation from the first band (4 orientations).

In general the HMAX descriptor and matching procedures are computationally heavy. In the next section we address more compact approaches to build local descriptors using Gabor filter responses.

3.2.2 Sparse Gabor filter-based component models

Sparse representations compute a feature vector that collects the Gabor filter responses at the interest point of the object component, using a filter bank with parameters $\{(\theta_1, f_1, \sigma_1), \ldots, (\theta_m, f_m, \sigma_m), \ldots, (\theta_M, f_M, \sigma_M)\}$. The most common way to build the feature vector is to stack the amplitude of the responses at point (x, y) as follows:

$$\mathbf{u}(x,y) = \Big(u^1(x,y), \quad \cdots \quad, u^m(x,y), \quad \cdots \quad, u^M(x,y)\Big)^T, \tag{3.3}$$

with

$$u^m(x,y) = |(g_{\theta_m, f_m, \sigma_m} * I)(x,y)|, \tag{3.4}$$

where $(\theta_m, f_m, \sigma_m)$ are the Gabor parameters used to compute the response at u^m, for a filter bank of size M. As an alternative the feature vector can contain both the real and the imaginary part of the Gabor responses:

$$\mathbf{u}(x,y) = \Big(u^1(x,y), \quad \cdots \quad, u^m(x,y), \quad \cdots \quad, u^{2M}(x,y)\Big)^T, \tag{3.5}$$

with

$$u^{2m}(x,y) = \mathrm{Re}((g_{\theta_m, f_m, \sigma_m} * I)(x,y)), \quad u^{2m-1}(x,y) = \mathrm{Im}((g_{\theta_m, f_m, \sigma_m} . * I)(x,y)).$$

The rationale is to model object components by analyzing their contents in terms of edges and textures of different scales, orientations, and frequencies. Huttenlocher and Felzenszwalb

[47] showed experimentally the adequacy of the Gaussian model of the filter bank in the case of Gaussian derivatives. We adopt this Gaussian assumption, modeling the local descriptor as a random feature vector that follows a normal distribution with mean $\mu_{\mathbf{u}_c}$ and covariance matrix Σ_c, $\mathbf{u}(x,y) \sim \mathcal{N}(\mu_{\mathbf{u}_c}, \Sigma_c)$. In difference to the Gaussian derivatives, the response of a Gabor filter is a complex number and its response can be considered as a 2D random vector that follows a normal distribution with a two dimensional covariance matrix Σ, which is included in the high dimensional matrix Σ_c. The Gaussian assumption allow us to evaluate in a straightforward classification test the ability of the descriptor to discriminate correctly object components.

In order to match the object component model in new images, we will compute the distance between the model learnt and the novel patterns. We consider both the Euclidean and Mahalanobis distances,

$$\begin{aligned} \text{Euclidean } d^2 &= (\mathbf{u}(x,y) - \mu_{\mathbf{u}_c})^T (\mathbf{u}(x,y) - \mu_{\mathbf{u}_c}) \\ \text{Mahalanobis } d^2 &= (\mathbf{u}(x,y) - \mu_{\mathbf{u}_c})^T \Sigma_c^{-1} (\mathbf{u}(x,y) - \mu_{\mathbf{u}_c}). \end{aligned} \tag{3.6}$$

The decision of whether a object component is present or not in a certain image pixel will depend on the distance values computed.

3.3 Adaptive filter-based descriptors

The dense and sparse models introduced in the previous section have different characteristics. Regarding the computational complexity of the matching procedure, the dense models need very large times for matching. To illustrate this fact, let us consider an object component with size $M \times N$ and descriptor size S. The matching complexity of the dense descriptor HMAX (with the addition of bands B and orientations T) is $O(M \times N \times B \times T \times S)$. On the other hand, the matching complexity of the sparse models depends linearly on the feature vector size, $O(S)$, and those models have shown good recognition rates in several applications [113, 122, 5, 47]. Thus, we consider sparse component models with high efficiency characteristics in the remaining of this chapter. The sparse model of each component presented in this section consists of a vector of Gabor filter responses. However, instead of using predefined values for the Gabor filter parameters we propose methods for the selection of these parameters, exploiting the specific properties of each object component. These descriptors are *adaptive* to the local image information, which will lead to better performance than fixed parameter descriptors.

3.3.1 Parameter selection

We address the selection of the Gabor filter bank with parameters $\{(\theta_1, f_1, \sigma_1), \ldots, (\theta_M, f_M, \sigma_M)\}$ that best models an object component. We recall the 2D zero mean isotropic Gabor function, written as:

$$g_{\theta,f,\sigma}(x,y) = \frac{e^{-\frac{x^2+y^2}{2\sigma^2}}}{2\pi\sigma^2}\left(e^{j2\pi f(x\cos(\theta)+y\sin(\theta))} - e^{-2\sigma^2 f^2\pi^2}\right), \tag{3.7}$$

where the parameters f, θ, and σ are the frequency, orientation, and width of the Gabor function. A straightforward approach to define the filter bank parameters would be to sample the parameter space uniformly in some limited range. However, this may not be the best strategy to exploit the particular characteristics of the object component under test since the choice of the parameters would not be driven by the object specific appearance. Alternatively, we can analyze the Gabor response function in the full parameter space (σ, f, and θ) and select those parameters that best describe the particular object component characteristics,

$$(\hat{\sigma}_c, \hat{f}_c, \hat{\theta}_c) = \arg\max_{\sigma,f,\theta} |(g_{\theta,f,\sigma}(x,y) * I)(x_c, y_c)|, \tag{3.8}$$

with

$$(g_{\theta,f,\sigma} * I)(x_c, y_c) = \int\int I(x,y) g_{\theta,f,\sigma}(x_c - x, y_c - y) dxdy,$$

where (x_c, y_c) denotes location of object component c in image I. However, the sampling strategy of Equation (3.8) would select a single Gabor function that is insufficient to discriminate the modeled object component from others. Even if we select the first M local maxima of the Gabor filter response magnitude, this strategy could bias the parameter distribution to a too narrow range and reduce the discrimination capability of the filter bank. In order to maintain a uniform parameter range and still be able to adapt the representation to the particular object component under test, we will sample one of the parameters uniformly and perform a 2D search of local extrema in the remaining dimensions. We explore the three different options, sampling uniformly θ, f, and σ.

We have several ways of coding the object component appearance, as we can choose:

1. Which parameter we sample uniformly (θ, f, σ),
2. The type of local extrema used to select the remaining parameters (e.g. only minima, only maxima, and minima and maxima),
3. The metric to match object component model (e.g. Euclidean distance, Mahalanobis distance), and
4. The response type (e.g. modulus, real + imaginary parts).

In order to choose the most adequate option for every item, we evaluate the performance of the different alternatives in a facial component detection experiment. Then, we check the robustness of the chosen local descriptor to image rotations and scalings.

Extended information diagram

The "Information Diagram" (ID) concept proposed in [53] selects the Gabor filter parameters semi-automatically. The ID represents the magnitude of the Gabor response at a certain interest point of an object component (x_c, y_c), as a function of θ and σ, keeping the value of $1/\sigma f$ constant. The ID function is defined as:

$$\mathrm{ID}_c(\theta, \sigma) = |(g_{\theta, \frac{1}{\sigma f}=1, \sigma} * I)(x_c, y_c)|.$$

Then, the parameters (θ, σ) corresponding to local maxima of ID are chosen as "good" Gabor function parameters because they represent the object component's characteristic orientations and widths. We extend the ID concept to consider variability also in the $1/\sigma f$ value, by sampling f values independently of σ values. Considering the parameters (θ, f, σ), the Extended Information Diagram is given by:

$$\mathrm{EID}_c(\theta, f, \sigma) = |(g_{\theta, f, \sigma} * I)(x_c, y_c)|. \tag{3.9}$$

EID_c is the parameter space function of Gabor filter responses at the interest point of the object component c. We analyze the EID function to select the adequate filter parameters.

Parameter selection in the Extended Information Diagram

Considering the three dimensional parameter set, the strategy to find the adequate parameters consists of "slicing" the parameter space and then searching for local extrema in the 2D slices. We observe in Figure 3.3 the different forms of slicing the EID function (Equation 3.9): θ slices, σ slices, and f slices. The strategy to find good parameters for each target is based on uniform discretization of one of the parameters (say θ) of Equation (3.9) and search for local maxima in the resulting set of EID slices. We denote the slices: θ-ID, σ-ID, and f-ID respectively, keeping constant one of the parameters, $\theta = \theta_0$, $\sigma = \sigma_0$ or $f = f_0$:

$$\begin{aligned} \theta\text{-ID}_c^{\theta_0}(\sigma, f) &= \mathrm{EID}_c(\theta_0, f, \sigma) \\ \sigma\text{-ID}_c^{\sigma_0}(\theta, f) &= \mathrm{EID}_c(\theta, f, \sigma_0) \\ f\text{-ID}_c^{f_0}(\theta, \sigma) &= \mathrm{EID}_c(\theta, f_0, \sigma) \end{aligned} \tag{3.10}$$

Figure 3.3 shows some examples of the θ-ID, σ-ID and f-ID computed at an eye's center point.

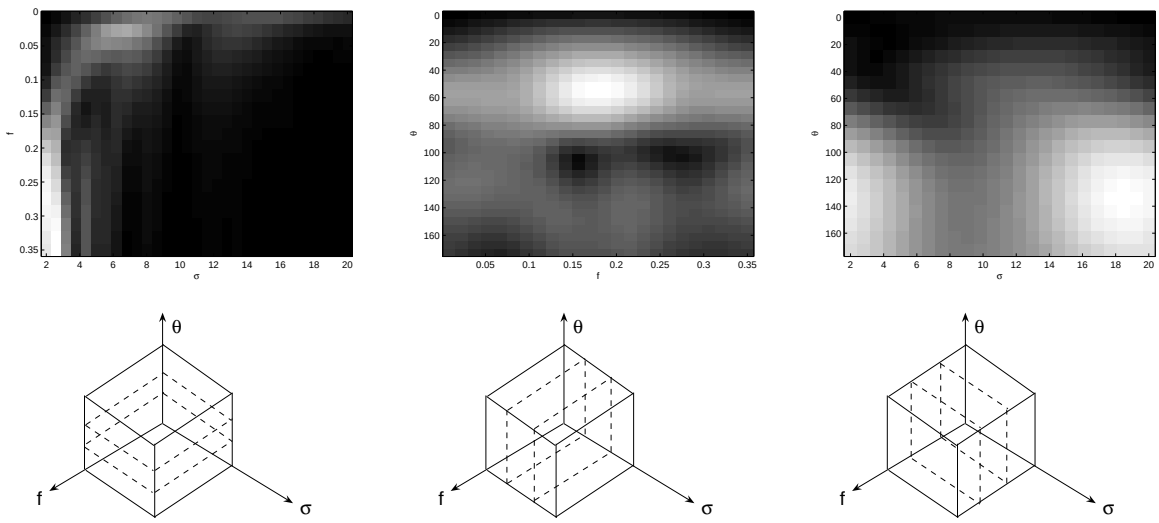


Figure 3.3: Examples of θ-ID, σ-ID, f-ID, and σ slices in the parameter space (from left to right).

Continuing the θ example, let us define a set $\mathcal{T} = \{\theta_1, \cdots, \theta_l, \cdots, \theta_L\}$, containing uniformly sampled values of θ within $[0, \pi)$. Thus, the set of θ-IDs for object component c is given by:

$$\Theta\text{-ID}_c^{\mathcal{T}} = \{\theta\text{-ID}_c^{\theta_1}, \cdots, \theta\text{-ID}_c^{\theta_l}, \cdots, \theta\text{-ID}_c^{\theta_L}\} \tag{3.11}$$

To select Gabor parameters, we assume that, in every slice, an object component has two representative textures that can be extracted with local extrema points of the slice. The initial hypothesis is to relate local maxima to significant textures in the component and local minima to textures with low weight. We do not know *a priori* which is the best way of combining the extrema points, so we consider some combinations of local minima and maxima, provided by the two highest local maxima

$$\begin{aligned}
(\hat{\sigma}_{l,1}^{\max}, \hat{f}_{l,1}^{\max}) &= \arg\max_{\sigma,f} \theta\text{-ID}_c^{\theta_l}, l = 1, \cdots, L, \\
(\hat{\sigma}_{l,2}^{\max}, \hat{f}_{l,2}^{\max}) &= \arg\max_{\sigma,f,\sigma\neq\hat{\sigma}_{l,1},f\neq\hat{f}_{l,1}} \theta\text{-ID}_c^{\theta_l}, l = 1, \cdots, L,
\end{aligned} \tag{3.12}$$

and the two smallest local minima:

$$\begin{aligned}(\hat{\sigma}_{l,1}^{\min}, \hat{f}_{l,1}^{\min}) &= \arg\min_{\sigma,f} \theta\text{-ID}_c^{\theta_l}, l = 1, \cdots, L \\ (\hat{\sigma}_{l,2}^{\min}, \hat{f}_{l,2}^{\min}) &= \arg \min_{\sigma,f,\sigma\neq\hat{\sigma}_{l,1},f\neq\hat{f}_{l,1}} \theta\text{-ID}_c^{\theta_l}.l = 1, \cdots, L, \end{aligned} \tag{3.13}$$

The parameter selection of Equations (3.12)-(3.13) computes the "good" filter parameters in the case of θ-ID. We can use the same strategy to select the parameters from σ-ID and f-ID, computing combinations of local maxima and minima. The appropriate sampling approach (ID slicing) and the local extrema combination will be chosen in an experimental basis, performing a facial component detection using eyes, nose, and mouth.

Now we have all the elements to build the component model of equations (3.3) and (3.5). In order to estimate the model, we carry on the following steps: (i) compute a mean component image $\bar{I}_c$, then (ii) compute the Gabor filter parameters from EID_c slices using $\bar{I}_c$, and (iii) compute the mean and covariance of the Gabor filter bank response by applying the filter bank selected from EID_c in the training set of component images.

3.3.2 Parameter selection tests

We perform facial component detection in order to select the object component model structure that achieves the best performance. Then, we evaluate the invariance properties of the model chosen, by detecting facial components in rotated and scaled images. Experiments are set-up for evaluating the discretized parameters (σ, f, or θ), the number and type of the extrema computed at each ID, the distance metrics (Euclidean and Mahalanobis), and the filter response type (magnitude *vs* real-imaginary parts). We present in the Table 3.2 the list of the degrees of freedom combined.

For each test shown in Table 3.2, we use 82 subjects from the AR face database [77], all without glasses, where half of them are used for training (compute object component model μ_c, Σ_c) and the remaining half for testing (object component detection). We represent four different facial components: left eye, right eye, nose, and mouth. We use $L = 12$ slices of EID and at each x-ID slice we choose either one local maximum and one local minimum or two local maxima, so the number of filters is kept constant ($M = 2L = 24$ in Equations (3.3) and (3.5)). The number of samples of the training data set is not large enough for estimating the full covariance matrix of the descriptor that contains the real and imaginary parts of the Gabor response. Thus, we approximate the covariance matrix by computing a diagonal matrix and consequently lose the covariance information between the real and imaginary parts of the response. The sets of values for the θ-ID, f-ID, and σ-ID are, respectively,

$\mathcal{T} = \{0, \pi/12, \cdots, 11\pi/12\}$, $\mathcal{F} = \{0.5, 0.4589, \cdots, 0.0063\}$, and $\mathcal{S} = \{4, 7, \cdots, 39\}$. All x-IDs are calculated from the mean images $\bar{I}_c$ in the training set at each object component (left eye, right eye, nose, mouth). In order to see the advantage over the common fixed Gabor filter approach, we compare against the biologically plausible fixed parameters presented in [61], using 4 orientations and 6 combinations scale-wavelength as shown in Table 3.1.

σ(rows),λ(columns)	1.7	3.7	7.4	14.8
0.95	X			
2.12		X		
4.35			X	
8.75			X	
17.5			X	
35.04				X

Table 3.1: λ and σ pairs used in the test with fixed parameters. The orientation values used are $\theta = \{0, \pi/4, \pi/2, 3\pi/4\}$

Test	ID type	# local max	# local min	distance	mag	re+im
1	θ	1	1	Mah	68.49	78.33
2	θ	2	0	Mah	85.92	95.83
3	f	2	0	Mah	58.19	74.16
4	f	1	1	Mah	54.41	75.83
5	σ	2	0	Mah	58.19	72.50
6	σ	1	1	Mah	50.21	72.50
7	θ	1	1	Euc	31.93	85
8	θ	2	0	Euc	38.87	87.5
9	f	2	0	Euc	17.86	53.33
10	f	1	1	Euc	15.55	45
11	σ	2	0	Euc	24.79	74.17
12	σ	1	1	Euc	15.97	75.83
13	fixed (Table 3.1)	-	-	Mah	75.40	78.30
14	fixed (Table 3.1)	-	-	Euc	68.51	71.33

Table 3.2: List of the performed tests to select the best target model. Recall rate in last two columns(%)

To evaluate the performance of each experiment we compute the recall rate of facial component detection,

$$recall = \frac{\#correct\ matches}{\#true\ positive\ components}. \tag{3.14}$$

The recall rate represents the number of object components detected correctly, so a feature vector with maximum recall will not miss any component. There is a correct match of an

object component in a new image if the global minima of the distance to the model is located in the proximity of the ground truth facial component location. Proximity is defined as a circular region around the component's interest point, so points inside the circle are marked as correct matches.

In Figure 3.4 we observe the average recall, marginalizing all tests for every degree of freedom of the model. The marginalized results show that the most adequate selections are:

1. θ is the parameter to sample uniformly,
2. two local maxima for each slice of EID are the type of local extrema to select filter frequency and width,
3. the Mahalanobis distance is the best metric to match object component models,
4. the union of real and imaginary parts of the Gabor filter response outperforms the use of magnitude alone, and
5. the adaptive parameter selection outperforms the fixed filter-based approaches.

In the second row of the rightmost column in Table 3.2, we see that the best recall rate is located in the test that combines the previous selections. The combination of θ-IDs, Mahalanobis distance and 2 local maxima has a success rate of 95%.

Let us summarize the results in a more formal way. The Gabor parameter selection that has the best performance is:

$$\Theta\text{-ID}_c^{\mathcal{T}} = \{\theta\text{-ID}_c^{\theta_1}, \cdots, \theta\text{-ID}_c^{\theta_l}, \cdots, \theta\text{-ID}_c^{\theta_L}\}$$

where $\theta_l \in \mathcal{T} = \{0, \pi/12, \cdots, 11\pi/12\}$. We select in each 2D slice $\theta\text{-ID}_c^{\theta_l}$ the parameters of the two strongest local maxima:

$$\begin{aligned} (\hat{\sigma}_{l,1}^{\max}, \hat{f}_{l,1}^{\max}) &= \arg\max_{\sigma,f} \theta\text{-ID}_c^{\theta_l} \\ (\hat{\sigma}_{l,2}^{\max}, \hat{f}_{l,2}^{\max}) &= \arg\max_{\sigma,f,\sigma\neq\hat{\sigma}_{l,1},f\neq\hat{f}_{l,1}} \theta\text{-ID}_c^{\theta_l} \end{aligned} \tag{3.15}$$

The chosen parameters define a Gabor filter bank of size $2L$ adapted to the object component c. The respective local descriptor is:

$$\mathbf{u}(x,y) = \Big(u^1(x,y), \quad \cdots \quad, u^{4L}(x,y)\Big)^T \tag{3.16}$$

$$u^{4l-3}(x,y) = \mathrm{Re}((g_{\theta_l,\hat{f}_{l,1}^{\max},\hat{\sigma}_{l,1}^{\max}} * I)(x,y)); \quad u^{4l-2}(x,y) = \mathrm{Im}((g_{\theta_l,\hat{f}_{l,1}^{\max},\hat{\sigma}_{l,1}^{\max}} * I)(x,y));$$

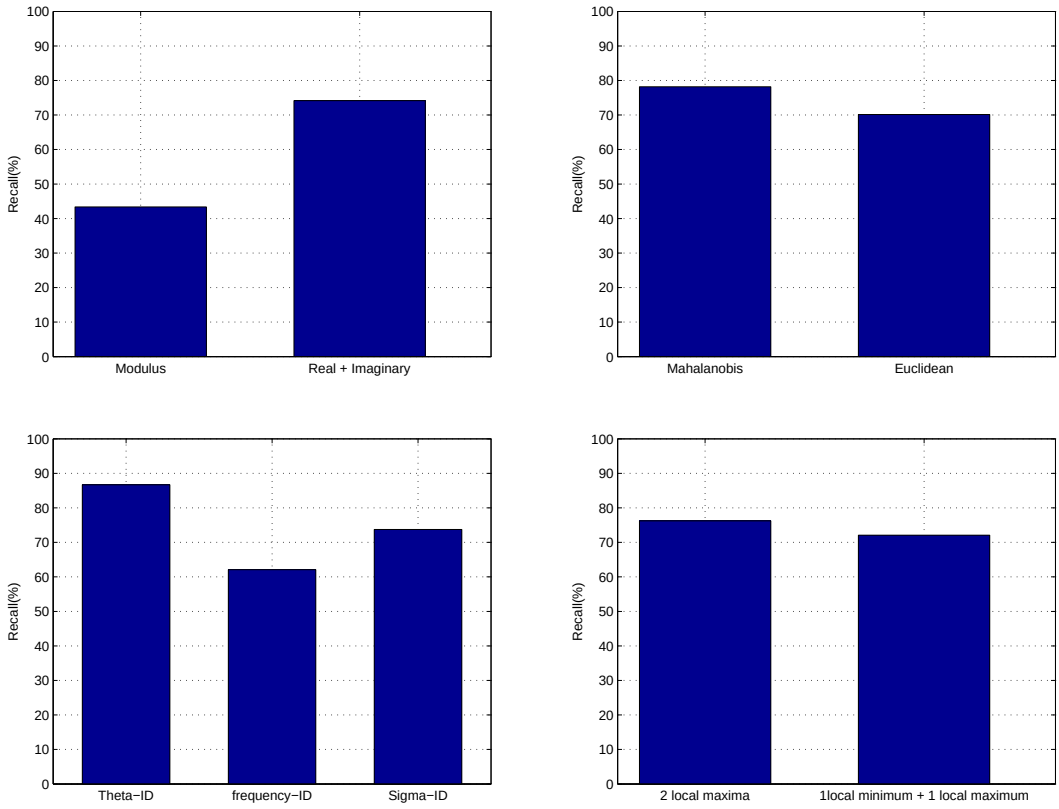


Figure 3.4: Mean detection rate of marginalized tests of Table 3.2

$$u^{4l-1}(x,y) = \mathrm{Re}((g_{\theta_l,\hat{f}_{l,2}^{\max},\hat{\sigma}_{l,2}^{\max}} * I)(x,y)); \quad u^{4l}(x,y) = \mathrm{Im}((g_{\theta_l,\hat{f}_{l,2}^{\max},\hat{\sigma}_{l,2}^{\max}} * I)(x,y)),$$

and the Mahalanobis distance for matching the target model. The object component model is the mean and covariance matrix of local descriptor in Equation (3.16). In order to add rotation and scale invariance to this model, we analyze first the robustness of the component model to image rotations and scalings.

Discretization effects on rotated filters

We test the rotation invariance of the Gabor filter response on a synthetic image and evaluate, in the face data set, the effects of Gabor response variations to rotated patterns. Due to discretization effects and imperfect filter symmetry, Gabor response presents small variations with the amount of rotation. To illustrate this fact, we (i) compute the response of a Gabor filter at center point of a synthetic edge image, then (ii) compute the response of a α-rotated Gabor filter at the center point of the α-rotated edge image, and (iii) compute the response differences in magnitude and phase, considering the initial image as reference. Figure 3.5

shows the magnitude and phase errors for several α values. We can observe that there are some errors in the magnitude and phase that, though not dramatic, can change the performance of the detection algorithm.

We perform a similar test in the facial detection problem, to see the variation of the success rate of the component model when using rotated images. We pick the component model of Equation (3.16) as the reference, then we generate α-rotated models. To generate the α-rotated component models we shift the angles in Equation (3.16),

$$\mathbf{u}(x,y) = \Big(u^1(x,y), \quad \cdots \quad , u^l(x,y), \quad \cdots \quad , u^{4L}(x,y)\Big)^T \tag{3.17}$$

$$u^{4l-3}(x,y) = \mathrm{Re}((g_{\theta_l+\alpha,\hat{f}^{\max}_{l,1},\hat{\sigma}^{\max}_{l,1}} * I)(x,y)); \quad u^{4l-2}(x,y) = \mathrm{Im}((g_{\theta_l+\alpha,\hat{f}^{\max}_{l,1},\hat{\sigma}^{\max}_{l,1}} * I)(x,y));$$

$$u^{4l-1}(x,y) = \mathrm{Re}((g_{\theta_l+\alpha,\hat{f}^{\max}_{l,2},\hat{\sigma}^{\max}_{l,2}} * I)(x,y)); \quad u^{4l}(x,y) = \mathrm{Im}((g_{\theta_l+\alpha,\hat{f}^{\max}_{l,2},\hat{\sigma}^{\max}_{l,2}} * I)(x,y)).$$

We compute the recall rate of the α-rotated model of Equation (3.17) in α-rotated images. We see the variation of recall rate for several α values in Figure 3.6, when rotating both the test images and the model. For simplicity, in this test we rotate the image regions every $\pi/4$, because it does not involve a recomputation of the target model, only a correct circular shift of the vector is needed. We observe a very good behavior of the rotated model in the rotated images, with recall above 91% (in the non-rotated test the performance is 95.8%). The implication of this result is important because we can add rotation invariance straightforwardly to the object component model, a method that will be explained in Section 3.4.2.

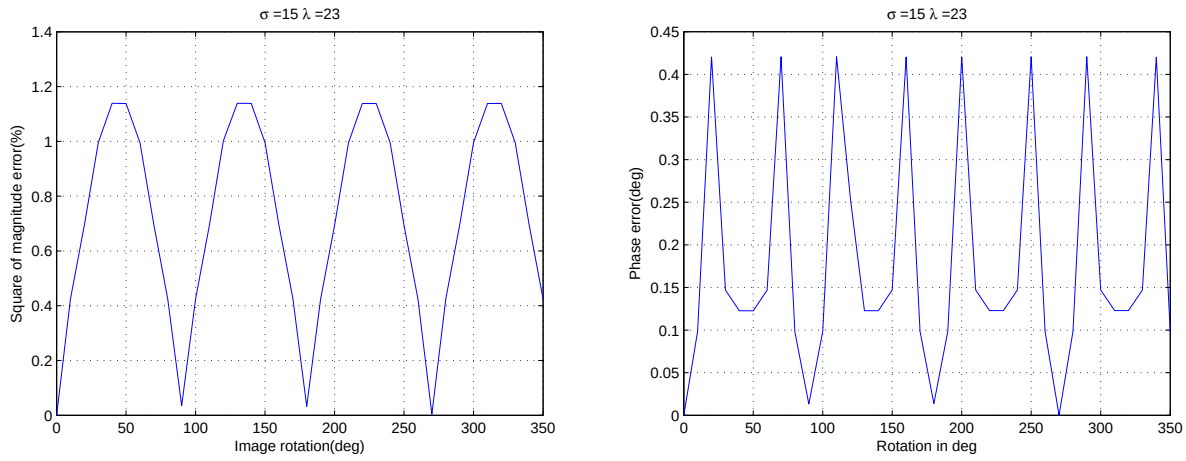


Figure 3.5: Gabor filter rotation robustness tests in synthetic images.

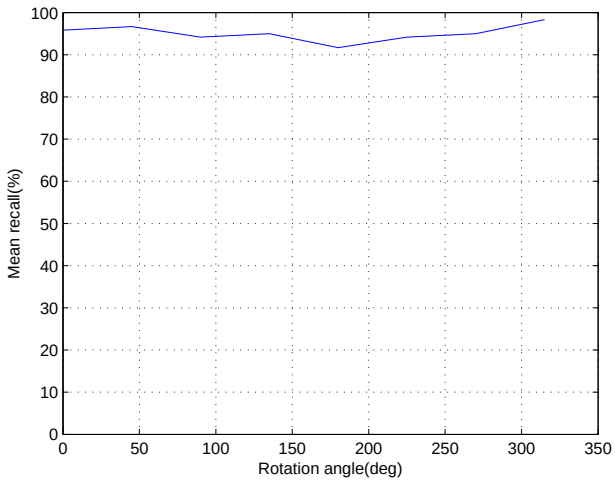


Figure 3.6: Rotation robustness of the component descriptor in rotated images.

Scale robustness

To check the robustness to scale variations, we compute the recall rate in rescaled images maintaining the object component model learned in the original images. Figure 3.7 shows a performance above 90% for image rescaling up to ±20%, corresponding to a range of about 0.6 octaves. To cope with larger scale variations, one should cover the scale dimension with additional object component models. If we sample the scale space every 0.6 octaves, we should be able to keep performance above 90%, provided that an adequate scale selection method is available, like the intrinsic scale from λ-signature presented in Chapter 2. In the next section we go further by explaining how to attain theoretical scale invariance of the object component model, which in practical terms enlarge the scale robustness.

3.4 Providing scale and rotation invariance

In the previous section we have derived an object component descriptor able to successfully detect facial components. Although we have shown experimentally its tolerance to scale and rotation changes of the image components, this was only valid for a small range. In this section we propose methods that provide invariance to those transformations.

3.4.1 Scale invariance

In order to provide scale invariance to the local descriptor in Equation (3.16), we first analyze how a Gabor response behaves with scale changes. Following the reasoning proposed in [59],

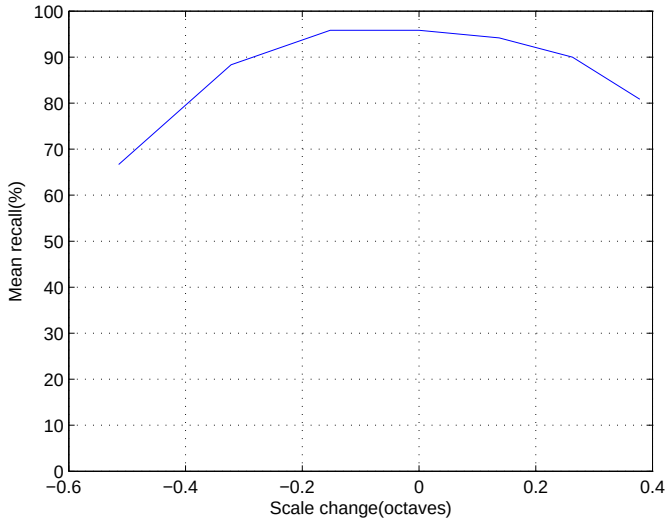


Figure 3.7: Scale robustness test of Gabor filter based local descriptor

we consider two images: $I(x, y)$ and an homogeneously scaled version of $I(x, y)$. The new image is scaled by a factor a as $I_s(x, y) = I(ax, ay)$. The response of the scaled image at point (x_0, y_0), is

$$\begin{aligned}(I_s * g_{\theta,f,\sigma})(x_0, y_0) &= (g_{\theta,f,\sigma} * I_s)(x_0, y_0) \\ &= \int\int g_{\theta,f,\sigma}(x, y) I_s(x_0 - x, y_0 - y) dxdy \\ &= \int\int g_{\theta,f,\sigma}(x, y) I(ax_0 - ax, ay_0 - ay) dxdy \end{aligned} \tag{3.18}$$

We now let $\tilde{x} = ax$ and $\tilde{y} = ay$. We then have $dx = d\tilde{x}/a$ and $dy = d\tilde{y}/a$. By making substitutions in the Gabor function of Equation (3.7),

$$(I_s * g_{\theta,f,\sigma})(x_0, y_0) = (I * g_{\theta,f/a,\sigma a})(ax_0, ay_0). \tag{3.19}$$

From Equation (3.19) we can see that the Gabor response remains constant in the scaled image if we change both the width parameter σ of the Gabor filter to σa and the frequency value f to f/a.

Thus, if we are able to estimate the scale factor a, we can calculate the adjusted values of the width σ and spatial frequency f in order to compute the object component descriptor in the scaled image. A common approach to compute the scale factor a is to define an intrinsic scale at the interest point of the object component. For a given object component we

compute the intrinsic scale σ_{int} in the training set, for example using the λ-signature method presented in Chapter 2. Then, to compute the component descriptor in scaled images, we take the following steps:

1. compute the intrinsic scale s in the new image.
2. compute the scale factor $a = s/\sigma_{\text{int}}$, and
3. compute an adequate approximation for the local maxima parameters of Equation (3.15),

$$\begin{aligned} {}_s\hat{\sigma}_{l,j}^{\max} &= \hat{\sigma}_{l,j}^{\max} a, j = 1, 2; l = 1, \ldots, L \\ {}_s\hat{f}_{l,j}^{\max} &= \hat{f}_{l,j}^{\max}/a, \end{aligned} \tag{3.20}$$

where the left subscript s in ${}_s\sigma_{\text{int}}$, ${}_s\hat{\sigma}_{l,j}^{\max}$ and ${}_s\hat{f}_{l,j}^{\max}$ stands for scaled image.

The new width and frequency in Equation (3.20) are the parameters to use now in order to compute the feature vector of Equation (3.16). In the next section we will see how to add invariance to image plane rotations.

3.4.2 Rotation invariance

The object component model in Equation (3.16) is obtained by sampling the angle θ of the Gabor filter uniformly. Thus, we must shift the angles to compute the feature vector in rotated versions of the object component. When we discussed the model robustness to image rotation in Section 3.3.2, we mentioned that if the rotations were known, we could match the object by shifting the orientation parameter θ by the corresponding amount.

Since the rotation is unknown, we could adopt an approach similar to the scale invariance and define an intrinsic orientation. However, the common approaches for computing the intrinsic orientation are based on the global maximum of the histogram of gradient orientations [70], a procedure that is very sensitive to noise and small image variations. Instead, we prefer to match all possible orientations, a procedure that is more demanding computationally, but very robust. We address the rotation invariance by matching all possible feature vectors shifted in the orientation parameter and choosing the θ-shifted vector that is the closest to the object component model.

Summarizing the procedure to match a feature vector in a scale and rotation invariant manner, we compute:

1. the feature vector parameters $\hat{\sigma}_{l,j}^{\max}$ and $\hat{f}_{l,j}^{\max}, j = 1, 2; l = 1, \ldots, L$, using the intrinsic scales, as shown in Equation (3.20).

2. rotated versions of the feature vector and finally picking the closest vector to the model.

3.5 Tests

We perform detection and location of facial components. The tests aim to: (i) verify if the adaptive Gabor bank object component model is able to correctly classify an image pixel, (ii) verify the rotation invariance of the object component model, and (iii) verify the scale invariance of the object component model.

We perform an additional test in order to illustrate the effect of the top-down saliency function introduced in Chapter 2 on the performance of the local descriptor presented in this chapter. We evaluate the effect of the interest point selection done by the top-down saliency model SM_c on the object component classification, using the adaptive Gabor bank descriptor to match the facial components in new images.

We use 82 subjects from the AR face database [77], where half of them are used for learning the model of five facial components and the remaining half for the component classification. We consider left eye, right eye, nose, left nostril, and right nostril as the facial landmarks. The object component model is learnt in a supervised manner and the model is computed in ground truth points.

3.5.1 Classification of object components

In the training stage we compute the model $(\mu_{\mathbf{u}_c}, \Sigma_c)$ of a facial component. Then, in the matching stage we compute the Mahalanobis distance between the facial component model and the local descriptor $\mathbf{u}(x, y)$ at image point $I(x, y)$. In order to classify the pixel (x, y) as facial component c, we utilize the chi-squared test confidence probability to accept or reject the local descriptor $\mathbf{u}(x, y)$ being drawn from the facial component model's distribution. The chi-squared test relates the feature vector size ("degrees of freedom") and the Mahalanobis distance value to a confidence probability value. Thus, we can accept or reject a local descriptor $\mathbf{u}(x, y)$ with a certain confidence by choosing the correspondent Mahalanobis threshold. The retrieved image points are those below the Mahalanobis distance threshold and those points are marked as facial components. To quantify the performance in facial component classification, we compute recall and precision for each facial component

$$recall = \frac{\#correct\ hits}{\#true\ positive\ components}, \tag{3.21}$$

$$precision = \frac{\#correct\ hits}{\#total\ hits}. \tag{3.22}$$

The recall rate of Equation 3.22 represents the number of object components detected correctly, so a feature vector with maximum recall will not miss any component. The precision rate represents the number of matches (hits) that find object components, so a feature vector with maximum precision will not find false positive matches.

We set the confidence threshold to 99.9% and compute the Mahalanobis distance in the test set images. We mark a hit if there is a facial component found within a circle of radius $r = 4$ pixels around the groundtruth component location. The hits located outside the groundtruth proximity circle are marked as false positives. Table 3.3 presents the results when classifying pixels as eye center, nose center, and nostril center.

Facial Point	Recall(%)	Precision(%)	130°Rot Recall
Left eye	100	64.36	97.56
Right eye	97.56	50.33	97.56
Nose	92.68	79	92.68
Left nostril	87.8	60.68	87.8
Right nostril	82.92	72.32	82.92

Table 3.3: Precision and recall rates of facial component classification.

Recall rates in Table 3.3 show the very good recognition capabilities of the adaptive Gabor filter-based descriptor. The precision rates reflect the amount of false positive detected components in the images, due to the exhaustive search performed (the entire image). We will explain in the next section how to improve the precision rates by using the top-down saliency model.

Scale invariance

To check the invariance to scale transformations, we compute the recall rate in rescaled images maintaining the object model learned in the original size images. In Table 3.4 we can see the average recall of all facial landmarks.

scale change(octaves)	Recall(%)
-0.5	83.19
-0.25	92.19
0	92.19
0.25	92.19
0.5	91.14

Table 3.4: Scale invariance test

Although we have demonstrated theoretical scale invariance of the Gabor response, due to discretization and sampling effects, invariance is not possible for the whole range of scales in real images. To attain full scale invariance, it is necessary to apply a multi-scale approach, computing the component model every octave (a very typical value [70]).

Rotation invariance

The rotation robustness of the component descriptor is tested in Section 3.3.2 and the results plotted in Figure 3.5. In that section we show experimentally that an appropriate circular shift of the component descriptor is a suitable method to match rotated facial components. However, the rotation robustness tests sampled angles only contained in the Gabor filter-bank parameters.

The procedure to match angles not sampled in the model is presented in Section 3.4.2 and we illustrate the application of the method by computing the recall rate in the image test set rotated by the angle 130°, keeping the object model learned in the standard pose images. The rightmost column of Table 3.3 shows that the recall remains approximately constant for an angle that is not sampled in the model.

3.5.2 Top-down saliency + adaptive Gabor filter-based descriptor

This group of tests integrates the saliency model based on the λ-signature presented in Chapter 2 and the facial component classification procedure presented in the previous section. In the initial stage, the saliency model makes a preselection of candidates for every facial component. Then, at the points selected by the saliency model the local descriptor $\mathbf{u}(x, y)$ is computed, to match components and classify points as components.

The approach of this section adds an extra step to the experimental setup described in Chapter 2 (Section 2.7.2), the component classification. Considering the additional step, for every object component c we perform:

1. Application of the Local maxima of LoG operator at several scales. This procedure provides an initial set of interest points $IP = \{(x_1, y_1), \ldots, (x_J, y_J)\}$ as in Equation (2.22).

2. Matching of the saliency model SM_c (Equation 2.20) with the scale invariant signature $\tilde{\Lambda}S_{x_j,y_j}$ (Equation 2.16) computed in the interest point set, keeping interest point locations with positive matches. The resulting interest point locations form a subset of IP and we denote the subset as $IP_c = \{(x_{c1}, y_{c1}), \ldots, (x_{cs}, y_{cs}), \ldots, (x_{cS}, y_{cS})\}$.

3. For every point in IP_c, the computation of the filter-based descriptor $u(x_{cs}, y_{cs})$ of Equation (3.16) and classifcation of the interest points using the Mahalanobis distance with the chi-squared test.

This algorithm reduces the computational complexity during facial component matching. While the test described in the previous section (3.5.1) computes the adaptive Gabor bank descriptor at every image pixel, the test presented in this section computes the descriptor in a selected set of interest points. We see that the top-down saliency model maintains practically the same recall and improves substantially the precision rate of the classification. Comparing the precision results in Table 3.3 *vs.* Table 3.5, we remark that the top-down saliency function removes around 10% of false positives that the object component classification method is not able to label correctly.

Facial Point	Recall(%)	Precision(%)	Precision from Table 3.3 (%)
Left eye	100	74.63	64.36
Right eye	97.56	57.99	50.33
Nose	90.24	100	79
Left nostril	87.8	67.94	60.68
Right nostril	82.92	94.47	72.32

Table 3.5: Top-down saliency and filter-based description tests

3.6 Discussion

We have introduced an adaptive Gabor bank local descriptor for object components. The presented descriptor belongs to the sparse type of filter-based representation, allowing lower feature vector sizes and more efficient matching procedures. While common approaches for Gabor filter bank descriptors adopt fixed filter parameters to represent local appearances, we introduce an automatic filter parameter selection method to compute local descriptors adapted to the particular object components. The technique for parameter selection is based on the Information Diagram concept [53] that is extended in this thesis to consider optimization along all dimensions of the Gabor function parameters. The adaptive Gabor bank descriptor presented is characterized by:

- selection of the Gabor filter parameters with largest energy to represent a particular object component,
- invariance to image rotations, and

- high tolerance to image scalings.

We have used the adaptive Gabor bank descriptor together with the top-down saliency model proposed in Chapter 2. The results obtained allow to conclude that the top-down saliency model:

- reduces the computational complexity of the local descriptor computation and matching procedure, maintaining recall rates practically equal to the recall rates computed in entire images, and
- improves classification precision of the adaptive Gabor bank descriptor.

We have explored the sparse filter-based representations using Gabor filters, small size local descriptors with efficient matching methods. We see that our proposed descriptor has excellent recall rates, but the precision rates are just acceptable. Thus, it is suited for applications with constant background, e.g. in human-machine interfaces in controlled situations. For other types of applications in more general environments, in the next Chapter we will explore histogram-based methods and propose improvements based again on the automatic selection of Gabor filters.

Chapter 4

Histogram-based descriptors

We have proposed a sparse (small size) descriptor using the filter-based approach, attaining very good detection results. In order to keep the recognition rates in more challenging applications such as recognition in cluttered images, it would be necessary to change the descriptor sampling approach, but maintaining the small size constraint for the descriptors. The appropriate representations that hold these requirements are the histogram-based descriptors, the subject of study in this chapter.

Histogram-based methods for computing local image descriptors follow a sequence of steps: (i) The initial step is to select interest points in the scale space (e.g. Hessian, Harris) and compute the image gradient in the neighborhood of interest points (e.g. pixel differences, Canny detector); (ii) the descriptor is then obtained by splitting the interest point neighborhood into smaller regions (e.g. cartesian grid, log-polar grid), and (iii) finally for every subregion the histogram of the gradient orientation is computed with an appropriate information selection procedure (e.g. weighting, PCA).

To date, the most remarkable descriptor in terms of distinctiveness is the SIFT local descriptor [70], which computes the image gradient from pixel differences, subdivides the interest point regions in a cartesian grid, and for each subregion, computes the gradient orientation histogram weighted by the gradient magnitude. The descriptor is the concatenation of all subregion's histograms, followed by a unitary normalization.

In this chapter we present an alternative approach for gradient computation using smooth derivative filters. In scale-normalized image regions, gradient computation using pixel differences, as in [70], is quite sensitive to noise and other artifacts induced by the image sensor and the normalization procedure. One common approach to diminish the noise sensitivity is to compute smoother approximations of the image derivatives using filters. We use Gabor filters, which have been shown to approximate any image directional derivative [58], by suitably tuning their parameters. We propose a methodology to define the filters' parameters

based on local maxima of the magnitude of the filter response. We analyze the response for several filter widths, selecting the width in which the local maximum is located [90].

Using Gabor functions as smooth filters, our approach improves the distinctiveness of the SIFT local descriptor. To quantify the impact of our approach we use the local descriptor evaluation framework proposed in [83]. Several types of images (natural, structured) and image transformations (viewpoint, scale, blur, JPEG, illumination) are employed in the evaluation process. This evaluation framework is presented in Section 4.1. Then, in Section 4.2 we present our approach based on the SIFT descriptor and Gabor parameter selection for gradient computation. Section 4.3 shows results of the comparison between our method and the original SIFT descriptor, under the framework presented in Section 4.1. Finally, in Section 4.4 we draw some conclusions.

4.1 Local descriptor evaluation

In this section we describe the main steps of the framework proposed in [83] to compare local image descriptors. The method can be summarized as follows:

1. Several image pairs are used for evaluation, each having a particular type of image transformation (blur, view-point, illumination, JPEG compression, and zoom+rotation). Each pair is obtained by taking two pictures of the same object in different conditions (position, camera/image settings).

2. For each pair a projective transformation H between the two images is computed by standard homography estimation methods. Corresponding regions between images are called covariant.

3. Salient image regions are computed using invariant region detectors, like the Harris or Hessian detectors. This process outputs elliptic regions in the two images that are good candidates for posterior matching. Knowing the ground truth projective transformation H between the images, a *correspondence test* is proposed to evaluate the quality of the invariant image detection process.

4. Candidate image regions are normalized for affine and illumination transformations using, respectively, the elliptic regions' parameters computed in the previous steps and image region gray level statistics.

5. Each candidate image region is represented by the several descriptors under comparison.

6. A *matching test* determines if two candidate regions (one on each image of the pair) are similar. Three different matching methods are employed: (i) thresholded euclidean distance between the two descriptors, (ii) nearest-neighbor, and (iii) nearest-neighbor distance ratio. Based on the ground truth data, matches are classified as correct or false.

7. As in the previous chapters, an evaluation metric is defined, based on precision (ratio between correct matches and all matches) and recall (ratio between correct matches and correspondences).

In the following sections we provide additional details on each of these steps. We start, in Section 4.1.1, by describing the types of images employed in the tests and the homography computation for the generation of ground truth data. In Section 4.1.2 we focus on the computation of salient image regions with several invariant region detectors. The local region normalization, descriptor computation and matching procedures are detailed in Section 4.1.3, and finally, the computation of the overall evaluation metric (*recall* vs. $1-precision$ curves) is described in Section 4.1.4.

4.1.1 Image data set

Figure 4.1 shows the test set images used to perform the local descriptor evaluation. These are the same as used in [83] for the sake of comparison with the other methods. For each image, one of five possible image transformations is applied: Zoom + rotation, viewpoint, image blur, JPEG compression, and illumination. For viewpoint transformations, scale + rotation, and image blur, two classes of images are considered: (i) *natural* images containing a large amount of randomly oriented textures, and (ii) *structured images* containing many distinctive long edge boundaries. In the case of JPEG compression and illumination transformations, only images from the *structured* type are employed.

An image pair is created for each transformation, containing both the reference image and the transformed image. In the viewpoint (locally affine) transformation, the camera position moves from a fronto-parallel view to one with foreshortening at 40 degrees to the camera. In the scale transformation, the scale factor is changed for 1.9 in the image of Figure 4.1(a) and 2.5 for Figure 4.1(b). In the image blur transformation, the focus ratio between the reference and transformed image is 4. The JPEG transformation keeps 10% of the quality of the original image. The illumination transformation varies the camera aperture by a factor of 4.

For the generation of ground truth data (computing the correct matches between the two images), each pair of images is related by a homography. The homography is computed

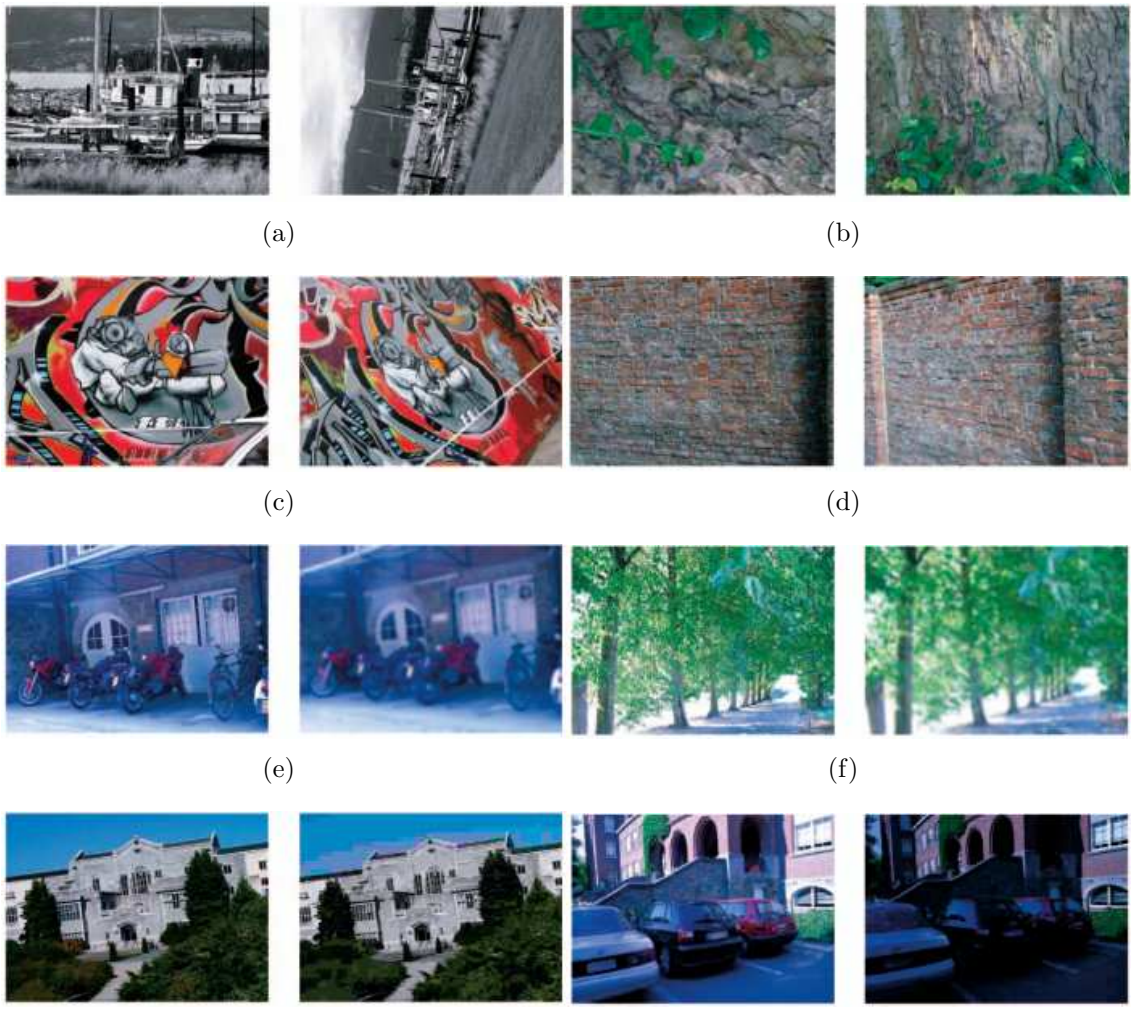


Figure 4.1: Data set used for local image descriptor evaluation. Zoom + rotation 4.1(a) and 4.1(b), viewpoint 4.1(c) and 4.1(d), image blur 4.1(e) and 4.1(f), JPEG compression 4.1(g) and illumination 4.1(h)

in two steps: (i) a first approximation is obtained using manually selected points, then the transformed image is warped with this homography, and (ii) a robust small baseline homography estimation algorithm is used to compute the residual homography between the reference image and the warped one.

4.1.2 Invariant region detectors

The following detectors have been considered "appropriate" for region matching. We will use them in our tests:

- The Harris-laplace detector [80] computes local maxima of a *cornerness* metric using the scale adapted second moment matrix [67] to find initial candidates. Then for every candidate, it is checked if there is a local maximum in scale of the normalized Laplacian of Gaussian. The regions detected are corners and junctions covariant to scale and rotation changes.

- The Hessian-laplace detector [70, 82] computes the local maxima of Hessian operator to locate candidates spatially. Candidates that attain local maxima in scale of normalized Laplacian of Gaussian are selected as interest regions. The method provides blobs and ridges covariant to scale and rotation changes.

- Harris-affine detector [82] is an affine extension of Harris-laplace. The final step is to compute the shape adaptation matrix [67] to perform an affine normalization. The regions detected are corners and junctions covariant to affine transformations up to a rotation factor.

- Hessian-affine detector [84] is an affine extension of Hessian-laplace. The final step is to compute the shape adaptation matrix [67] to perform an affine normalization. The regions detected are blobs and ridges covariant to affine transformations up to a rotation factor.

These methods provide not only the localization of the salient regions but also geometrical information regarding the intrinsic scale of the image region. Then, the region's dominant orientation is obtained by selecting the peak of the gradient histogram. With this information, each image region can be associated to an ellipse (R_μ) representing its dominant shape.

To evaluate the quality of the region detectors, a correspondence test is defined. Two image regions R_{μ_a} and R_{μ_b} are corresponding if the overlap error is less than threshold ϵ_0,

$$1 - \frac{R_{\mu_a} \cap R_{H^T \mu_b H}}{R_{\mu_a} \cup R_{H^T \mu_b H}} < \epsilon_0. \tag{4.1}$$

In the previous equation R_μ is the elliptic region defined by $x^T \mu x = 1$, where μ has the ellipse parameters, and H is the homography between images. For all tests performed in this chapter, we have fixed $\epsilon_0 = 0.5$.

4.1.3 Local image descriptors

To represent the detected regions in a suitable way for matching, an extended description of its photometric properties must be provided. Before computing the local descriptors, every local image region must be normalized for invariance to affine transformations and illumination. Geometrical normalization is done using the region elliptic parameters, computed in the previous step and illumination normalization is obtained by performing a contrast stretching using the mean and standard deviation of the region's gray values.

After region normalization and descriptor computation, the matching step evaluates the similarity between the descriptors (feature vectors) of image regions. Three matching procedures are compared: threshold-based, nearest neighbor, and nearest neighbor distance ratio. In the case of threshold-based matching, two descriptors (feature vectors) $\mathbf{u}_a$ and $\mathbf{u}_b$ are matched if the Euclidean distance is below a threshold. In the case of nearest neighbor, a match exists if u_b is the nearest neighbor to $\mathbf{u}_a$ and the Euclidean distance between descriptors is below a threshold. In the case of nearest neighbor distance ratio, we have the descriptor $\mathbf{u}_a$, the nearest neighbor $\mathbf{u}_b$, and the second nearest neighbor $\mathbf{u}_c$. The descriptors $\mathbf{u}_a$ and $\mathbf{u}_b$ are matched if $||\mathbf{u}_a - \mathbf{u}_b||/||\mathbf{u}_a - \mathbf{u}_c|| < t$. The threshold-based method may assign several matches to the same descriptor, while the other two methods assign one match only to each descriptor.

4.1.4 Overall evaluation

The overall matching process cascades two main phases: detection of salient points and matching the regions' descriptors. To evaluate the overall matching process, a *recall* versus $1 - precision$ curve is computed for each image pair. The recall of the regions detected in two images is defined as:

$$recall = \frac{\#correct\,matches}{\#corresponding\,regions}. \tag{4.2}$$

The ratio between false matches and the total number of matches is given by $1 - precision$ value:

$$1 - precision = \frac{\#false\,matches}{\#correct\,matches + \#false\,matches}. \tag{4.3}$$

After completing the steps of Equations (4.2)-(4.3) above, one is able to compare the matching performance of any local descriptor using the *recall* versus $1 - precision$ curve. A perfect descriptor would give $recall = 1$ for any precision. So the descriptor with largest area in the *recall* vs. $1 - precision$ curve is considered to have the best performance. In the next section we describe in detail our local descriptor proposal.

4.2 Improving a histogram-based descriptor using Gabor filters

In this section we first review the SIFT local descriptor computation in a normalized image region. Then we present a modification of the SIFT descriptor, using odd Gabor filters to compute first order image derivatives.

4.2.1 SIFT local descriptor

In the original formulation of the SIFT descriptor [70], a scale-normalized image region is represented with the concatenation of gradient orientation histograms relative to several rectangular subregions. First, to obtain the scale-normalized patches, a salient region detection procedure provides image point neighborhoods. The saliency function is computed from the scale-space of Difference of Gaussians (DoG) and the image regions (position and scale) are selected by the local extrema in the scale-space. In order to compute the local descriptor, the regions are scale normalized and the derivatives I_x and I_y of the image I are computed with pixel differences:

$$\begin{aligned} I_x(x,y) &= I(x+1,y) - I(x-1,y) \\ I_y(x,y) &= I(x,y+1) - I(x,y-1). \end{aligned} \tag{4.4}$$

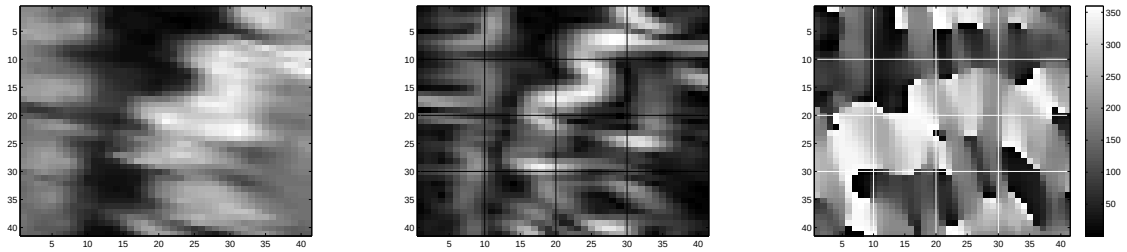
Then the image gradient magnitude and orientation are computed for every pixel in the image region:

$$M(x,y) = \sqrt{I_x(x,y)^2 + I_y(x,y)^2} \tag{4.5}$$

$$\Theta(x,y) = \tan^{-1}(I_y(x,y)/I_x(x,y)). \tag{4.6}$$

The interest region is then subdivided in a rectangular grid. Figure 4.2 shows examples of the gradient magnitude and orientation of an image region and its corresponding 16 subregions.

The next step is to compute for each subregion the histogram of gradient orientation,



(a) normalized image region (b) $M(x, y)$ (c) $\Theta(x, y)$

Figure 4.2: Example of gradient magnitude and orientation images

weighted by gradient magnitude. Orientation is quantized into 8 bins and each bin is set with the sum of the windowed orientation difference to the bin center, weighted by the gradient magnitude:

$$h_{r_{(l,m)}}(k) = \sum_{x,y \in r_{(l,m)}} M(x,y)(1 - |\Theta(x,y) - c_k|/\Delta_k), \quad \Theta(x,y) \in \text{bin } k, \tag{4.7}$$

where c_k is the orientation bin center, Δ_k is the orientation bin width, and (x, y) are pixel coordinates in subregion $r_{(l,m)}$. The SIFT local descriptor is the concatenation of the several gradient orientation histograms for all subregions:

$$\mathbf{u} = (h_{r_{(1,1)}}, \ldots, h_{r_{(l,m)}}, \ldots, h_{r_{(4,4)}}) \tag{4.8}$$

The final step is to normalize the descriptor in Equation (4.8) to unit norm in order to reduce the effects of uniform illumination changes.

The gradient orientation is not invariant to rotations of the image region. To provide orientation invariance, Lowe proposed to compute the orientation of the image region and set the gradient orientation relative to the region's orientation. The orientation of a region is given by the highest peak of the gradient orientation histogram of the image region.

We have based our work on an approach similar to the one described here. However, the gradient computation in the original SIFT descriptor is done with pixel differences which are very sensitive to noisy measurements and not adapted to the natural scale of edges in the normalized region. In next section we explain an alternative way to compute the image derivatives of Equation (4.4), using Gabor filters with properly tuned parameters.

4.2.2 Gabor functions as smooth image derivative filters

The computation of image derivatives with pixel differences is an inherently noise sensitive process. Pixel differences implement a *high-pass* filtering operation on the image spectrum, amplifying the high frequency range, which is mainly composed by noise. To avoid such sensitivity, it is common to combine a *low-pass* filter (image blurring or smoothing) with the *high-pass* derivative filter, resulting in a *band-pass* filter, which we denote by *smooth derivative filter*. This effect can be implemented by either pre-smoothing the image followed by the derivative computation, or by convolving the image with a *band-pass* filter combining both phases. The important question to address at this point is "how much blurring should we apply to the image ?", or equivalently, "which frequency band should the *band-pass filter* focus on ?"

Several smooth derivative filters have been proposed for image filtering. Both Gaussian derivatives [58] and Gabor filters [38, 21] are common choices because of their properties and the availability of fast computation methods [123]. Gaussian derivatives [58] are smooth filters that can compute the image derivatives of any order. They have good noise attenuation properties due to an implicit image Gaussian filtering. In Figure 4.3 we show examples of the

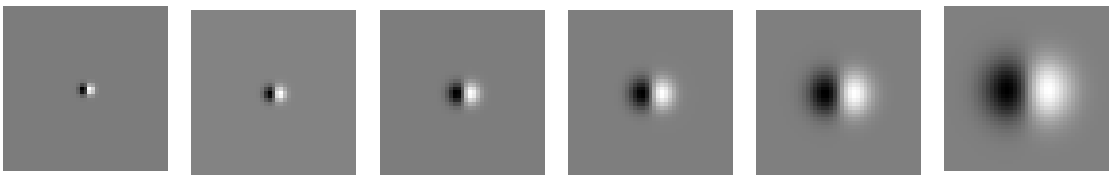


Figure 4.3: Examples of Gaussian first order kernel in the x direction for $\sigma = \{2\sqrt{2}/3,\ 4/3,\ 4\sqrt{2}/3,\ 8/3,\ 8\sqrt{2}/3,\ 16/3\}$.

Gaussian first order derivative kernel. We use Gabor filters for the computation of smooth image derivatives due to the following facts:

- With appropriate parameters, odd Gabor filters can approximate odd-order Gaussian directional derivatives [58].
- Gabor filters have a larger number of parameters than Gaussian derivatives, thus being more easily customized to each particular purpose [101, 22, 122, 75, 68, 50, 24]. Previous works have shown the advantage of Gabor filter parameter selection in edge computation [93, 124] by defining an edge threshold criterion based on Gabor filter parameters.

Notice that the first fact listed above tells us that the best performance with Gaussian derivative filters can also be achieved with Gabor filters, and the second fact suggests that a more

careful parameter tuning of the Gabor parameters may possibly lead to better performance.

4.2.3 Gabor filters for image derivative computation

Gabor functions are defined by the multiplication of a complex exponential function (the carrier) and a Gaussian function (the envelope).

$$g_{x,y,\theta} = \frac{1}{2\pi\sigma_1\sigma_2}\cdot\exp\left(-\frac{(x\cos\theta + y\sin\theta)^2}{2\sigma_1^2} - \frac{(y\cos\theta - x\sin\theta)^2}{2\sigma_2^2}\right)\cdot\exp\left(i\frac{2\pi}{\lambda}(x\cos\theta + y\sin\theta)\right) \tag{4.9}$$

In the previous expression, (x, y) are the spatial coordinates, θ is the filter orientation, λ is its wavelength, and σ_1 and σ_2 are the Gaussian envelope standard deviations, oriented along directions θ and $\theta + \pi/2$, respectively.

To compute the first order image derivatives I_x and I_y we will use the odd (imaginary) part of the filter. The orientations will be $\theta = 0$ and $\theta = \pi/2$ for, respectively, the horizontal and vertical derivatives. To approximate the shape of an odd Gabor Filter to that of a Gaussian derivative, we set $\sigma_1 = \sigma_2 = \sigma$ and we introduce $\tilde{\lambda} = \lambda/\sigma$, a variable that is proportional to the number of wave periods within the filter width. By fixing an appropriate $\tilde{\lambda}$ value, we will obtain an expression of the Gabor filter with a single parameter, the filter width σ.

If we look at the shape of the first order Gaussian derivatives at any scale in the derivative direction, there is one wave period within the spatial support of the filter, which roughly corresponds to $\lambda = 6\sigma$. Replacing this value in $\tilde{\lambda} = \frac{\lambda}{\sigma}$ yields $\tilde{\lambda} = 6$. By replacing $\sigma = \sigma_1 = \sigma_2$ and $\tilde{\lambda} = 6$ in Equation (4.9), we obtain the filter being used in the remainder of the chapter:

$$g_{x,y,\theta}(\sigma) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)\cdot\sin\left(\frac{2\pi}{6\sigma}(x\cos\theta + y\sin\theta)\right), \tag{4.10}$$

where $\theta = 0$ computes I_x, and $\theta = \pi/2$ computes I_y. The choice of σ will be done by an optimization procedure, based on the filter energy at locations with high gradient magnitude.

4.2.4 Scale selection

In this section we propose a methodology to select a value for the scale parameter σ, such as to maximize the energy output of the smooth derivative filters in the analysis of the normalized regions obtained in the interest point selection procedure. We notice that, at this point, we have image regions that are already scale-normalized, therefore the scale-selection procedure we are proposing here should choose one single scale value for all regions.

Figure 4.4 shows examples of the odd Gabor filter to compute the I_x at several σ values. In

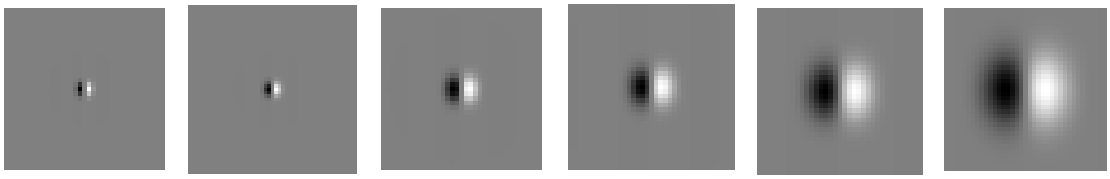


Figure 4.4: Examples of odd Gabor functions at $\theta = 0$, $\gamma = 6$, and $\sigma = \{2\sqrt{2}/3,\ 4/3,\ 4\sqrt{2}/3,\ 8/3,\ 8\sqrt{2}/3,\ 16/3\}$.

order to select the best scale, we will use the gradient magnitude over all selected features in all images in the data set, due to its key role of weighting the gradient orientation histogram in the SIFT computation. In fact, the scale-normalized gradient magnitude has been used to measure edge strength in scale-space [64]. However, this measure is not very stable at large scales, sometimes leading to the selection of scale values higher than the actual feature scale [64]. This problem has been addressed in the context of edge scale selection [65], using the concept of γ-normalized derivatives.

We have made some preliminary test with this methodology, but the results were not promising, mainly because the features obtained in the interest point selection phase are not only edges, but also blobs, corners, junctions, and other structures. Additionally, the image regions we are considering are already scale-normalized, so the scale selection procedure is a local search, as opposed to γ-normalized derivatives in [65]. Therefore, we propose the following methodology to avoid the bias toward large scales in the scale-normalized gradient magnitude by:

- considering independently the components of the normalized gradient magnitude, and
- biasing the scale selection criterion to smaller scale values for each component, to avoid the non-decreasing behavior of the normalized derivatives for large scales [64].

Following these criteria, we pick the Gabor filter with largest energy in the x and y directions and, from these, we select the smaller scale:

$$\begin{aligned}
\hat{\sigma}_x &= \arg\max_{\sigma} |(I * g_{x_i,y_i,\theta=0}(\sigma))| \\
\hat{\sigma}_y &= \arg\max_{\sigma} |(I * g_{x_i,y_i,\theta=\pi/2}(\sigma))| \\
\hat{\sigma}(x_i, y_i) &= \min(\hat{\sigma}_x, \hat{\sigma}_y), \\
I_x(x_i, y_i) &= (I * g_{x_i,y_i,0}(\hat{\sigma}))(x_i, y_i) \\
I_y(x_i, y_i) &= (I * g_{x_i,y_i,\pi/2}(\hat{\sigma}))(x_i, y_i).
\end{aligned} \tag{4.11}$$

where (x_i, y_i) is a point in the scale-normalized region, and $\hat{\sigma}$ is the adequate filter width at position (x_i, y_i).

Computational complexity

The local minima selection of Equation (4.11) has an obviously higher computational complexity than the pixel difference of Equation (4.4). In a scale-normalized image of size $S \times S$, the complexity of the pixel difference and filtering is $O(S^2)$, while the odd Gabor scale selection of Equation (4.11) has a complexity value of

$$O(S^2 \times (C \times F + 2F + 1)), \tag{4.12}$$

where C is the number of operations per pixel to compute the response of one Gabor filter and F is the number of Gabor filters applied. Using the state-of-the-art fast implementation of Gabor filters, $C = 60$[1] operations per pixel [6, 7], and F depends on the type of multi-scale implementation and the size of the normalized region. As we are dealing with scale-normalized regions, the search along F scales of Eqs. (4.11-4.11) can be replaced by a single scale suitable for all normalized images, thus yielding a complexity of $O(S^2 \times C)$.

4.3 Experimental results

First we address the selection of a single scale value of the Gabor filter suitable to compute the image derivatives for all image regions. Then, we present the results of image region matching experiment and evaluate the advantages of smooth derivative filters in SIFT computation.

4.3.1 Gabor filter scale selection

Aiming to reduce the computational complexity presented in Equation (4.12), we select a single filter suiting all cases. The single filter selection reduces the complexity of the image derivative computation from $O(S^2 \times (C \times F + 2F + 1))$ to $O(S^2 \times C)$. We compute the relative frequency (i.e. histogram) of the filter width $\hat{\sigma}$ in Equation (4.11), using all the scale-normalized image regions of the image data set presented in Figure 4.1. To avoid noisy $\hat{\sigma}$ values, we pick pixels with gradient magnitude above a certain threshold. We plot the marginalized (structured and textured) histograms and the total histogram in Figure 4.5. When comparing structured versus textured images, we observe that in the case of textured images the bins located at the left side of the histogram peak are all larger than the

[1] Considering an isotropic and non-zero mean Gabor filter implementation

Pixel difference of Eqs. (4.4-4.4)	0.44 ms
Multi-scale optimization (Gabor) of Eqs. (4.11-4.11)	9.75 ms
Single scale (Gabor) of Eqs. (4.13-4.13)	1.01 ms

Table 4.1: Execution time of C implementations, in a Pentium 4, 2.80 Ghz. Average value of the x derivative computation for all the normalized regions (size:41×41) selected in the images of Figure 4.1.

equivalent bins in the structured images histogram. This is an expected behavior because the high gradient magnitude points in very textured images have a very small spatial support, while in structured images the points with high gradient magnitude have a larger spatial support. We also notice the difference of peak location between structured ($\hat{\sigma} = 1.88$) and textured ($\hat{\sigma} = 1.58$) images.

Although we biased the filter width selection to small values using Equation (4.11), it still will select high filter width values in some of the image points (around 10% of image pixels), blurring the image gradient in some regions. This behaviour would lead to the loss of important histogram information in some subregions. In order to avoid these high filter width values, we select the peak of the $\hat{\sigma}$ histogram in textured images (Figure 4.5).

$$\begin{aligned} I_x(x,y) &= (I * g_{x,y,0}(1.58))(x,y) \\ I_y(x,y) &= (I * g_{x,y,\pi/2}(1.58))(x,y). \end{aligned} \tag{4.13}$$

Equation (4.13) provides a fast approximation of the scale selection of Equation (4.11), keeping the advantage of a smoother image derivative approximation versus the pixel differences of Equation (4.4). In the next sections we present the performance improvement of the SIFT descriptor by using Equation (4.13). However, we pay the price of performance improvement by increasing the computational load of the image derivative computation, as shown in Table 4.1. Despite that the theoretical complexity analysis indicates a 60 times slow down with our approach, in practice we verified that it only slows down 2-3 times, thus maintaining a real-time functionality. The explanation may be related to the pixel access times to perform the subtraction, that were not considered in the theoretical analysis. Additionally, the fixed computational cost of the image normalization will further smooth out the differences between the two methods.

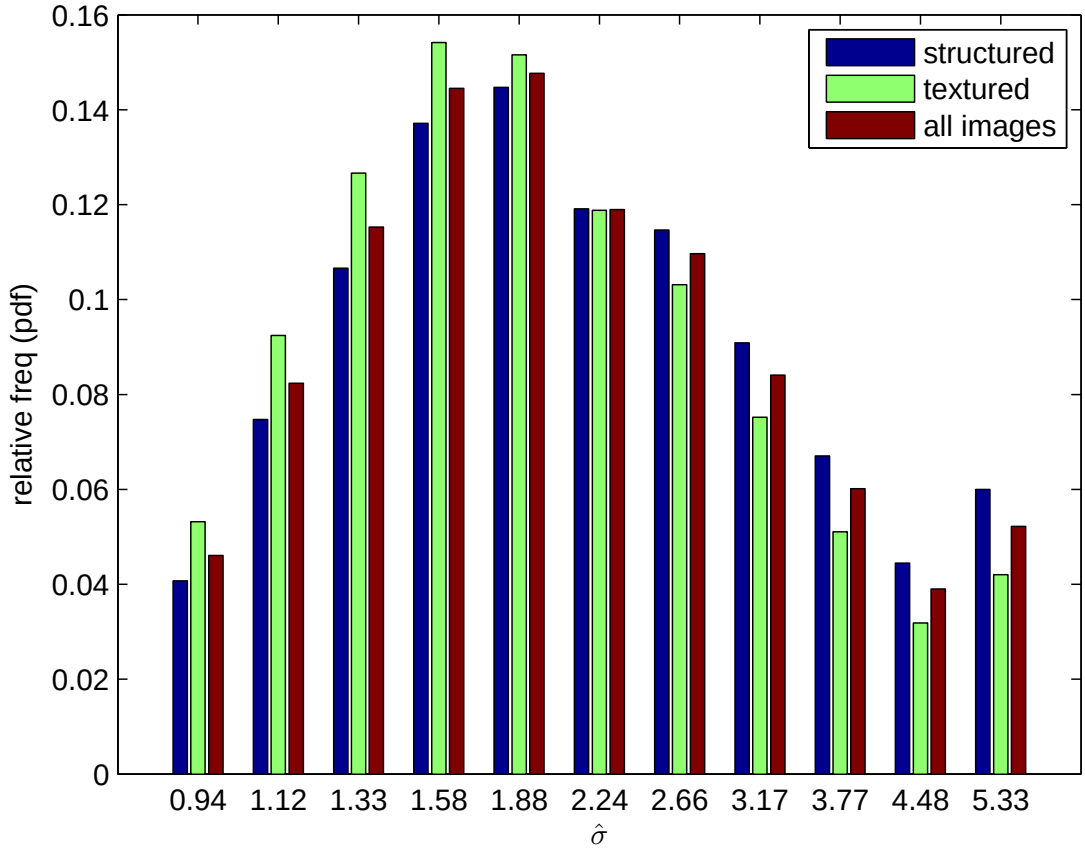


Figure 4.5: Histograms of $\hat{\sigma}$ for various image types.

4.3.2 Image region matching

In this set of tests, we compute *recall* vs $1 - precision$ curves for all types of: (i) image transformations, (ii) image detectors, and (iii) structured and textured images. We show in Figures 4.6-4.7 samples of the *recall* vs $1 - precision$ curves, remarking that the curve of our descriptor is always located above the original SIFT curve for the threshold-based criterion. We notice a similar behavior for all experiments (in appendix B), improving the SIFT matching performance by the utilization of smooth derivative filters (Gabor filters).

4.3.3 Discussion

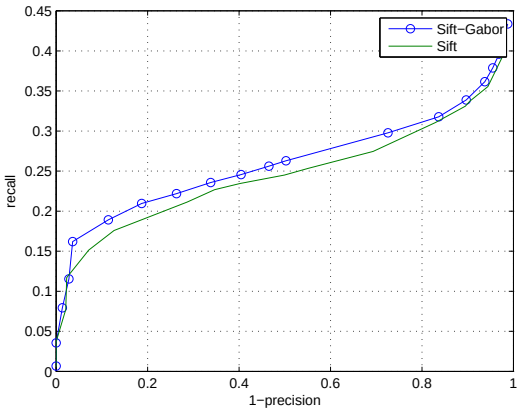
In order to evaluate quantitatively the improvement of our descriptor over the original SIFT descriptor, in every experiment we compute the difference in recall rate for a fixed precision value of 0.5. We see in Table 4.2 that our method for computing SIFT local descriptor improves SIFT distinctiveness for all the matching experiments. It is also important to note that the improvement attained by our descriptor depends on: (i) type of detector and (ii) matching criterion. In the case of detectors, Hessian detectors have a improvement greater than Harris detectors for every matching criteria. Also the improvement depends highly on the matching criterion, as recall improvement in the threshold-based method is about 10 times larger than the improvement in the nearest-neighbor methods. This difference is related to the difficulty of improving the performance of the nearest-neighbor methods, because it demands a high precision rate with very few correspondences.

	Harr	Hess	Struc	Text	Total
Threshold	2.7	4.3	3.7	2.3	3
NN	0.36	0.75	0.59	0.56	0.54
NN ratio	0.23	1.33	0.5	1.02	0.68

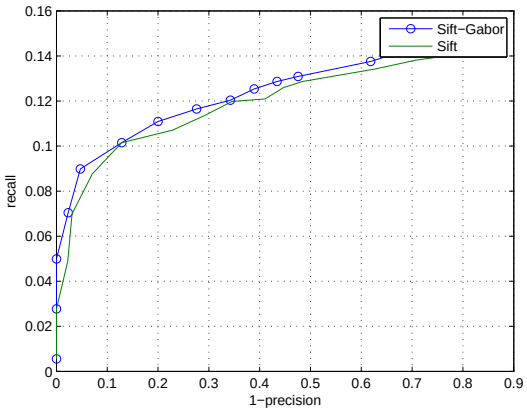
Table 4.2: Mean value of the recall difference (%) between our SIFT descriptor and original SIFT [70], at $precision = 0.5$

4.4 Conclusions

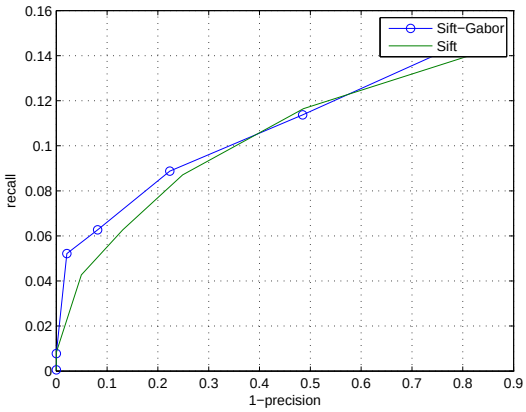
In this chapter we have presented a modification of SIFT descriptor based on odd Gabor filters as smooth derivative filters. The modification proposed computes the first order image derivatives using odd Gabor filters as convolution kernels. The filters' parameters are selected by maximizing the filter response at locations with high image gradient. To evaluate the



(a) threshold-based matching



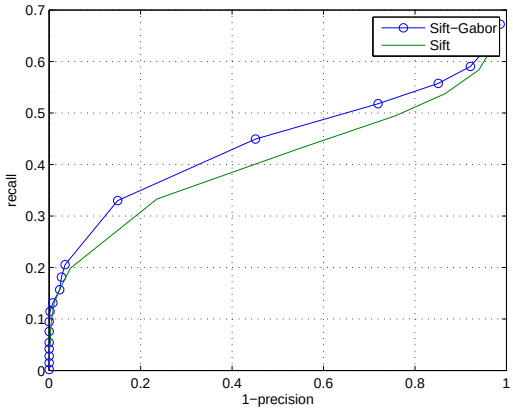
(b) nearest neighbor matching



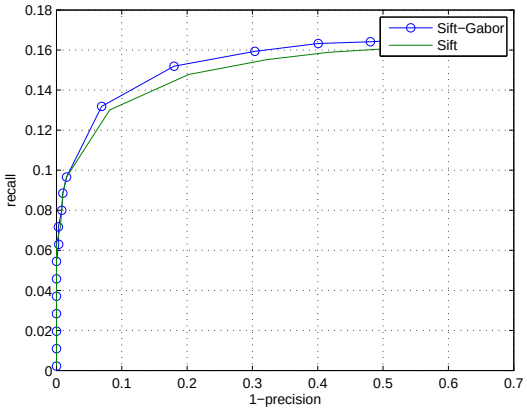
(c) nearest neighbor ratio matching

(d)

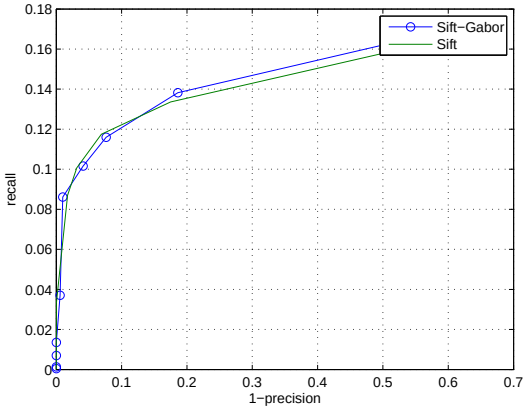
Figure 4.6: *recall* vs. $1 - precision$ curves of Harris-affine regions matched using textured images in Figure 4.6(d), related by a viewpoint transformation.



(a) threshold-based matching



(b) nearest neighbor matching



(c) nearest neighbor ratio matching

(d)

Figure 4.7: *recall* vs. 1−*precision* curves of Hessian-laplace regions matched using structured images in Figure 4.7(d), related by a scale + rotation transformation.

performance of our descriptor we use the Mikolajczyk and Schmid framework [83], computing *recall* vs. $1-precision$ curves of image regions matched by local descriptors. Our descriptor improves the SIFT distinctiveness in average.

The results of the image region matching experiments show that distinctiveness improvement is highly dependent on: (i) the matching criterion and (ii) the interest point detector. We obtain the best improvement results using the threshold-based matching criterion and Hessian-based interest point detectors.

We show experimentally that our descriptor proposal improves the performance in the single image region matching task. In the next chapter we will evaluate its performance in a full object recognition scenario.

Chapter 5

Object recognition experiments

In this chapter we use state-of-the-art recognition methods based on object components to validate, in a full recognition system, the component selection and matching methods proposed in the previous chapters [91]. We do not aim to propose new methods for object recognition, but perform comparisons between different types of object component detectors and descriptors.

Component-based object models have been shown to perform well in cluttered scenes and enjoy nice properties such as invariance to rigid transformations and robustness to partial occlusion and non-rigid transformations. Objects are represented as the collection of their parts [70, 62, 47, 1, 114, 4, 29, 30] and each part is modeled by a local descriptor. Then, the overall object model is built either by the concatenation of the appearance of each part (i.e. appearance-only model) [96, 109] or considering the pose between components (i.e. shape-and-appearance model) [33, 47]. The experiments conducted in this chapter consider both types of models.

To represent an image category, the appearance-only models select a set of descriptors from training images containing objects in that category. Often, because it is a tedious process, no object segmentation is performed in the training images. The model uses both foreground (object) and background data and can be seen as an "object+context" representation. We adopt the model presented in [109] that considers object categorization as a two class problem (object samples vs. no object samples). The number of local descriptors that represent the category is a parameter of the learning algorithm. We consider nine different object classes: airplanes, cars (lateral view), cars (rear view), camels, faces, guitars, leaves, leopards, and motorbikes. *Google things* is used as the no-object (background) category (negative examples). We employ AdaBoost and SVM learning algorithms to estimate the class models and perform recognition. The local descriptors used in these tests include SIFT [70], HMAX [103], and the SIFT improvement introduced in Chapter 4.

Shape-and-appearance models consider both the geometric configuration of the components and their appearances. In our tests we will use the pictorial structure model of [47]. It is a probabilistic approach that models the objects using a star-like graph model whose vertices represent object components and edges represent the relative position between components. This model allows object translations and is robust to small scalings, but it is not fully invariant to object rotations and scalings. We apply the pictorial structure in a face detection task to evaluate the techniques previously presented in this thesis: (i) the top-down saliency model, (ii) the adaptive Gabor filter bank, and (iii) the Gabor-based SIFT descriptor.

Additionally, we compare the adaptive Gabor descriptors proposed in this thesis against the state-of-the-art, the HMAX [103] features. We compare the result of the recognition procedures, including not only the recognition performances, but also the in the computational complexity, which may be a factor to take into account when deciding on a recognition method in particular applications.

We start with a brief review of the state-of-the-art approaches for component-based object recognition.

5.1 Component-based object models

To select appropriate object models for the experiments we review briefly the state-of-the art literature and evaluate qualitatively the different approaches in the field.

Early approaches to object recognition model the object by its contours [71, 48, 104]. These approaches are able to cope with image affine transformations and have been shown to be computationally efficient. However, these approaches have two serious caveats: (i) they assume that contours of objects can be reliably found in the images, which is not often the case in natural images, and (ii) since they rely on the boundaries, they neglect important information contained in the object's interior.

Instead, most of the recent approaches have adopted a component-based approach [10, 121, 70, 62, 47, 1, 114, 4, 29, 30]. The appearance of each component can be combined in two distinct ways: (i) disregarding geometric relations between components (i.e. appearance-only) and (ii) using pose between object components (i.e. shape-and-appearance). In both cases, most of the works propose probabilistic approaches to combine component appearances and build the object model. Probabilistic approaches have been preferred over other techniques due to their ability to compute a confidence value of the object detections and the availability of machine learning methods using such approaches.

5.1.1 Appearance-only approaches

Appearance-only models combine information of a large number of local descriptors in a bag-of-features approach and have been shown to be robust to occlusion and other noise sources. Additionally, this approach allows to use both foreground (object) and background (context) data to model each object category. Since these methods do not consider the pose between parts, it is difficult to obtain a precise estimate of their location in the images. The most remarkable examples of the appearance-only approach are:

- "Bag of keypoints" [19]. A bag of keypoints corresponds to a cluster in the local descriptor space. The object model consists of a histogram that counts the number of occurrences of clusters for a given class. The main contribution of this work is the experimental demonstration of visual categorization using appearance only.

- Boosting of local descriptors [96, 97]. Opelt et al. present a new version of boosting in which the weak classifier is replaced by a weak-hypotheses-finder. The weighted sum of all weak-hypotheses provides the final classification. The main contribution of this work is a new boosting algorithm that uses different kinds of local descriptors to classify objects.

- The kernel recipe to apply Support Vector Machines (SVMs) with local descriptors [119]. Wallraven et al. propose kernels that guarantee the linearity of the SVM classification function (Mercer kernels). The main contribution of this work is the definition of kernels to apply the linear SVM in non-linear feature spaces.

- Cortex-like local descriptors [109]. The main contributions of this work are: (i) a new general framework for object recognition, which is highly motivated by biology, and (ii) the versatility of the different levels of the hierarchy to perform a wide range of recognition tasks such as scene understanding, multi-class categorization, and single-object recognition.

5.1.2 Shape-and-appearance approaches

Shape-and-appearance models consider both the geometric configuration of the components and their appearances. In its original form [33], the model consists of a set of templates (i.e. parts, appearance, local descriptors) arranged in some geometric configuration (i.e. structure). Object deformations can be modeled as a series of springs connecting the individual parts. The goal of the model is to minimize a cost function with two terms: parts matching and deformation. Several works have adopted this idea, proposing various alternatives for

both terms of the cost function. In the following, we review the most remarkable works using this approach:

- Affine invariant object detection and location [70] based on SIFT features [69]. An initial interest point set is provided by local maxima in space and scale of the Difference of Gaussians (DoG) operator. Appearance is represented by the histogram of image gradient orientations at interest point neighborhoods. Shape is modeled by the interest point locations in the training image. The main contribution of this work is the real-time and very robust performance in matching objects, mainly in the first two main stages, the interest point detection and local appearance computation.

- Weakly supervised scale invariant object recognition [29]. Fergus et al. revisited Fischler and Elschlager's model [33], proposing a model that considers jointly the location, scale, and appearance of every object component as parameters to estimate. Given a training set with labeled images (object/no object), the parameters are learnt in an unsupervised manner, using the expectation maximization (EM) algorithm. The main contributions of this work are: (i) unsupervised and joint learning of appearance, location and scale parameters, and (ii) the addition of component scale to the object model.

- Efficient learning and exhaustive recognition [30]. Later Fergus et.al. explored several variations of the constellation model in order to address previous short-comings: (i) the joint nature of shape model results in an exponential explosion in computational cost, (ii) good performance is highly dependent on the interest point detection phase, and (iii) the model has many parameters, so the number of training images must be large. The main contribution of this work is the reduction of computational complexity that allows to: (i) model objects with several (more than six) components, (ii) perform efficient learning in terms of computation time, and (iii) perform exhaustive recognition by removing the interest point detection stage.

- Pictorial structures [47]. The main contribution of this work is the reduction of complexity when matching the model by: (i) using a star-like graph (i.e. tree) instead of an unconstrained graph, and (ii) using the generalized distance transform [105] to efficiently compute the model probability.

- Weakly supervised learning of part-based spatial models [17]. This work is a generalization of pictorial structures [47] in two aspects: (i) the star-like graph model is generalized to a k-fan model, a graph with a central clique of k reference nodes, with the remaining nodes connected to all k reference nodes, but to none of the non-reference

nodes, and (ii) both the appearance model and shape model are learnt jointly in a weakly supervised manner (object label, but no segmentation). The main contribution of this work is the proposal of a weakly supervised learning of an object model, computing appearance and shape jointly in a graph model.

- Multiple object detection [79]. Objects are modeled by a joint distribution of shape and appearance, computed in a hierarchical manner. The contributions of this work are: (i) the capability of detecting multiple object classes simultaneously, and (ii) the complexity and run times are improved compared to other object recognition approaches.

- Sharing features for multi-class and multi-view object detection [114, 115]. This work shows that sharing simple features (image patches, binary spatial masks) across classes attains very good recognition rates. The main contribution of this work is the reduction of the number of features while maintaining very high recognition rates in multi-class problems.

5.1.3 Qualitative comparison of object models

Appearance-only methods have shown very good recognition rates, even though the poses between object components are not utilized. Their robustness to occlusions and non-rigid transformations allows their application in cluttered images with objects in several configurations. The main drawback of appearance-only methods is the difficulty of locating objects. However, in particular object categories it is possible to give a crude estimate of the object location, provided by a bounding box that separates object from background. Another issue to consider is the amount of foreground (object) and background data contained in the model. Learning object models in unsegmented images can lead to a "background model" instead of "object model," as reported by Opelt et. al. [97]. However, this problem is common to all methods that do not perform a pre-segmentation of the object parts in the training set.

Amongst all appearance-only models, the Cortex-like mechanisms for object recognition [109] have shown very good performance and versatility in several kinds of visual tasks. Additionally, they use a dense Gabor filter-based representation (HMAX) to compute local descriptors, which will allow us to compare the performance of dense (large size) Gabor filter-based local descriptors (HMAX) against small size histogram-based descriptors (SIFT) in object recognition tasks. Thus, we choose the appearance-only object model of Serre et al. [109]. Within this context, we will compare the following descriptors: the SIFT local descriptor [70], the proposed version of SIFT using Gabor filters (Chapter 4), and the HMAX descriptor [103].

Regarding shape-and-appearance models, there is a much wider range of works and distinct approaches, thus making a qualitative comparison between methods harder. Initial approaches [63, 10, 121] used appearances only as a mean to reduce the complexity during the shape matching procedure. Subsequent approaches explicitly included appearance in the model [29, 30], but resulted in an increased complexity on parameter learning, implying a huge number of training samples. Recent works tackle these drawbacks by proposing efficient learning and matching probabilistic models such as the "star" graph [30, 47] and the generalization of the "star" graph to the k-fan model [17]. A very recent work [79] is able to estimate jointly the appearance and shape distribution parameters and locate multiple instances of several classes in one image. Amongst the shape-and-appearance models reviewed, the pictorial structure [47] integrates several state-of-the-art properties: (i) the parameters of appearance and shape probability distributions are estimated jointly, (ii) it is very fast in computational terms, and (iii) it handles partial occlusions of the object. Thus, we choose the Huttenlocher and Felzenszwalb pictorial structure [47] to assess the models presented in this thesis: (i) the top-down saliency model to make a pre-selection of every face component, (ii) the adaptive Gabor filter-bank to represent components, and (iii) the Gabor-based SIFT descriptor.

We proceed now with a more thorough explanation of the selected methods, followed by the description of the experimental setup.

5.2 Appearance-only object recognition

Due to the success of the HMAX approach [103] in appearance-only object recognition in clutter, as well as scene understanding [109], we will adopt a similar architecture for our tests. First, we will briefly revise the method proposed in [109], explained in detail in Chapter 3. Then, we will explain how this architecture can be adapted to perform not only with the HMAX features but also with other types of local descriptors. Finally, we will present results of the application of such architecture in a object recognition problem and compare the performance of the employed descriptors, namely the original HMAX and SIFT, our SIFT-Gabor descriptor, and a baseline cross-correlation method.

The HMAX appearance-only model represents an object class by a large number of randomly extracted patches. The first two steps of the HMAX procedure (S1 and C1 maps) compute, for each pixel in the images, a vector containing the local maximum in adjacent scales of the Gabor filters responses for the different orientations. Therefore, they provide the representation with local scale robustness. The third step selects C1 patches that contain the object or class representation, using images in a training set. The collection of all C1

patches extracted will provide an appearance-only representation of the object class.

In [109] the selected C1 patches are obtained from a large set of randomly selected points from the class images. The rationale is that, from unsegmented images, is is not possible to decide *a priori* where to obtain points in the objects of interest. Only selecting a large number of points, we will have a reasonable likelihood of selecting points in the object region. Obviously, the background will also be represented, but if the image set is large, it is likely that the background is very different in the data set images and constancy of the object appearance will bias the representation to include more object related information. Anyway, even when the background does not change significantly between images (e.g. airplanes, cars), the data set will provide contextual information that is also useful in the appearance-only object recognition. In our experiments we will also test selections at bottom-up interest points computed at DoG local maxima in scale–space.

The remaining steps (S2 and C2 maps) comprise the multi-scale and multi-orientation matching steps. Every C1 patch in the collection that represents the object, is matched in the entire new C1 image, retaining the strength of the most similar point in the image to the C1 patch under consideration. The maximum similarity value of each patch is collected into a vector, that is used to train a binary classifier with positive and negative examples of the object class.

5.2.1 Appearance-only object model

The HMAX model described above employs a particular type of filter-based dense descriptors to represent the appearance of an object class. In this section we want to benchmark the performance of several different types of descriptors, not only the HMAX but also the original SIFT, our SIFT-Gabor model, and the normalized cross-correlation. One of the peculiarities of the HMAX recognition architecture is that, instead of directly using the descriptors of the class in a supervised classification framework, it uses vectors that already express some degree of match between an image and the object class to recognize. Given the success of this approach in appearance-only recognition, we adopt this idea and adapt the HMAX recognition architecture to cope with different types of descriptors in a unifying framework. We also consider an additional step of initial interest point selection oriented in a bottom-up fashion, introducing an attentional mechanism to avoid performing the computations in the whole image and thus reducing the computational cost.

The following lines describe the steps of the training methodology:

1. Select M interest point locations from the training set images $\{I_1, \ldots, I_t, \ldots, I_T\}$. In [109], all points are processed (full sampling). Additionally, we test interest point

selection with DoG maxima.

2. Compute local descriptors at the interest point locations: $\mathbf{u}_i$, $i = 1, \ldots, M$.

3. From the considered interest points, randomly pick N of them to model the positive and negative class samples. The selected local descriptors are denoted by $\mathbf{u}_{s_1}, \ldots, \mathbf{u}_{s_n}, \ldots, \mathbf{u}_{s_N}$, and constitute the appearance–only object class representation. This corresponds to step 3 in the HMAX model.

4. Compute the *class–similarity* feature vector $\mathbf{V}_t = [v_1, \ldots, v_n, \ldots, v_N]$ for each image in the training set. Pick the descriptors $\mathbf{u}_i$ that belong to image I_t and compute the similarity v_n of the descriptor $\mathbf{u}_{s_n}$

$$v_n = \begin{cases} \min_i \|\mathbf{u}_{s_n} - \mathbf{u}_i\|^2 \quad i = 1, \ldots, M \wedge i \neq s_n, \mathbf{u}_i \in I_t & \text{SIFT} \\ \max_i \max_{b,\theta} \exp(-\gamma \|\mathbf{u}_{s_n} - \mathbf{u}_i\|^2) \quad i = 1, \ldots, M \wedge i \neq s_n, \mathbf{u}_i \in I_t & \text{HMAX} \end{cases} \tag{5.1}$$

   This corresponds to steps 4 and 5 in the HMAX model.

5. Use similarity vectors $\mathbf{V}_1, \ldots, \mathbf{V}_t, \ldots, \mathbf{V}_T$ with their respective label $c = \{0, 1\}$ in the learning algorithm.

If we use C1 features as descriptors in step 2 and a full image sampling in step 1, this methodology is equivalent to the original HMAX [103] and the class similarity feature vector $\mathbf{V}_t$ correspond to the C2 features of Serre et al. [109].

After learning the object model, the steps to detect an instance of the object category in a new image are as follows:

1. Select J interest point locations.

2. Compute local descriptors in the new image $\mathbf{u}_j, j = 1, \ldots, J$ at interest point locations.

3. Create *class–similarity* feature vector $\mathbf{V} = [v_1, \ldots, v_n, \ldots, v_N]$ by matching each class model point descriptor $\mathbf{u}_{s_n}$ against all image descriptors $\mathbf{u}_j$.

$$v_n = \begin{cases} \min_j \|\mathbf{u}_{s_n} - \mathbf{u}_j\|^2 \quad j = 1, \ldots, J & \text{SIFT} \\ \max_j \max_{b,\theta} \exp(-\gamma \|\mathbf{u}_{s_n} - \mathbf{u}_j\|^2) \quad j = 1, \ldots, J & \text{HMAX} \end{cases} \tag{5.2}$$

4. Classify $\mathbf{V}$ as object or background image, with a binary classifier.

The classifier will operate on vectors of dimension N. Similarly to Serre et.al. [109], in our tests we will use SVM and AdaBoost binary classifiers.

Figure 5.1: Selected images from each category

5.2.2 Experiments with the appearance-only model

Experiments are performed with a set of image categories provided by Caltech [1] : *airplanes side–view, cars side-view, cars rear–view, camels, faces, guitars, leaves, leopards*, and *motorbikes side–view*, plus the *Google things* dataset [30]. For every category, we use the *Google things* as negative samples. For training we randomly choose 100 images from the positive set and other 100 from the negative set. Figure 5.1 shows some sample images from each category. For all experiments, images have a fixed height of 140 pixels, keeping the original image aspect ratio and converted to gray-scale format.

The setup of every category recognition experiment requires the selection of: (i) a local descriptor type, (ii) the number of descriptors, and (iii) the learning algorithm. Each experiment is repeated 10 times using different training and testing sets and the evaluation criterion is the classification performance at the equilibrium point of the ROC curve (i.e. when the false positive rate is equal to the miss rate), along with the confidence interval (at 95%). The local descriptors used in this test are:

- original HMAX, as explained in Section 3.2.1.
- HMAX computed at DoG. The final three steps of the original HMAX are computed at DoG interest points. Thus, patch extraction and matching is done at DoG points, instead of the random procedure of [109].
- original SIFT, as proposed by Lowe [70].
- SIFT non-rotation-invariant (NRI). The orientation normalization procedure is removed from the original SIFT descriptor.
- SIFT-Gabor. The modification of SIFT descriptor introduced in Chapter 4.

[1] Datasets are available at: http://www.robots.ox.ac.uk/~vgg/data3.html

- SIFT-Gabor NRI. The modification of SIFT descriptor, removing the orientation normalization.

- Cross-Corr. Normalized cross-correlation

We vary the number of local descriptors that represent an object category, $N = \{5, 10, 25, 50, 100, 250, 500\}$. In order to evaluate the influence of the learning algorithm, we utilize two classifiers: SVM [98] with a linear kernel[2] and AdaBoost [37] with decision stumps.

Support Vector Machines										
	Airplane		Camel		Car-side		Car-rear			
TF/NF	10	500	10	500	10	500	10	500		
HMAX	**87.3**, 2.2	**95.9**, 1.0	**70.4**, 3.1	**84.3**, 2.2	87.9, 4.0	98.1, 1.5	**93.0**, 1.1	**97.7**, 0.8		
HMAX-DoG	80.3, 2.6	94.9, 0.8	70.2, 3.9	83.9, 1.4	**88.9**, 3.8	**99.5**, 0.9	86.6, 1.8	97.0, 0.7		
SIFT-Gabor-NRI	70.0, 4.6	84.2, 1.8	59.9, 2.7	66.6, 0.8	73.9, 3.8	78.3, 2.8	71.9, 1.9	84.8, 1.1		
SIFT-Gabor	69.6, 4.1	82.8, 2.4	55.4, 3.1	65.3, 2.8	66.7, 5.3	74.9, 4.0	63.4, 2.9	79.2, 2.4		
SIFT-NRI	70.2, 2.5	84.9, 2.1	57.5, 2.5	66.7, 1.6	73.4, 2.2	78.3, 3.8	74.1, 2.0	84.0, 2.4		
SIFT	67.5, 4.9	82.9, 1.4	57.7, 4.8	64.7, 3.1	68.1, 3.8	72.5, 4.3	63.0, 3.1	78.8, 2.0		
Cross-corr	67.5, 3.7	81.4, 1.2	55.2, 4.5	64.5, 2.3	71.5, 2.9	74.9, 3.1	65.1, 3.3	78.9, 1.2		
Faces		Guitar		Leaves		Leopard		Motorbike		
10	500	10	500	10	500	10	500	10	500	
79.8, 3.4	**96.6**, 0.7	**87.1**, 4.0	**96.7**, 1.1	**88.6**, 3.1	**98.3**, 0.6	**81.4**, 3.4	**95.7**, 0.9	**81.9**, 3.4	93.7, 0.9	
82.7, 1.8	96, 0.6	82.9, 4.0	95.9, 0.8	84.6, 2.0	**98.3**, 0.9	70.9, 3.9	94.2, 1.3	81.6, 2.3	**94.7**, 0.7	
81.1, 3.3	87.1, 1.9	70.0, 3.6	82.3, 2.4	78.3, 1.8	85.0, 2.4	64.7, 2.9	74.7, 1.6	64.7, 2.8	75.4, 2.4	
75.2, 2.3	83.4, 2.3	69.2, 2.3	73.0, 2.7	72.5, 3.4	83.6, 2.2	66.5, 2.3	75.5, 2.2	61.5, 2.4	73.3, 1.7	
80.1, 3.5	86.1, 1.8	69.2, 3.8	80.9, 1.4	77.2, 2.3	85.8, 1.7	65.5, 2.7	78.2, 1.2	64.9, 2.4	73.9, 2.5	
72.2, 3.1	83.8, 2.5	71.3, 4.6	75.7, 2.2	71.2, 3.5	83.8, 2.4	68.0, 4.1	78.0, 1.5	58.4, 2.1	71.3, 1.8	
72.3, 1.9	82.4, 1.9	72.6, 3.0	78.6, 2.4	67.4, 3.5	82.1, 1.5	60.8, 1.8	67.6, 2.4	61.5, 2.2	70.4, 1.4	

Table 5.1: Results for the SVM learning algorithm. (TF: type of feature, NF: number of features). For each experiment, the mean value and standard deviation of the EEP point of the ROC curve for 10 repetitions. For every object category and number of descriptor, the best result is in bold face.

The classification results of the SVM learning algorithm are shown in Figure 5.2, Table 5.1 and Appendix C. We observe in Figure 5.2 examples of the performance evolution as a function of the number of local descriptors N, in the case of rigid (*airplanes*) and articulated (*leopards*) objects. For the remaining classes, the corresponding plots are in Appendix C. To illustrate a global view of the results, we show in Table 5.1 a partial view of all SVM experiments with $N = 10$ and $N = 500$. The best average performances are obtained by the original HMAX, followed by HMAX computed at DoG (HMAX-DoG), SIFT-Gabor-NRI, SIFT-NRI, SIFT-Gabor, SIFT, and CrossCorr.

In the case of the AdaBoost algorithm, we show in Table 5.2, Figure 5.3, and Appendix C, the correspondent results. The ranking of classifiers of AdaBoost is equal to SVM, but in average, recognition rates of AdaBoost are below SVM.

[2]Implementation provided by *libsvm*[13]

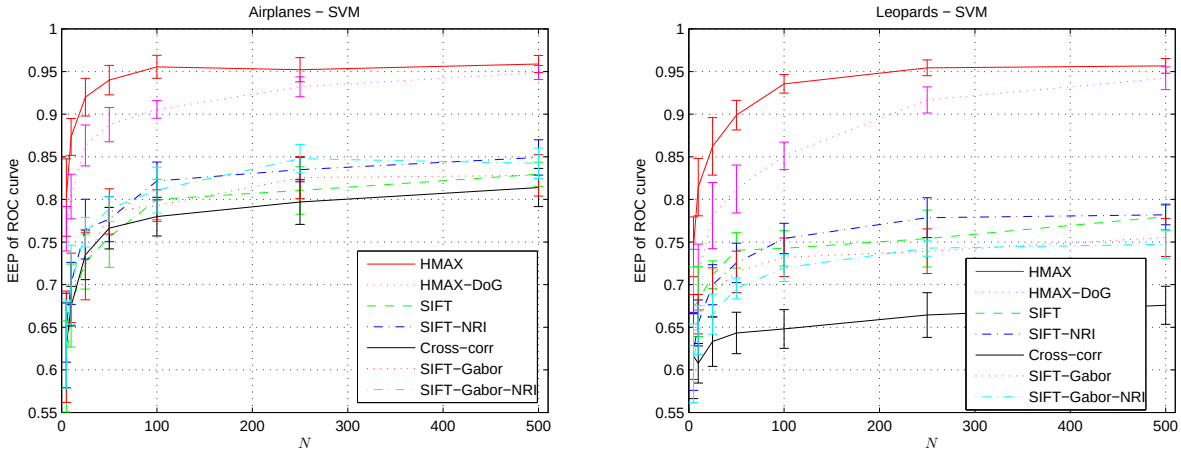


Figure 5.2: Comparison of performance depending on the type and number of features representing the images. The classifier used is SVM.

AdaBoost										
	Airplane		Camel		Car-side		Car-rear			
TF/NF	10	500	10	500	10	500	10	500		
HMAX	**81**, 0.7	**94.3**, 1.1	**67.7**, 3.3	**83.1**, 1.0	84.1, 2.8	94.2, 2.0	**90.1**, 5.1	**98.3**, 0.7		
HMAX-DoG	77.8, 3.6	93.2, 1.3	63.9, 4.5	79.1, 1.8	**85.5**, 5.5	**96.6**, 1.3	74.1, 15.7	96.4, 1.3		
SIFT-Gabor-NRI	71.9, 2.1	82.9, 1.6	60.1, 2.5	64.9, 2.4	73.9, 2.5	69.1, 2.2	72.6, 3.2	83.5, 2.1		
SIFT-Gabor	71.4, 3.3	82.7, 1.4	57.5, 3.4	63.3, 1.8	67.6, 4.9	68.6, 2.8	62.8, 3.5	77.0, 2.0		
SIFT-NRI	72.0, 2.0	82.6, 1.3	59.3, 2.0	64.9, 1.6	73.4, 3.0	69.6, 2.8	74.6, 2.3	83.5, 2.0		
SIFT	69.1, 2.7	81.5, 1.8	56.1, 3.3	64.2, 2.1	67.1, 4.0	65.7, 4.4	61.4, 3.4	77.8, 1.9		
Cross-corr	66.9, 3.6	78.1, 1.7	53.1, 2.6	60.6, 2.4	70.5, 3.7	77.3, 3.7	62.4, 2.6	75.7, 2.6		
Faces		Guitar		Leaves		Leopard		Motorbike		
10	500	10	500	10	500	10	500	10	500	
77.1, 4.7	94.9, 1.1	**83.7**, 7.1	**96.6**, 1.0	**83.1**, 6.2	**97.7**, 0.7	**76.8**, 2.8	**85.6**, 1.1	74.7, 4.8	92.0, 1.7	
74.4, 6.1	**95.7**, 1.2	78.0, 6.9	92.7, 1.5	76.0, 4.6	97.0, 0.9	70.2, 5.5	83.1, 2.0	**75.2**, 3.7	**93.4**, 0.9	
80.9, 2.9	86.1, 2.1	72.4, 3.2	81.0, 2.5	77.6, 1.7	82.5, 1.4	63.5, 3.7	72.1, 1.7	66.4, 2.6	74.3, 1.9	
75.7, 2.9	84.0, 1.9	73.1, 1.7	72.4, 2.1	71.9, 2.4	80.5, 2.2	66.0, 1.8	70.7, 2.3	61.4, 2.7	73.3, 1.8	
80.2, 3.0	86.1, 2.5	73.6, 4.3	80.3, 2.2	77.4, 1.5	83.3, 1.4	67.6, 2.7	74.9, 1.4	64.6, 3.2	74.6, 2.2	
72.9, 2.6	82.1, 1.6	73.5, 2.4	73.4, 2.4	70.0, 2.5	79.4, 2.4	68.8, 2.4	73.5, 2.7	60.0, 1.9	72.8, 1.4	
71.7, 3.3	81.1, 1.6	75.5, 2.8	77.2, 2.8	66.3, 4.7	78.0, 1.9	60.2, 1.3	62.4, 2.7	59.6, 3.2	68.7, 1.1	

Table 5.2: Results for the AdaBoost learning algorithm. (TF: type of feature, NF: number of features). For each experiment, the mean value and standard deviation of the EEP point of the ROC curve for 10 repetitions. For every object category and number of descriptor, the best result is in bold face.

Local descriptors can be clustered in three groups using the average performance for both SVM and AdaBoost: HMAX, SIFT-NRI, and SIFT. HMAX descriptors have the best performance, followed by SIFT-NRI descriptors and SIFT descriptors. The separation between the groups depends on the learning algorithm, in the case of SVM the distance between groups is larger than AdaBoost. AdaBoost groups are closer to each other and for some categories (*motorbikes*, *airplanes*, and *leopards*) all descriptors have practically the same performance.

We can observe in the plots that HMAX descriptors are clearly ahead in recognition per-

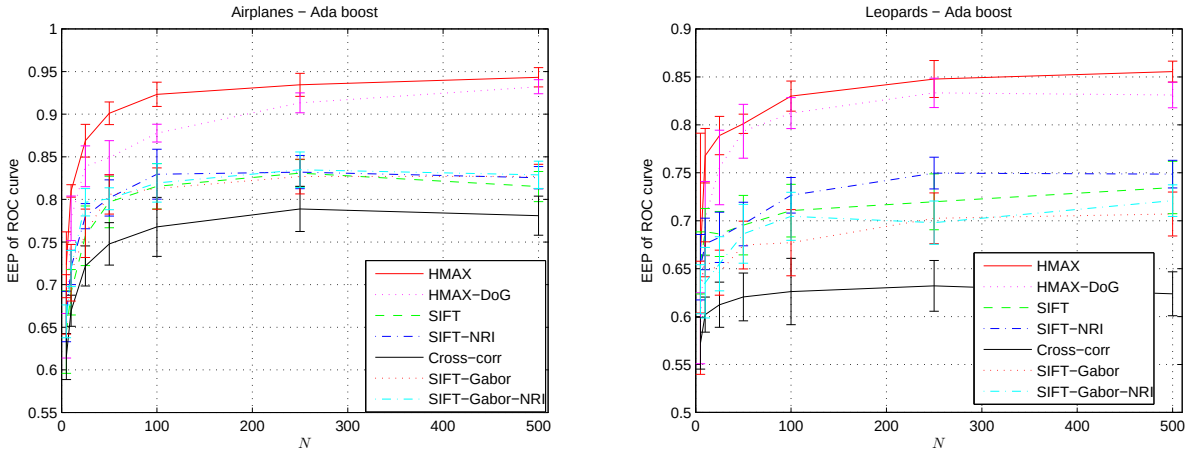


Figure 5.3: Comparison of performance depending on the type and number of features representing the images. The classifier used is Ada Boost.

formance with respect to the SIFT based descriptor. The very expensive matching procedure of HMAX is one of the reasons for the superior performance over the SIFT descriptors. Considering an object component with size $M \times N$ and descriptor size S, the complexity of the SIFT matching procedure is $O(S)$, while in the case of HMAX (with the addition of bands B and orientations T) the matching complexity is $O(M \times N \times B \times T \times S)$. This very large difference in the matching complexity is reflected in the recognition rate difference. Additionally, it may be the case that the adopted recognition architecture, inspired in the original HMAX work, in fact gives an advantage to HMAX descriptors. Anyway, it is not surprising that a methodology using a dense Gabor filter-based representation and a biologically motivated recognition architecture based on a multi-scale multi-orientation matching procedure provides state-of-the-art results. In fact, this supports the idea of utilizing biological inspiration in addressing the challenging problem of object recognition.

Considering the comparison between Gabor-SIFT and SIFT, we note the better recognition rates of Gabor-SIFT. These results confirm the performance improvement of the Gabor-based SIFT descriptor presented in Chapter 4, now applied in a full recognition task.

5.3 Shape–and–appearance object recognition

Pictorial structures represent objects by a set of C components (i.e. parts) and their relative positions. This information is encoded into a graph $G = (V, E)$ where $V = \{v_1, \ldots, v_c, \ldots, v_C\}$ is the set of vertices modeling the appearance of the different object components and $E = \{e_1, \ldots, e_k, \ldots, e_K\}$ is a set of edges representing the object shape, i.e. the geometric relationships between some of the vertices. The connectivity of the graph will depend on the

particular problem – full connectivity is not always necessary.

For our tests we will consider rigid, or almost rigid, objects. In this case an efficient graph structure is given by a star model, where one of the nodes, denoted "landmark," is connected to all other nodes, as illustrated in Figure 5.4. For other types of objects, different graph connectivity structures may be more suited, for instance articulated objects are more appropriately represented by tree-like models [47]. In a star graph structure, the set of edges can be represented as $E = \{e_{12}, \ldots, e_{1c}, \ldots, e_{1C}\}$, where e_{1c} represents a connection between node 1 (the landmark) and node c.

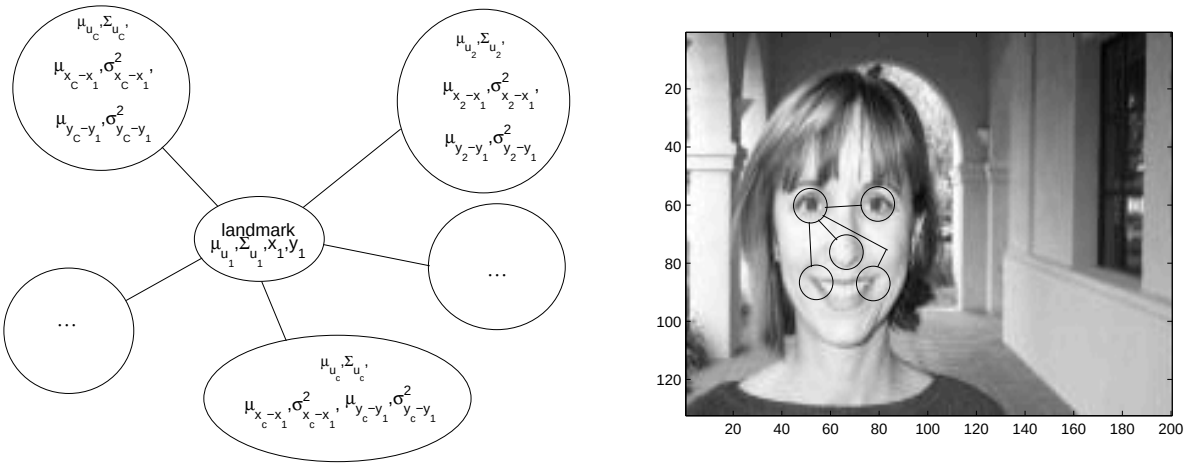


Figure 5.4: Pictorial structure model

For learning and recognition, a graph is modeled in a probabilistic framework [47] that computes the probability of an image[3] I, given an object configuration $L = \{\mathbf{l}_1, \ldots, \mathbf{l}_i, \ldots, \mathbf{l}_n\}, \mathbf{l}_i = (x_i, y_i)$ as

$$p(L|I, \theta) \propto p(I|L, \theta)p(L|\theta). \tag{5.3}$$

Nodes are represented by the appearance of object parts. This is usually expressed by the distribution of local descriptor values on the components image patches, encoded as a random vector $\mathbf{u}_c$, $c = 1, \cdots, C$. Often, a Gaussian distribution is assumed and component appearance is represented by its mean and covariance:

$$v_c = (\mu_{\mathbf{u}_c}, \Sigma_{\mathbf{u}_c}). \tag{5.4}$$

For almost rigid objects, shape is represented by the distribution of relative displacements between parts. This is usually modeled by Gaussian distributions of the node x and y

[3]set of intensity values that visually represents the object

coordinates referenced to the landmark location. Thus, the shape model can be written as:

$$e_{1c} = (\mu_{x_c - x_1}, \sigma^2_{x_c - x_1}, \mu_{y_c - y_1}, \sigma^2_{y_c - y_1}),\ c = 2, \ldots, C; e_{1c} \in E, \tag{5.5}$$

Both the appearance and shape models can be learned from a set of training data. In Figure 5.2 we show a graphical representation of the model.

To detect objects in new images, we adopt the probabilistic framework of [47]. Objects are detected by computing the posterior likelihood of object configurations $L = \{\mathbf{l}_1, \ldots, \mathbf{l}_c, \ldots, \mathbf{l}_n\}$, $\mathbf{l}_c = (x_c, y_c)$, given the model $\theta = (V, E)$ and the image data I. We consider that the likelihood of an object part at a certain image location $\mathbf{l}_c$, can be measured by an observation process that "matches" the local descriptor at that image location with the part's appearance model $\mathbf{u}_c$. With a Gaussian model for the appearance descriptors, the observation model is:

$$p(I|\mathbf{l}_c, \mathbf{u}_c) \propto \mathcal{N}(\mu_{\mathbf{u}_c}, \Sigma_{\mathbf{u}_c}). \tag{5.6}$$

Assuming statistical independence between the individual object part observation models, which is a good approximation when the parts do not overlap, we can write the following:

$$p(I|L, \theta) = p(I|L, V) \propto \prod_{c=1}^{C} p(I|\mathbf{l}_c, \mathbf{u}_c). \tag{5.7}$$

The prior $p(L|\theta)$ is captured by the Markov random field with edge set E, expressed by

$$p(L|\theta) = \frac{\prod_{(v_1, v_c) \in E} p(\mathbf{l}_1, \mathbf{l}_c|\theta)}{\prod_{v_c \in V} p(\mathbf{l}_c|\theta)^{\deg\, v_c - 1}} = \frac{\prod_{(v_1, v_c) \in E} p(x_1, x_c|\theta) p(y_1, y_c|\theta)}{\prod_{v_c \in V} p(\mathbf{l}_c|\theta)^{\deg\, v_c - 1}}, \tag{5.8}$$

where $\deg\, v_c$ is the degree (depth) of vertex v_c in the graph defined by E. If we do not have preferences over the location of each part, the denominator in Equation (5.8) is constant and can be discarded (it is just a normalization factor). Thus replacing Equations (5.7) and (5.8) in Equation (5.3), we obtain:

$$P(L|I, \theta) \propto \left(\prod_{c=1}^{C} p(I|\mathbf{l}_c, \mathbf{u}_c) \prod_{(v_1, v_c) \in E} p(x_1, x_c|e_{1c}) p(y_1, y_c|e_{1c}) \right). \tag{5.9}$$

Computing the negative logarithm of Equation (5.9), the MAP solution can be obtained by

the folowing minimization problem:

$$L^* = \arg\min_{L} \left(-\sum_{c=1}^{C} \log p(I|\mathbf{l}_c, \mathbf{u}_c) - \sum_{(v_1, v_c \in E)} \log p(x_1, x_c|e_{1c}) - \sum_{(v_1, v_c \in E)} \log p(y_1, y_c|e_{1c}) \right), \tag{5.10}$$

With Gaussian assumption on both the appearance and shape models, the probability density functions involved in the previous expression are:

$$p(x_1, x_c|e_{1c}) \propto \mathcal{N}(\mu_{x_c - x_1}, \sigma^2_{x_c - x_1}) \tag{5.11}$$

$$p(y_1, y_c|e_{1c}) \propto \mathcal{N}(\mu_{y_c - y_1}, \sigma^2_{y_c - y_1}) \tag{5.12}$$

$$p(I|\mathbf{l}_c, \mathbf{u}_c) \propto \mathcal{N}(\mu_{\mathbf{u}_c}, \Sigma_{\mathbf{u}_c}). \tag{5.13}$$

By solving the Equation (5.10), we will obtain the most probable object configuration L^* in a new image I.

5.3.1 Experiments with shape-and-appearance model

We aim to detect and locate faces in images using local-appearance models (adaptive Gabor filter-bank, SIFT, SIFT-Gabor, and HMAX) and the pictorial structure model. We use a subset of the Caltech faces (100 images), background (100 images) database images, and the software provided at the "ICCV'05 Short Course" [27]. In this experiment background images do not model a negative class, but they are utilized only to test the object model in images without faces. All images are subsampled to a 200x132 size. We select 10% of the face images to learn the local descriptor model ($\mu_{\mathbf{u}_i}, \Sigma_{\mathbf{u}_i}$) and the pictorial structure model ($\mu_{x_j - x_1}, \sigma^2_{x_j - x_1}, \mu_{y_j - y_1}, \sigma^2_{y_j - y_1}$), with $C = 5$ parts. The covariance matrices of the descriptors are assumed diagonal. In the case of SIFT descriptors, it is necessary to reduce from 128 to 32 dimensions in order to compute covariance matrices elements. We recognize objects in the remaining 90% of face image set and background images.

The model learning (i.e. estimation) is a supervised procedure in which the user clicks in the image to select and locate face parts. We model five parts: left and right eye, nose center, and left and right mouth corners. Model classification (i.e. recognition) computes the most probable location of the object in novel images (L^* in Equation (5.10)).

During classification we use the set-up presented in Figure 5.5. The modules are: (i) Salient point detection, (ii) local descriptor computation, and (iii) face recognition. The salient point detection module is equal to the one used in Chapter 2 (Section 2.7.2), composed by: (i) initial interest point selection, provided by the local maxima of the Laplacian of Gaussian response applied at several scales in the image, and (ii) selection of candidates for

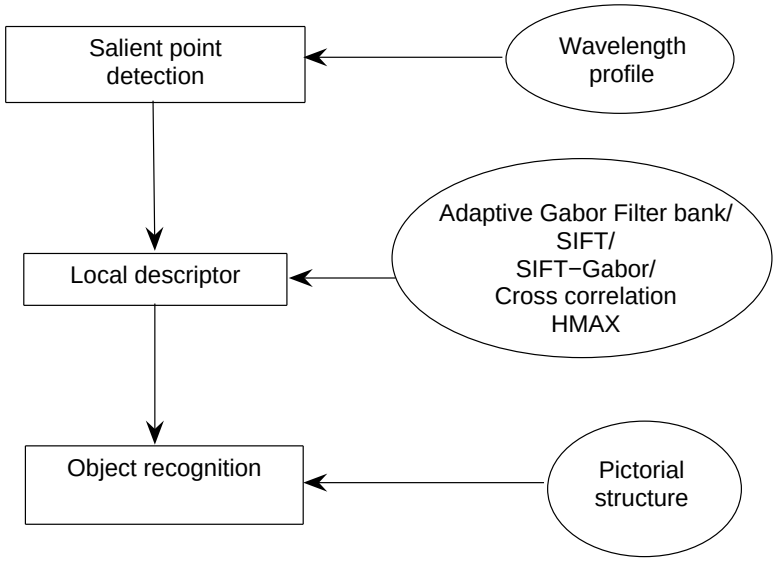


Figure 5.5: Set-up of the face recognition experiments

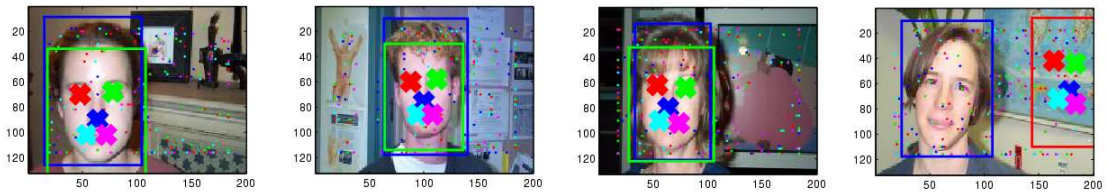


Figure 5.6: Face detection samples: 3 hits and 1 miss (right).

facial components using the saliency model SM_c, based on the wavelength signature.

The second module of the set-up is the local descriptor computation. We compute several object component descriptors in order to compare the performances of: (i) the adaptive Gabor filter-based presented in Chapter 3, (ii) the original SIFT descriptor, (iii) the Gabor-based SIFT descriptor presented in Chapter 4, (iv) HMAX, and (v) cross correlation. The last module is the object model recognition, using the pictorial structure explained in Section 5.3.

Evaluation criteria comprises object detection and location. For object detection we compute the Receiver Operator Characteristic (ROC) curve, varying the threshold in L^* value. For object location we compute precision *vs.* recall curve, varying the ratio between the intersection of ground truth and detected bounding boxes and the union of the bounding boxes. The quantitative criterion for comparing two descriptors is the equal-error-point (EEP) and area of the ROC curve (detection) and precision *vs.* recall curve (location).

In Table 5.3 we see the EEP results for face detection and localization. We also consider a variant of the set-up in Figure 5.5, by removing the top-down saliency model, to see the influence in the recognition rate.

	ROC		Recall-precision			
	EEP (%)	Area(%)	EEP (%)	Area(%)		Complexity
SIFT	82.1	91.3	79.9	82.2	saliency	$O(S)$
	86.3	93.4	80.3	81.2	no saliency	
SIFT-Gabor	83.1	91.7	80.5	84.1	saliency	$O(S)$
	85.3	94.9	81.7	83.1	no saliency	
HMAX	89	94.8	84.9	85.5	saliency	$O(MNBTS)$
	90.8	95.6	86	85.8	no saliency	
Adaptive Gabor	82.1	91.5	56.7	57.8	saliency	$O(S)$
Cross correlation	80	70.5	38.1	33.7	saliency	$O(MN)$

Table 5.3: Face recognition using pictorial structures [47]. Equal-error-point (EEP) and area of the ROC curve (detection) and precision *vs.* recall curve (location), along with the computational complexity of matching.

The very expensive matching procedure of HMAX is one of the reasons for the superior performance over the SIFT descriptors. To illustrate the difference in the complexity between descriptors, let us consider an object component with size $M \times N$ and descriptor size S. Then, the complexity of SIFT matching procedure is $O(S)$, while in the case of HMAX (with the addition of bands B and orientations T) the matching complexity is $O(M \times N \times B \times T \times S)$ and for normalized cross-correlation it is $O(M \times N)$.

Considering the matching complexity and performance, we see that SIFT has a very good balance due to the efficient matching procedure and good detection rates. By contrast, the adaptive Gabor filter descriptor of Chapter 3 has an efficient matching procedure, having a good face detection rate, but the localization rate is just above from chance. In the bottom line we see the HMAX descriptor, having good performance in detection and localization, but with a very high computational complexity to match descriptors. These results show the suitability of dense Gabor filter-based descriptors in challenging recognition problems.

Using object detection and localization criteria, we see that the top three descriptors are: (i) HMAX, (ii) SIFT-Gabor, and (iii) SIFT. These descriptors are able to recognize faces correctly in cluttered environments. Figure 5.6 shows examples of three correct detections and one wrong detection when using HMAX.

The addition of the saliency module has two effects: (i) a drop in average of 3.1% in the detection rate, and (ii) an increase of 1.13% in the localization rate. Nevertheless, the experiments with saliency reduce in average, 65% of the computations during recognition.

These results confirm the advantages of the saliency model found in Chapters 2 and 3.

5.4 Summary and conclusions

We present exhaustive experiments in component-based object recognition, using two kinds of models: (i) an appearance-only-model, and (ii) a shape-and-appearance model. The appearance-only model was utilized to detect object categories in a binary class problem, in which the objective is to decide if there is an object of the class modeled in new images. We apply the shape-and-appearance model in a face detection and localization problem. The purpose of the amount and variety of experiments done is to evaluate in object recognition in cluttered scenes the capabilities of:

- the top-down saliency model provided by the wavelength signature. This model brings efficiency and improves the precision rate, reducing up to 65% the computations of the subsequent steps of object recogntion.
- the Gabor-based SIFT descriptor, improving the recognition rates when compared to the original SIFT.
- the SIFT descriptor, having a very good balance between computational complexity during matching and good detection and localization rates of objects in cluttered data sets.
- the HMAX descriptor, being able to discriminate categories and recognize faces correctly, showing that a very dense Gabor-based component description is feasible, but at the expense of a matching procedure with very high computational complexity.

Chapter 6

Conclusions and future work

In this thesis we addressed the problem of component-based object recognition using biologically motivated Gabor filters for interest point detection and representing image neighborhoods (image descriptors). The visual cortex of the monkey brain contains layers of cortical cells resembling Gabor functions that serve as a feature extraction front–end for subsequent visual processing tasks. Similarly to Gabor functions, these cells are able to analyze low-level texture properties of the images and can be tuned for different types of structures like edges, bars and gratings, having different scales, orientations and spatial frequencies.

Most of the existing works utilizing Gabor functions do not properly exploit the full richness of their parameterization, limiting their application to the analysis of orientation and scale "degrees of freedom." Instead we have explored the whole range of parameters of the Gabor functions and show that a proper selection of their values is advantageous in four important steps of the object recognition problem: (i) the selection interest points, (ii) the computation of the intrinsic scale of image regions, (iii) the design of robust local image descriptors, and (iv) the representation of object categories. We have executed extensive tests comparing the performance of our models with other state-of-the-art methods in all stages of the object recognition architecture.

First, we proposed a coarse top-down **interest point selection method** able to introduce object related information very early in the recognition process and thus significantly reduce the overall computational cost. By contrast, most state-of-the-art object recognition methods consider top-down information only in the final recognition decision. We introduced an additional intermediate step in the recognition architecture in order to filter bottom-up candidates that are very different from the model, thus excluding them from costly subsequent recognition steps. The proposed method consists in modeling the local texture properties of object parts by means of the analysis of isotropic patterns of different wavelengths, invariant to position, scale and orientation. This is achieved by computing a wavelength profile that

collects the response of multiple Gabor filters tuned to different scales and orientations at every wavelength. We derived a new filter kernel able to compute the wavelength profile of an image region without having to actually perform multiple Gabor filter convolutions. We showed the ability of such a top-down saliency procedure to significantly reduce the computations required in object detection in cluttered scenes, with very few rejections of true positives.

Secondly, we proposed a new method to compute the **intrinsic scale** of an interest point. This is obtained by using the unique properties of the wavelength profile function used to build the top-down saliency function. The method proposed is able to compute the scaling factor between an image region and scaled versions of it, even in cases where standard methods, such as scale-normalized Laplacian of Gaussian, fail: (i) it is able to correctly compute the intrinsic scale in ridge-like structures, and (ii) it presents very low variance in its output for regions with very similar visual appearance. These advantages allow computing the intrinsic scale in a broader set of textured regions than standard methods.

Regarding the description of image neighborhoods we have considered filter-based and histogram-based methods. One of the best known state-of-the-art methods in filter-based methods is the HMAX, which serves as a baseline for comparison. To overcome the extreme computational load of HMAX or similar methods, we presented an alternative method that relies on a local descriptor vector composed of a limited number of Gabor responses computed with the selection of best frequencies (wavelength inverse) and scales. Instead of using a Gabor filter bank with a fixed set of parameters, this approach performs a data driven automatic selection of the most informative Gabor parameters. This is achieved by looking for local extrema of the Extended Information Diagram (EID) function of each object component. An additional step allows the representation of components in a way invariant to scale and orientation. We presented experimental results where we have successfully recognized object components using either: (i) the adaptive Gabor local descriptor model only or (ii) both the adaptive Gabor local descriptor model and the top-down saliency model to successfully detect and locate facial components.

We also exploited the paradigm of Gabor parameter selection for improving well-known state-of-the-art histogram-based descriptors. We presented a method to select the best scale to compute first order image derivatives in order to improve SIFT local descriptor distinctiveness. The parameter selection procedure looks for local extrema of odd Gabor filter responses, providing the best filter width to compute the image derivatives in scale-normalized regions. In order to evaluate the improvement over the original SIFT descriptor, we use the comparison framework introduced by Mikolajczyk and Schmid [83]. The results obtained have shown that the matching capabilities of SIFT are improved on average.

Finally, we evaluated the proposed approaches for *automatic selection* of Gabor filter parameters by applying them in a full object recognition setup. We evaluated their performance in comparison to other state-of-the-art approaches in two important classes of object recognition problems: appearance-only and shape-and-appearance.

In a first group of tests we applied an appearance-only object model that disregards pose between local descriptors. The model is applied to the recognition of several object categories in order to compare the performance of various local descriptors. Results from this group of tests have shown that our proposed improvement of the SIFT descriptor surpasses the original SIFT, using the recognition rate as performance criterion. In a second group of tests, we applied a shape-and-appearance object model to evaluate the performance of: (i) the top-down saliency model from the wavelength signature, (ii) the adaptive Gabor filter bank, and (iii) the Gabor-based SIFT descriptor. The results of both groups of experiments in cluttered images, presented in Chapter 5, reinforced the conclusions obtained previously in Chapters 2-4:

- The top-down saliency model with wavelength signature is able to significantly reduce the computational complexity of the object component matching, giving rise to very few rejections of the actual object components.

- The adaptive Gabor filter bank is able to match object components with a performance comparable to the HMAX and with significantly smaller computational cost. The rate of false positives may become important for complex, highly cluttered images.

- The Gabor-based SIFT descriptor outperforms the original SIFT local descriptor in terms of recognition rate, at the cost of a small addition to the computational effort.

Another important conclusion of the experimental work is related to the performance of the HMAX method in category recognition. Although this method was not proposed by us, it is a nice example of the utilization of biologically motivated principles. The inspiration by the primate brain is twofold: the use of Gabor functions as low-level feature and a matching procedure inspired in the structure of the visual cortex.

In summary, we have shown the applicability of Gabor filters in several steps of the object recognition process, from salient point detection to object component description. Results are comparable to current state-of-the-art, but three important points should be noticed and distinguished from other works:

- The early introduction of coarse appearance models in the recognition process allows important computational savings.

- Purposeful selection of the Gabor function parameters has proven to be a successful technique, overcoming fixed Gabor filter approaches.
- The promising results obtained with the application of biological principles in object recognition, in particular the performance of the HMAX method, encourage further investigations in Gabor-based approaches, which may lead to novel methods that surpass the current state-of-the-art.

6.1 Future work

One of the main results of this thesis is the derivation of a saliency function based on the scale and rotation invariance texture properties of the Gabor function. The successful results provided by the top-down saliency operator encourage the application of the wavelength signature operator in other image domains. The wavelength signature extracts the contribution of several textures, a property that is suitable to detect abnormal textures and perform dense frequency estimation in images.

The successful recognition rates provided by the HMAX-based model show that features based on Gabor responses are suitable to detect objects in images if appropriate sampling of the responses and matching procedures are designed. The matching procedure of the adaptive Gabor filter descriptors provides, however, a high number of false positives, mainly due to the local match in the scale dimension and the sparseness of the representation. Future approaches should consider both the adaptive nature of the Gabor filters presented in this thesis, along with a more dense representation, and an exhaustive matching criterion to obtain high numbers in the true positive rate and very low numbers in the false negative rate.

One of the main contributions of this thesis is the introduction of a top-down module in the early stages of object recognition, demonstrating the efficiency improvements in visual search tasks. The next step to boost the recognition process is the consideration of hierarchical features in the initial steps of the object recognition architecture. Common approaches to component-based recognition consider every part independently during interest point selection and description. By contrast, it seems that humans learn visual features in a hierarchical manner, by encoding a sort of superfeatures that can be explained in terms of smaller subfeatures [2].

Experiments with human subjects have shown that adults and infants are able to extract conditional probability statistics between particular elements in specific spatial configurations [15]. The experimental evidence also suggests that the human visual system eliminates (reduces the weight of) features embedded in larger features, but only if they never appear

outside of the larger features [34]. This can be seen as a dimensionality reduction procedure. Furthermore, an open question in human vision research is the construction process of these superfeatures. The experiments presented in [34] suggest that the boundaries between superfeatures contain low-level features with low predictive power (i.e. we can not rely on them to predict the appearance of other features).

Recent works in computer vision [32, 31] have addressed the construction of hierarchies of low-level features, but the construction process relies on the "parts composed by parts" approach, instead of the biologically motivated construction based on the feature predictive power. Thus, the addition of a hierarchical feature construction module to the object recognition architecture will bring more flexibility and efficiency to the state-of-the-art object models.

Appendix A

Gabor wavelength saliency function

A.1 Gabor wavelength saliency kernel

The closed form expression of Equation (2.10) is:

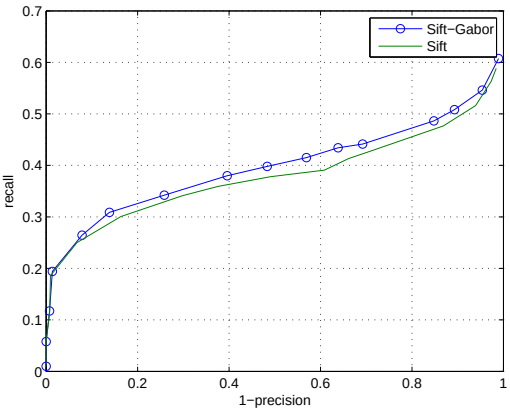
$$
\begin{aligned}
& w_d(r,\lambda) = \frac{J_0(2\pi r/\lambda)\sqrt{\pi/2}}{r}(\mathrm{erf}(3\sqrt{2}r/\lambda) - \mathrm{erf}(\frac{r}{\sqrt{2}\lambda})) + \\
& \frac{1}{4r^2}\left[-24e^{-\pi^2/18-\frac{18r^2}{\lambda^2}} + 4e^{-2\pi^2-\frac{r^2\lambda^2}{2}} - 2\sqrt{2}e^{-\frac{2\pi r}{\lambda}}\pi^{3/2}\mathrm{erf}\left(\frac{\pi - 18r/\lambda}{3\sqrt{2}}\right) - \right. \\
& \frac{\lambda}{r}\left(\sqrt{2\pi}e^{-\frac{2\pi r}{\lambda}}(1+2\pi r/\lambda)\mathrm{erf}\left(\frac{-2\pi + r/\lambda}{\sqrt{2}}\right)\right) + 2\sqrt{2}e^{\frac{2\pi r}{\lambda}}\pi^{3/2}\mathrm{erf}\left(\frac{2\pi + r/\lambda}{\sqrt{2}}\right) - \\
& \frac{e^{\frac{2\pi r}{\lambda}}\sqrt{2\pi}\mathrm{erf}\left(\frac{2\pi + r/\lambda}{\sqrt{2}}\right)\lambda}{r} + \frac{e^{-\frac{2\pi r}{\lambda}}\sqrt{2\pi}\mathrm{erf}\left(\frac{-\pi + 18r/\lambda}{3\sqrt{2}}\right)\lambda}{r} - 2\sqrt{2}e^{\frac{2\pi r}{\lambda}}\pi^{3/2}\mathrm{erf}\left(\frac{\pi + 18r/\lambda}{3\sqrt{2}}\right) \\
& \left. + \frac{e^{\frac{2\pi r}{\lambda}}\sqrt{2\pi}\mathrm{erf}\left(\frac{\pi + 18r/\lambda}{3\sqrt{2}}\right)\lambda}{r}\right]
\end{aligned}
$$

where $r = \sqrt{x^2 + y^2}$, $\mathrm{erf}(z)$ is the error function, $J_0(z)$ is the Bessel function of first kind.

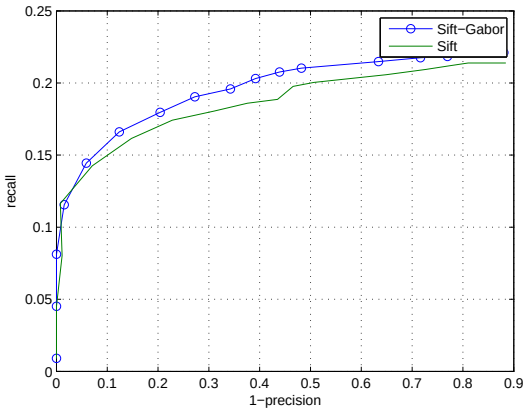
Appendix B

Image matching results

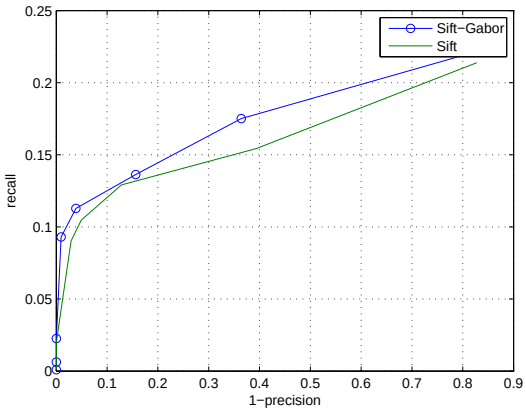
This appendix contains the matching results of the image neighborhood matching experiment [83] to evaluate comprehensively the improvement in the SIFT descriptor [70] explained in Chapter 4.



(a) threshold-based matching



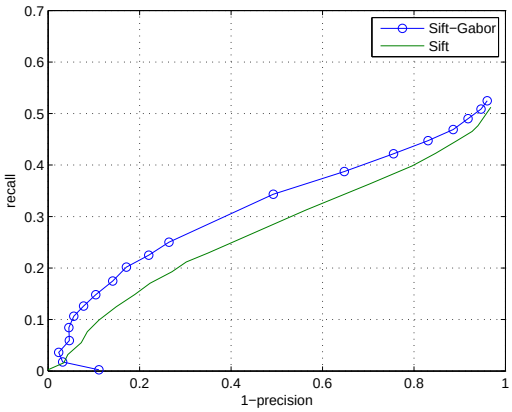
(b) nearest neighbor matching



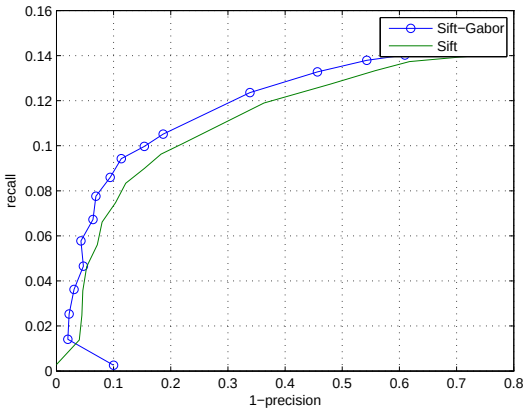
(c) nearest neighbor ratio matching

(d)

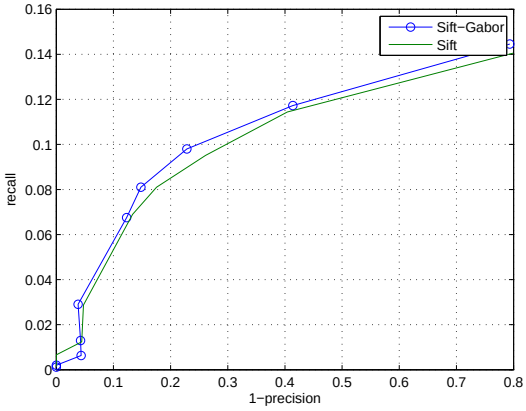
Figure B.1: *recall* vs. $1 - precision$ curves of Hessian-affine regions matched using textured images in Figure B.1(d), related by a viewpoint transformation.



(a) threshold-based matching



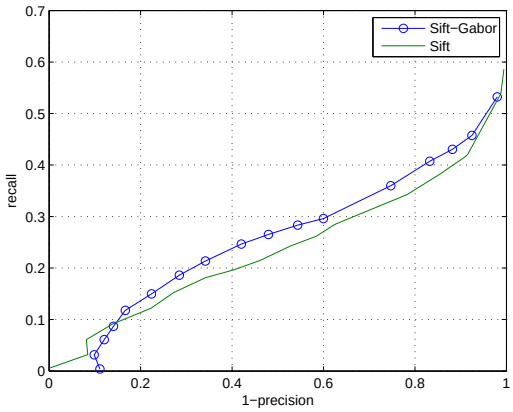
(b) nearest neighbor matching



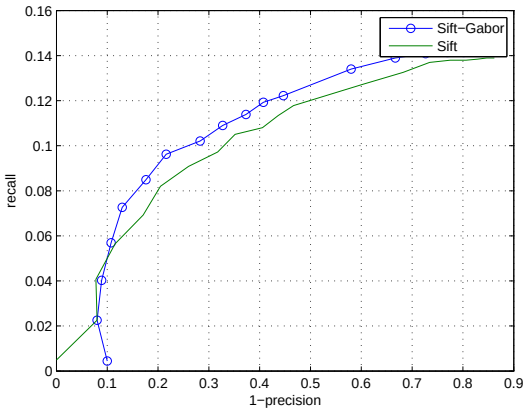
(c) nearest neighbor ratio matching

(d)

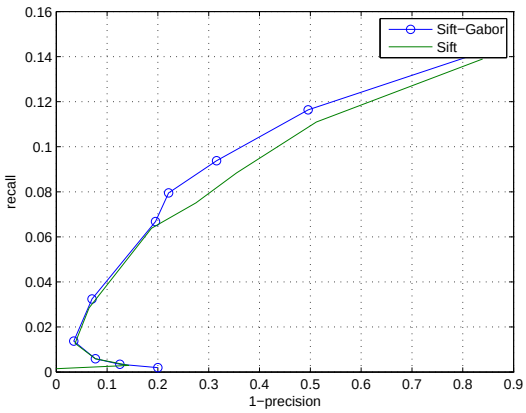
Figure B.2: *recall* vs. $1 - precision$ curves of Harris-affine regions matched using structured images in Figure B.2(d), related by a viewpoint transformation.



(a) threshold-based matching



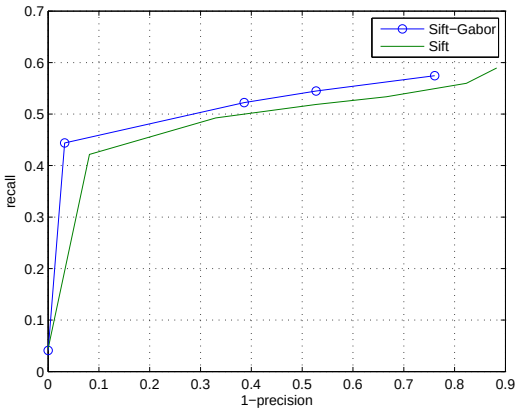
(b) nearest neighbor matching



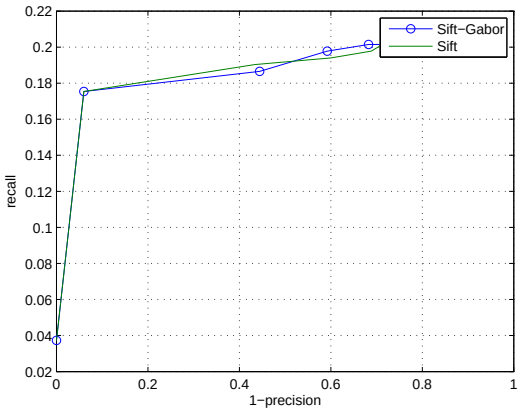
(c) nearest neighbor ratio matching

(d)

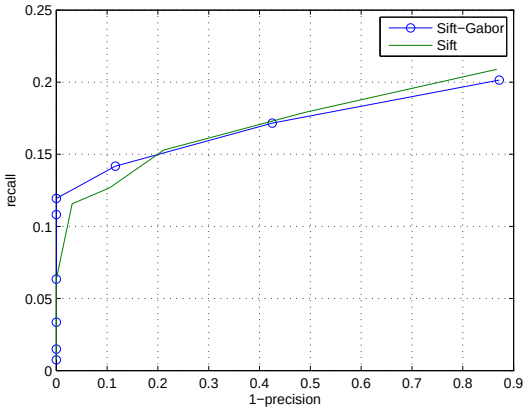
Figure B.3: *recall* vs. $1 - precision$ curves of Harris-affine regions matched using structured images in Figure B.3(d), related by a viewpoint transformation.



(a) threshold-based matching



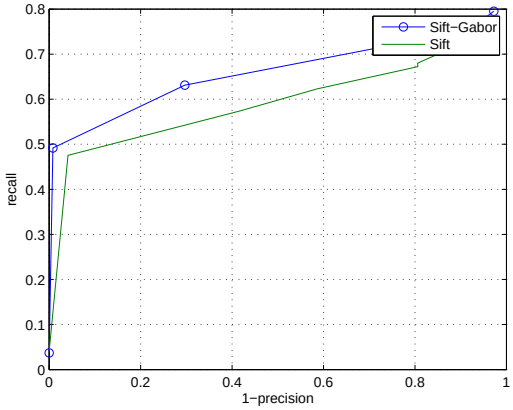
(b) nearest neighbor matching



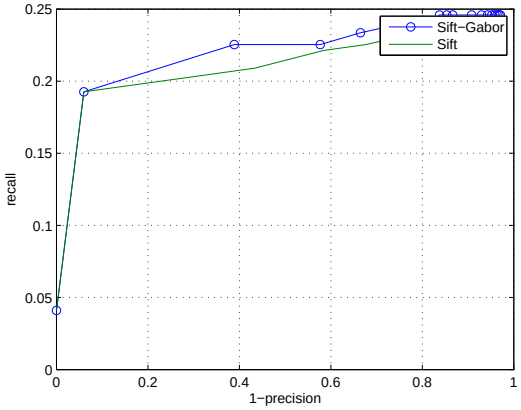
(c) nearest neighbor ratio matching

(d)

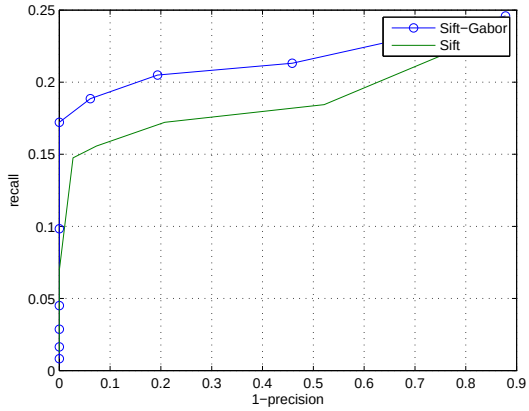
Figure B.4: *recall* vs. $1 - precision$ curves of Harris-laplace regions matched using textured images in Figure B.4(d), related by a scale + rotation transformation.



(a) threshold-based matching



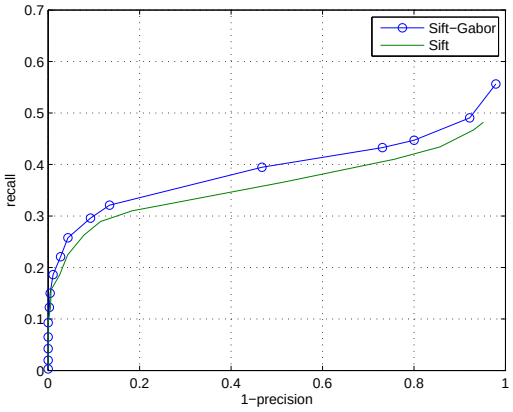
(b) nearest neighbor matching



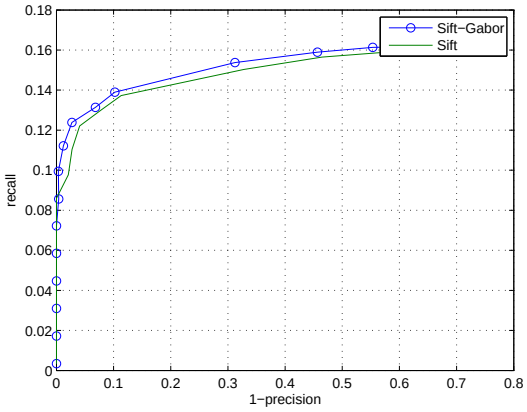
(c) nearest neighbor ratio matching

(d)

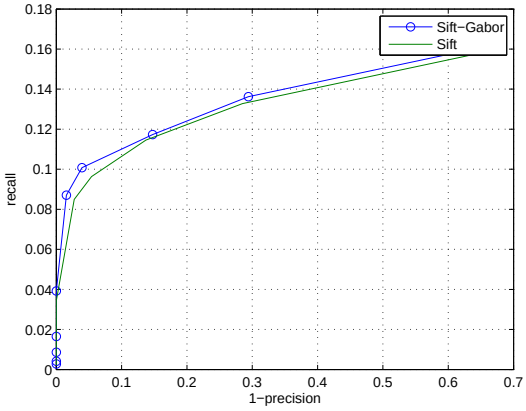
Figure B.5: *recall* vs. $1-precision$ curves of Hessian-laplace regions matched using textured images in Figure B.5(d), related by a scale + rotation transformation.



(a) threshold-based matching



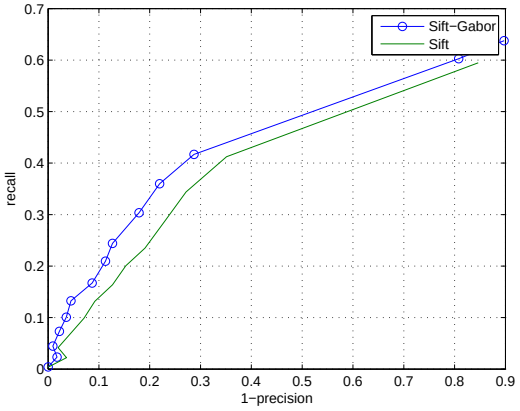
(b) nearest neighbor matching



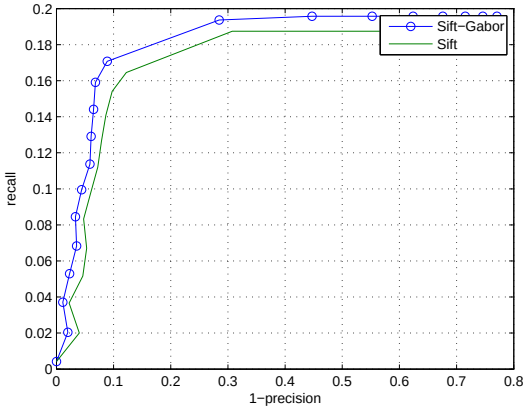
(c) nearest neighbor ratio matching

(d)

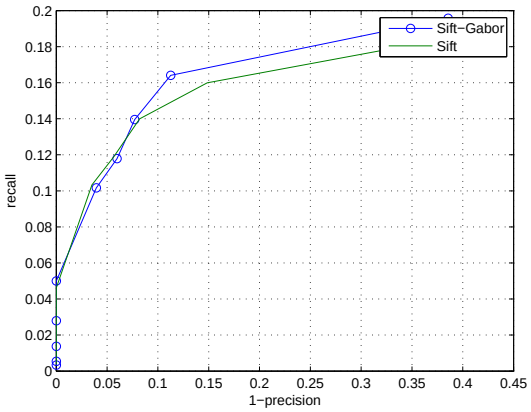
Figure B.6: *recall* vs. $1-precision$ curves of Harris-laplace regions matched using structured images in Figure B.6(d), related by a scale + rotation transformation.



(a) threshold-based matching



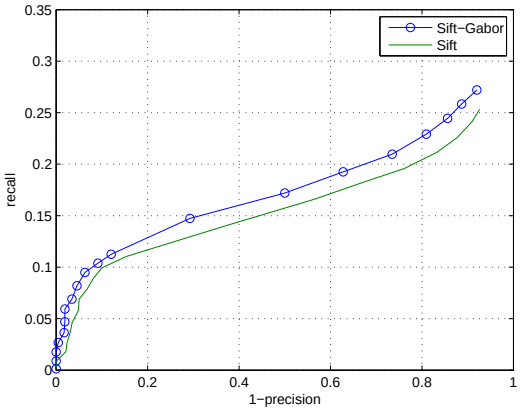
(b) nearest neighbor matching



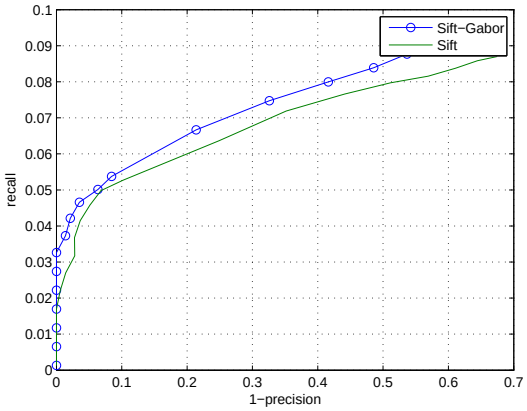
(c) nearest neighbor ratio matching

(d)

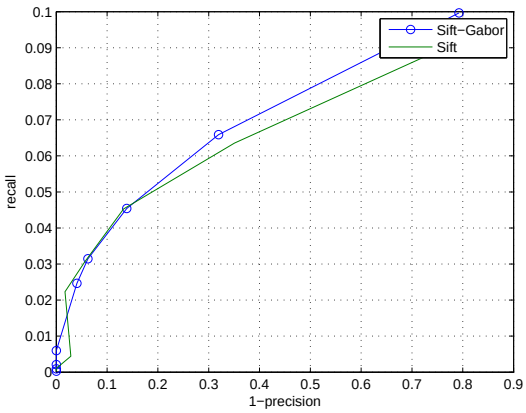
Figure B.7: *recall* vs. $1-precision$ curves of Hessian-affine regions matched using structured images in Figure B.7(d), related by a blur transformation.



(a) threshold-based matching



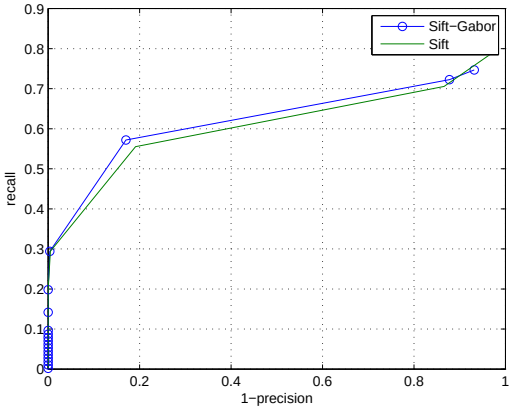
(b) nearest neighbor matching



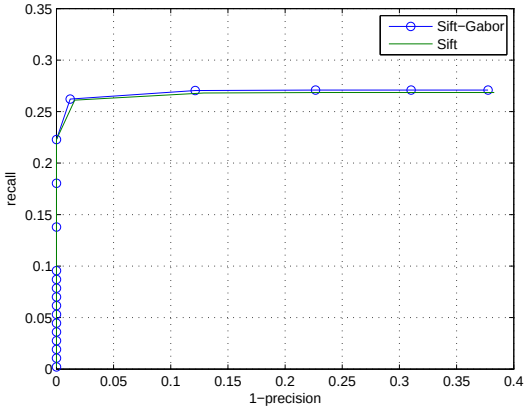
(c) nearest neighbor ratio matching

(d)

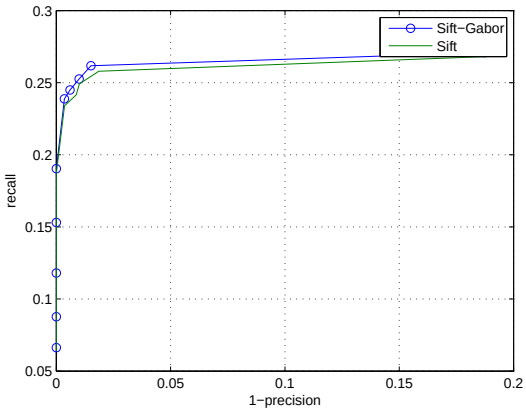
Figure B.8: *recall* vs. $1 - precision$ curves of Hessian-affine regions matched using textured images in Figure B.8(d), related by a blur transformation.



(a) threshold-based matching



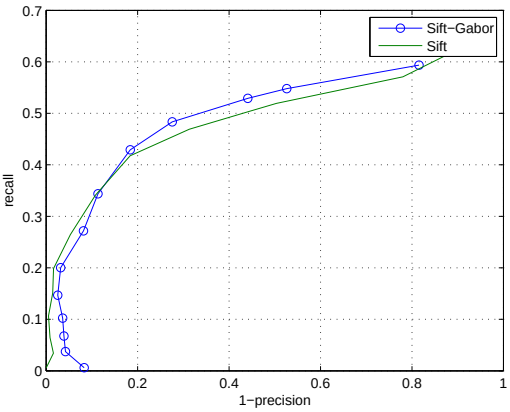
(b) nearest neighbor matching



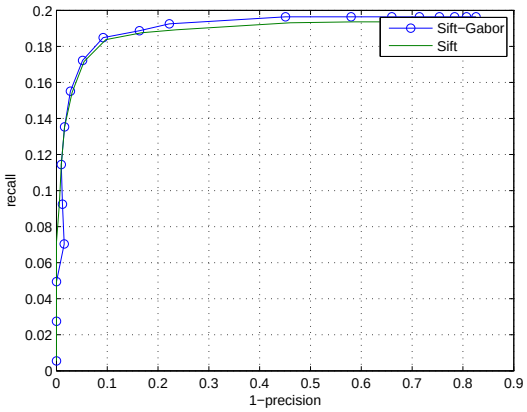
(c) nearest neighbor ratio matching

(d)

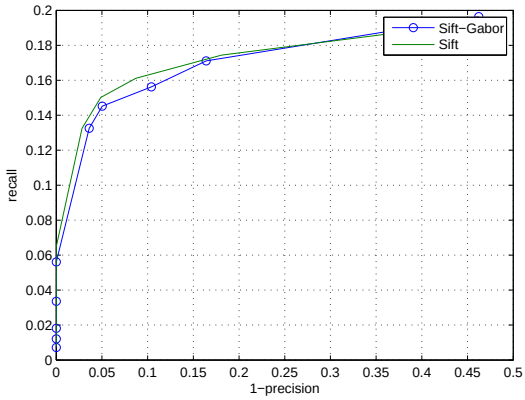
Figure B.9: *recall* vs. $1-precision$ curves of Hessian-affine regions matched using structured images in Figure B.9(d), related by a JPEG transformation.



(a) threshold-based matching



(b) nearest neighbor matching



(c) nearest neighbor ratio matching

(d)

Figure B.10: *recall* vs. 1−*precision* curves of Hessian-affine regions matched using structured images in Figure B.10(d), related by an illumination transformation.

Appendix C

Object category results

This appendix contains the object recognition results of the following categories: camel, car (side view), car (rear view), face, guitar, leaves and motorbikes. These experiments evaluate several local descriptors using an appearance-only object model (Chapter 5).

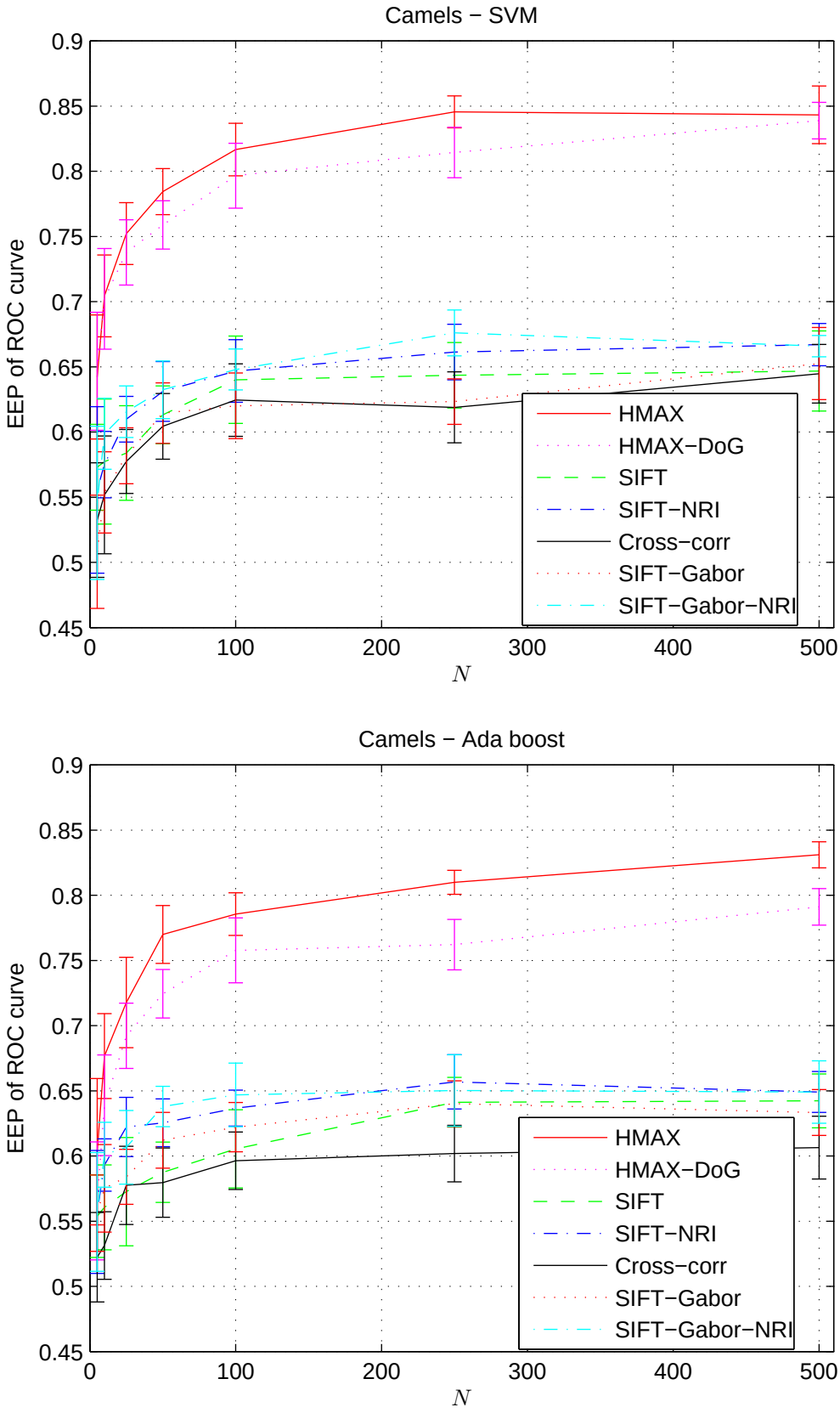


Figure C.1: Recognition performance of the camel category, for several local descriptors.

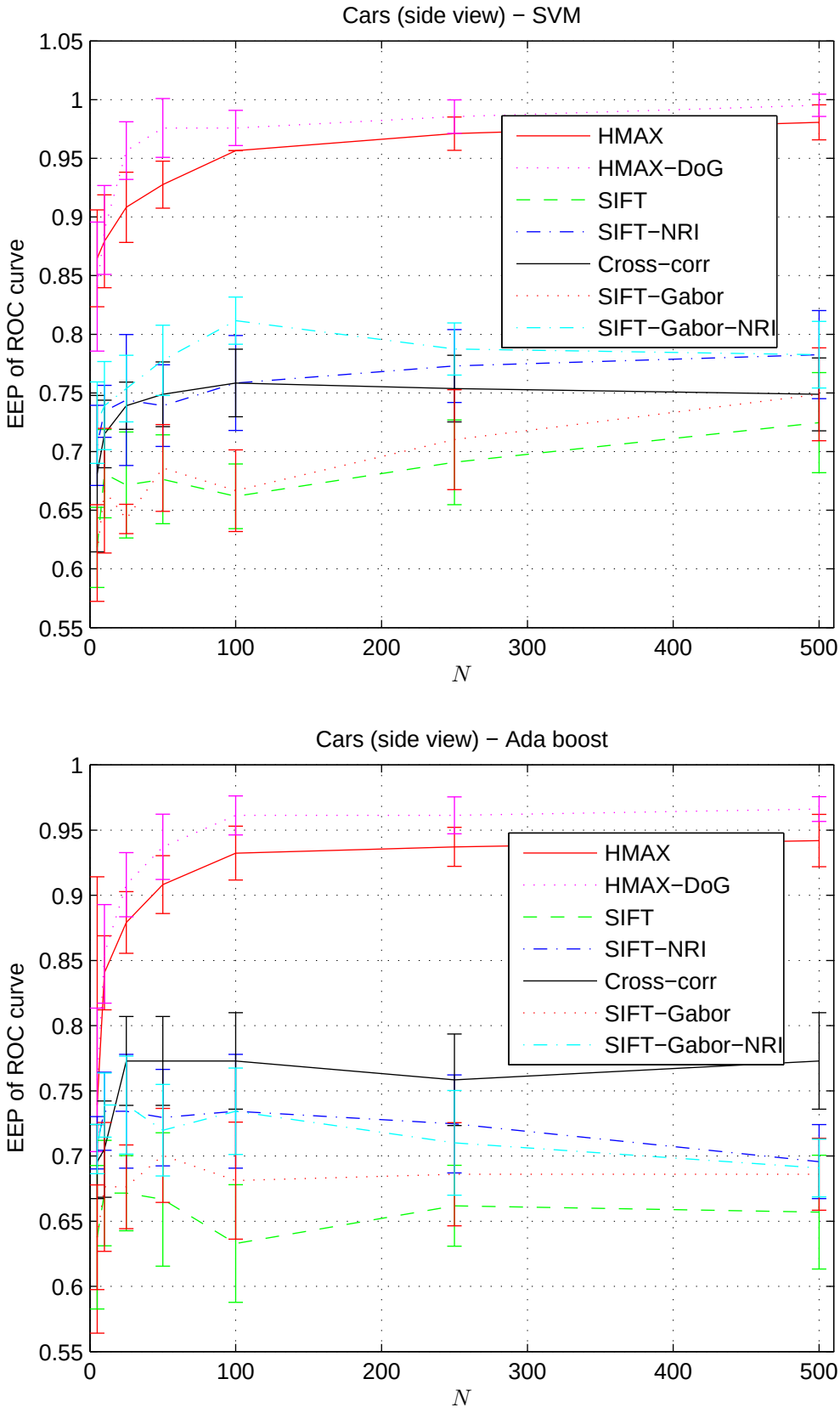


Figure C.2: Recognition performance of the car category (side view), for several local descriptors.

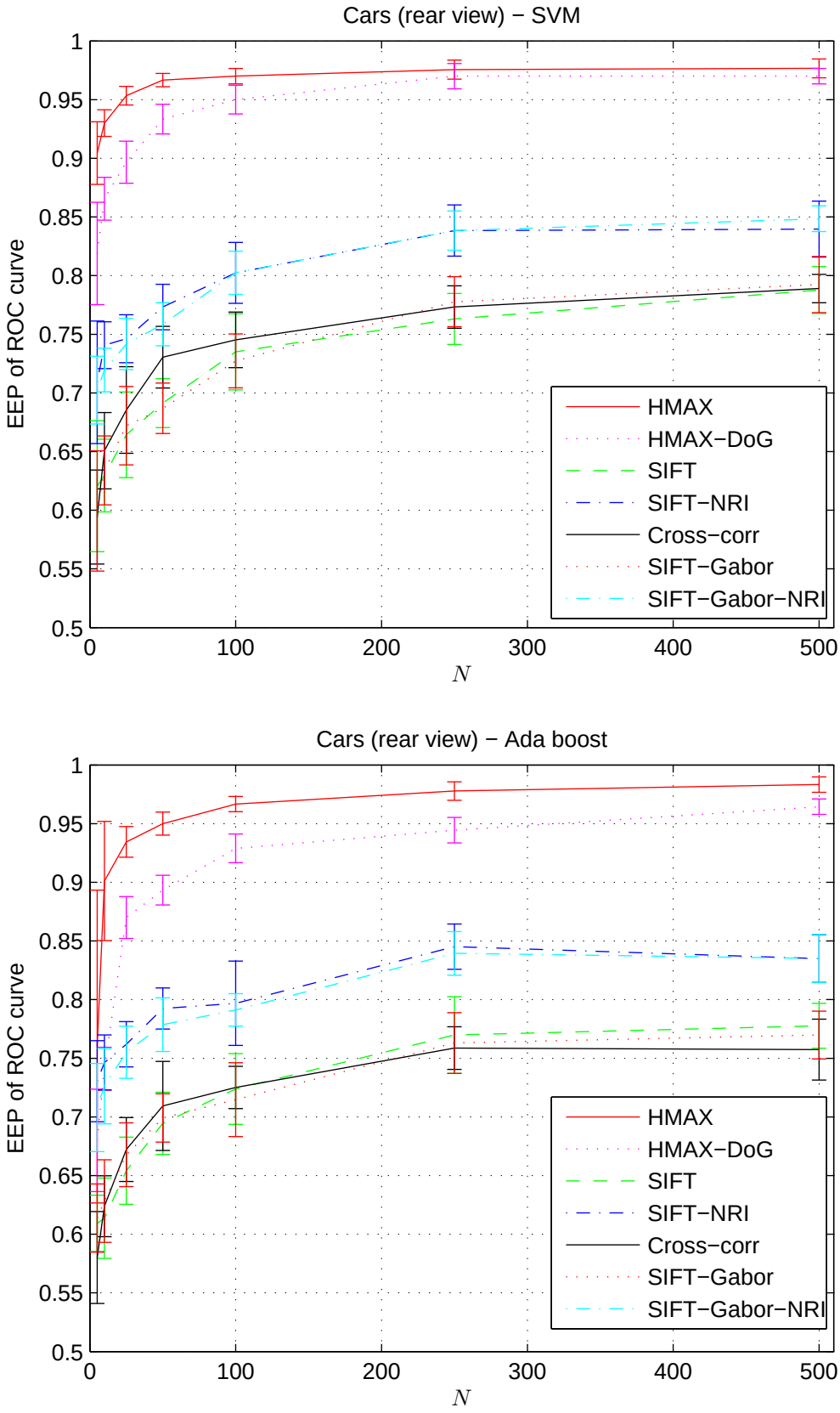


Figure C.3: Recognition performance of the car category (rear view), for several local descriptors.

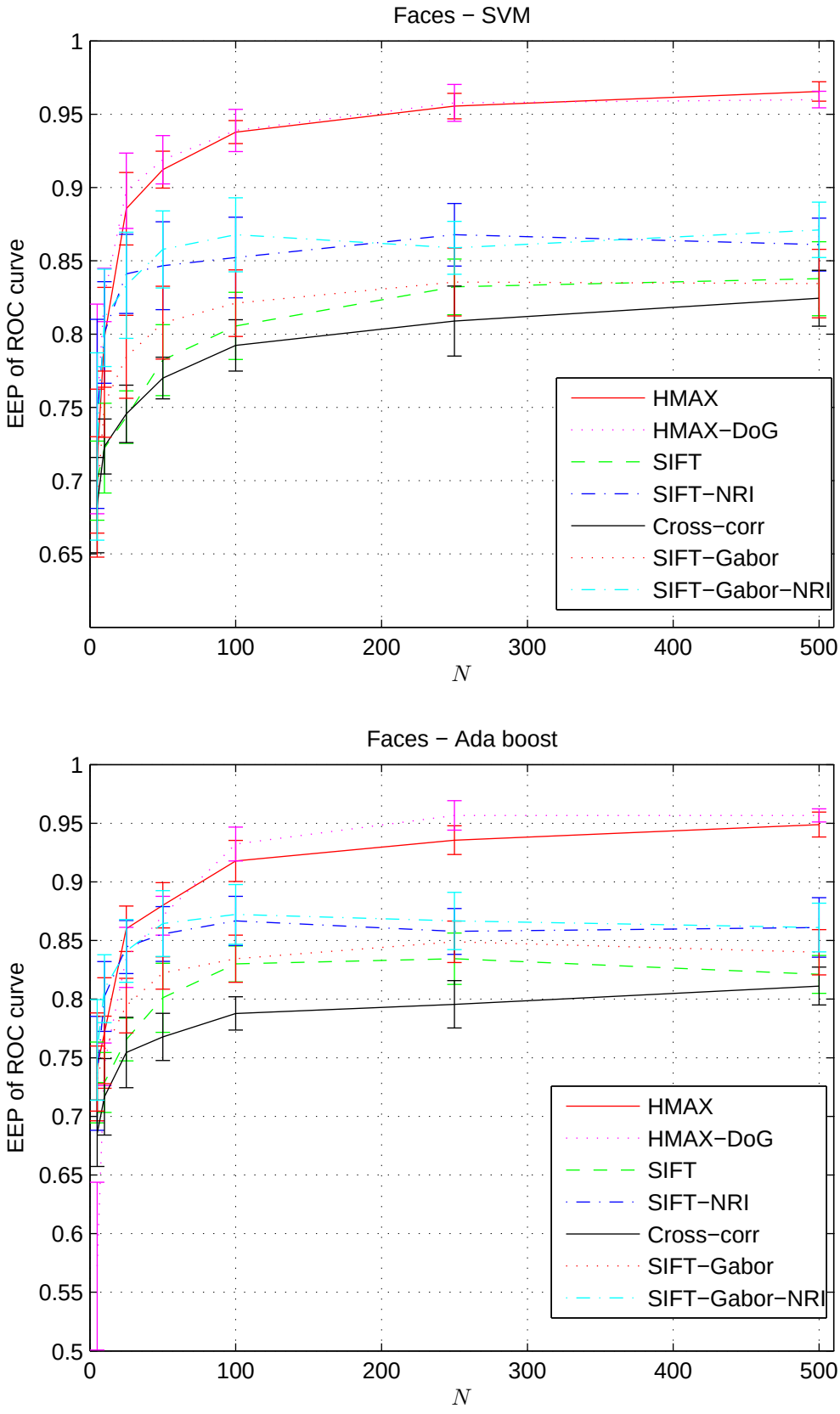


Figure C.4: Recognition performance of the face category, for several local descriptors.

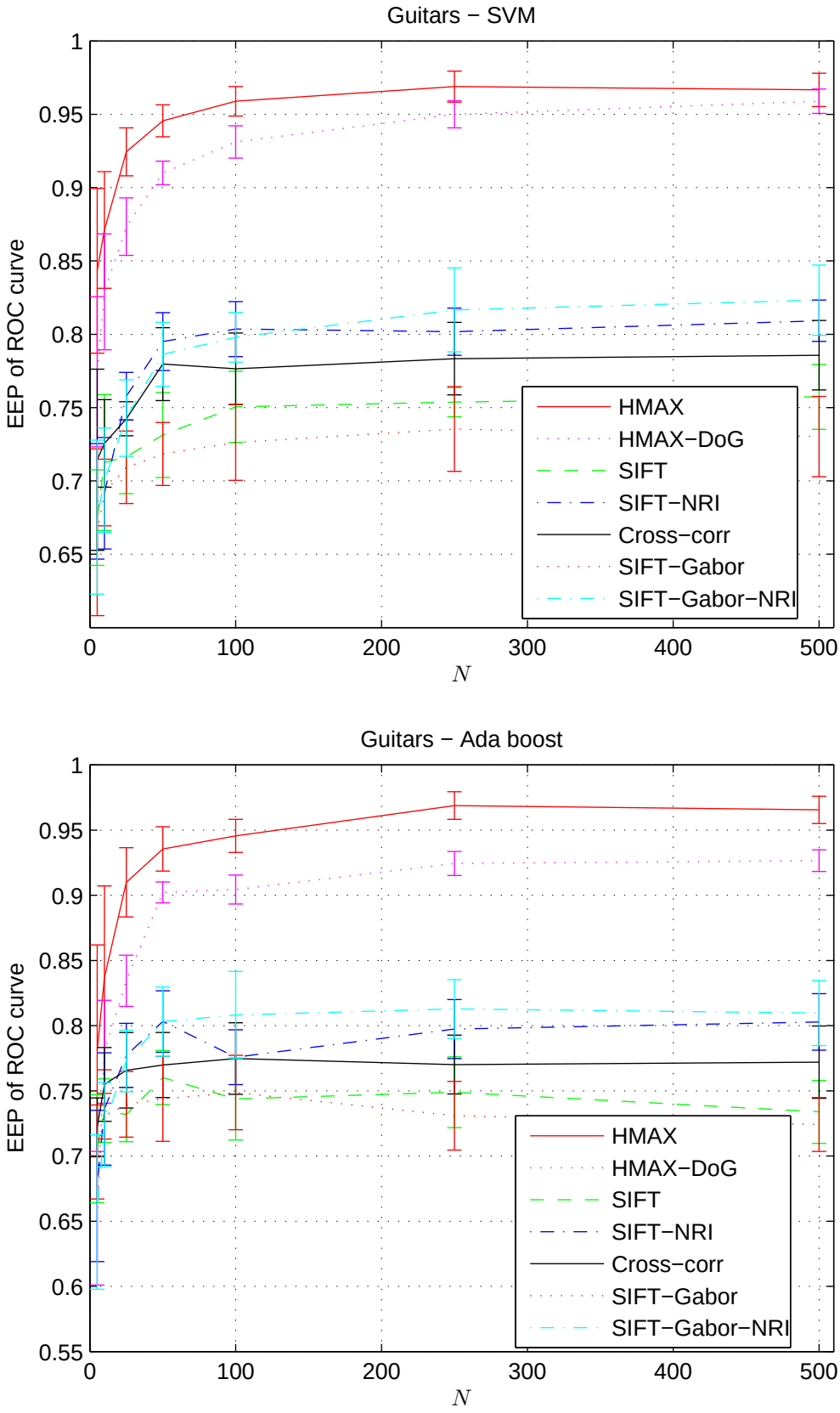


Figure C.5: Recognition performance of the guitar category, for several local descriptors.

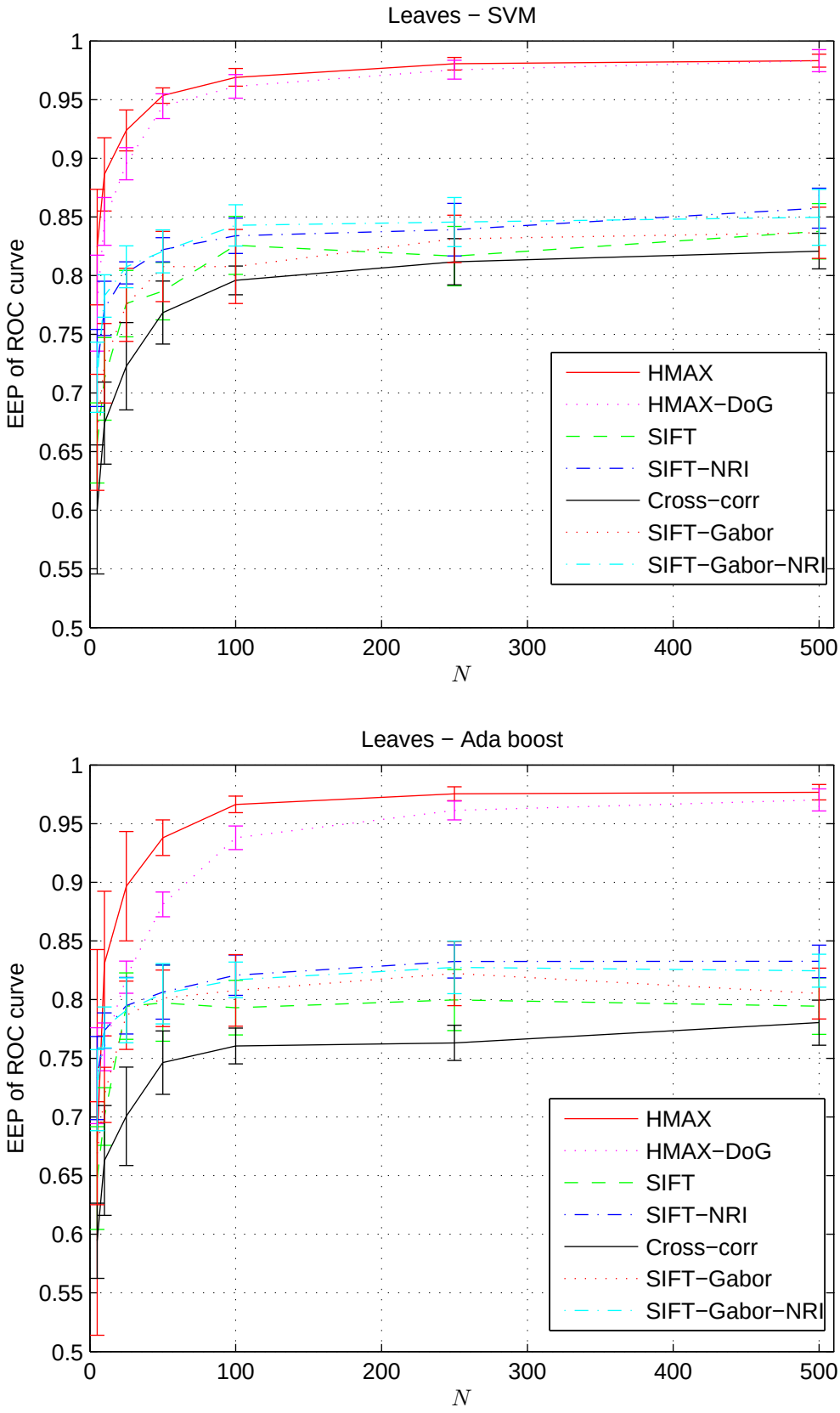


Figure C.6: Recognition performance of the leaves category, for several local descriptors.

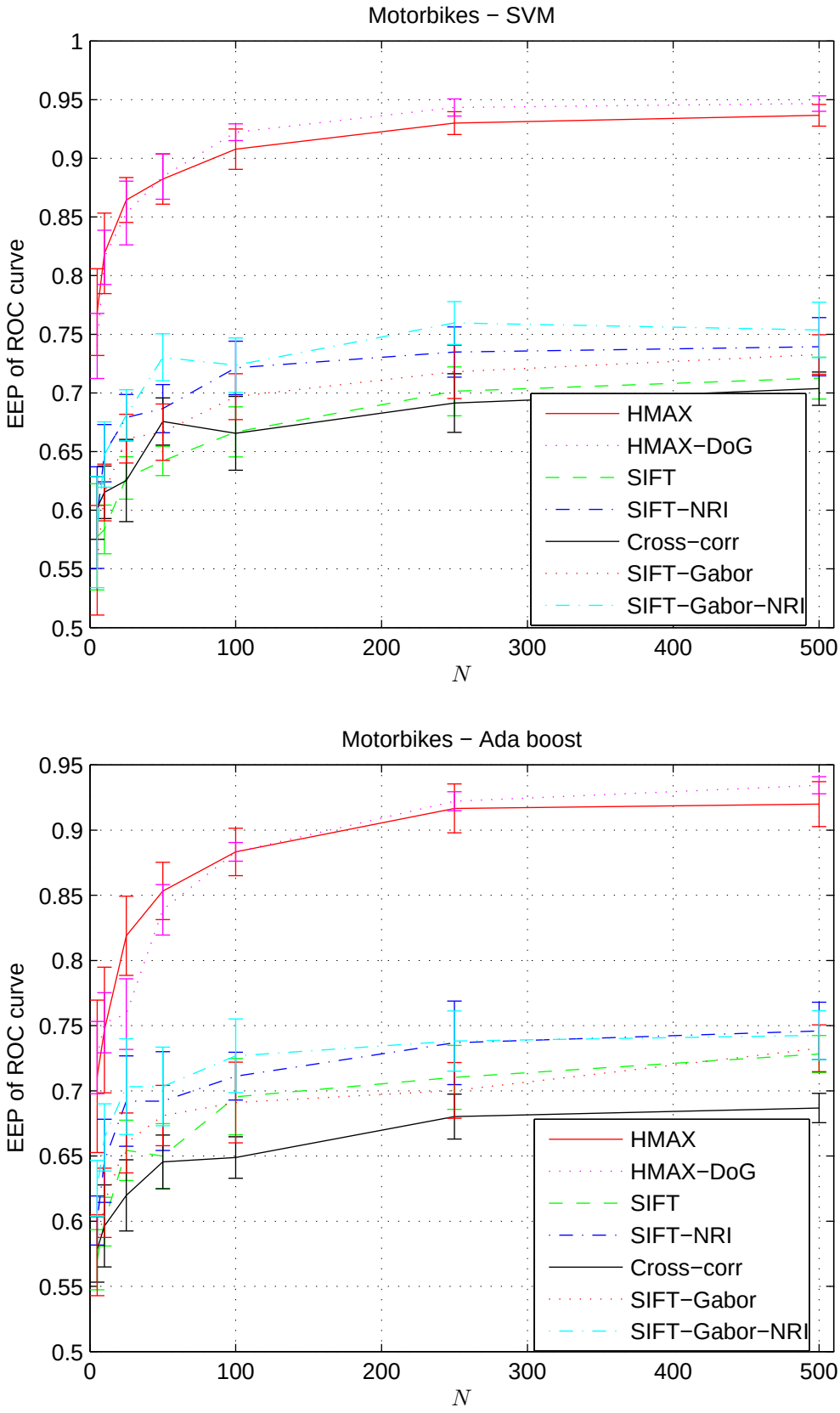


Figure C.7: Recognition performance of the motorbike category, for several local descriptors.

Bibliography

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1475–1490, 2004.

[2] M. Ahissar and S. Hochstein. The reverse hierarchy theory of visual perceptual learning. *Trends in Cognitive Sciences*, 8(10):457–464, October 2004.

[3] A. Ashbrook, N. Thacker, P. Rockett, and C. Brown. Robust recognition of scaled shapes using pairwise geometric histograms. In *Proceedings Sixth British Machine Vision conference*, pages 503–512, 1995.

[4] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(4):509–522, April 2002.

[5] J. Ben-Arie and Z. Wang. Pictorial recognition of objects employing affine invariance in the frequency domain. *IEEE Journal on Pattern Analysis and Machine Inteligence*, 20(6):604–618, 1998.

[6] A. Bernardino and J. Santos-Victor. A real-time Gabor primal sketch for visual attention. In *Proceedings of the Iberian Conference on Pattern Recognition and Image Analysis*, pages 335–342, 2005.

[7] A. Bernardino and J. Santos-Victor. Fast IIR isotropic 2-D complex Gabor filters with boundary initialization. *IEEE Transactions on Image Processing*, 15(11):3338–3348, November 2006.

[8] I. Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94:115–147, 1987.

[9] A. Bosch, A. Zisserman, and X. Muñoz. Scene classification via plsa. In *Proceedings of the European Conference on Computer Vision*, pages 642–651, 2006.

[10] M.C. Burl and P. Perona. Recognition of planar object classes. In *1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1996*, pages 223–230, 1996.

[11] J. F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 679–698, 1986.

[12] G. Carneiro and A. Jepson. Local phase-based features. In *European Conference on Computer Vision*, pages 282–296, 2002.

[13] C. Chang and C. Lin. LIBSVM: a library for support vector machines, April 2005. Available from: `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`.

[14] M. Chun and J. Wolfe. Visual attention. In E. Goldstein, editor, *Blackwell Handbook of Perception*. Blackwell Publishers, July 2000.

[15] C. Conway, R.L. Goldstone, and M. Christiansen. Spatial constraints on visual statistical learning of multi-element displays. In *Proceedings of the Twenty-ninth Annual Conference of the Cognitive Science Society*, pages 185–190, 2007.

[16] T. Cootes. Available from: `http://www.isbe.man.ac.uk/~bim/data/tarfd_markup/tarfd_markup.html`.

[17] D. J. Crandall and D. P. Huttenlocher. Weakly supervised learning of part-based spatial models for visual object recognition. In *Proceedings of the European Conference on Computer Vision (1)*, pages 16–29, 2006.

[18] J. L. Crowley and A. C. Parker. A representation for shape based on peaks and ridges in the difference of low pass transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(2):156–170, March 1984.

[19] G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning in Computer Vision*, pages 1–22, 2004.

[20] P. Daniel and D. Whitterridge. The representation of the visual field on the cerebral cortex in monkeys. *Journal of Physiology*, 159:203–221, 1961.

[21] J. Daugman. Two-dimensional spectral analysis of cortical receptive fields profiles. *Vision Research*, 20:847–856, 1980.

[22] J. Daugman. High confidence visual recognition of persons by a test of statistical independence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1148–1161, 1993.

[23] D. Deco and T.S. Lee. The role of early visual cortex in visual integration: a neural model of recurrent interaction. *European Journal of Neuroscience*, 20:1089–1100, 2004.

[24] D. Dunn, W. Higgins, and J. Wakeley. Texture segmentation using 2-D Gabor elementary functions. *IEEE Journal on Pattern Analysis and Machine Inteligence*, 16(2):130–149, 1994.

[25] L. Fei-Fei, R. Fergus, and P. Perona. A bayesian approach to unsupervised one-shot learning of object categories. In *IEEE International Conference on Computer Vision*, volume 2, pages 1134–1141, October 2003.

[26] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594– 611, 2006.

[27] L. Fei-Fei, R. Fergus, and A. Torralba. Available from: `http://people.csail.mit.edu/torralba/iccv2005/`.

[28] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proceedings IEEE conference on Computer Vision and Pattern Recognition*, volume 2, pages 524–531, June 2005.

[29] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings IEEE conference on Computer Vision and Pattern Recognition*, volume 2, pages 264–271, 2003.

[30] R. Fergus, P. Perona, and A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 380 387, 2005.

[31] S. Fidler, M. Boben, and A. Leonardis. Similarity-based cross-layered hierarchical representation for object categorization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2008.

[32] S. Fidler and A. Leonardis. Towards scalable representations of object categories: Learning a hierarchy of parts. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1–8, 2007.

[33] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, c22(1):67– 92, January 1973.

[34] J. Fiser and R. Aslin. Encoding multielement scenes: Statistical learning of visual feature hierarchies. *Journal of Experimental Psychology*, 134(4):521–537, November 2005.

[35] W. Förstner. A framework for low-level feature extraction. In *Proceedings of the European Conference on Computer Vision*, volume 2, pages 383–394, 1994.

[36] W. Freeman and E. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, September 1991.

[37] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28(2):337–407, 1998.

[38] D. Gabor. Theory of communication. *Journal of IEE*, 93:429–459, 1946.

[39] D. Gao and N. Vasconcelos. Discriminant saliency for visual recognition from cluttered scenes. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 481–488. MIT Press, 2005.

[40] J. M. Geusebroek, A. W. M. Smeulders, and R. van den Boomgaard. Measurement of color invariants. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 50–57, June 2000.

[41] G.D. Hager and P.N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(10):1025–1039, October 1998.

[42] D. Hall, V. C. de Verdière, and J. Crowley. Object recognition using coloured receptive fields. In *Proceedings of European Conference on Computer Vision*, page 164 177, June 2000.

[43] M. Hamouz, J. Kittler, J.-K. Kamarainen, P. Paalanen, H. Kälviäinen, and J. Matas. Feature-based affine-invariant detection and localization of faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(9):1490–1495, September 2005.

[44] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.

[45] D. Heeger. Optical flow using spatiotemporal filters. *International Journal of Computer Vision*, 1:279–302, 1988.

[46] D. Hubel and T. Wiesel. Receptive fields of single neurones in the cat's striate cortex. *Journal of Physiology*, 148(3):574 591, October 1959.

[47] D. P. Huttenlocher and P. Felzenszwalb. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, January 2005.

[48] D.P. Huttenlocher and S. Ullman. Object recognition using alignment. In *Proceedings of the 1st International Conference on Computer Vision*, pages 102–111, 1987.

[49] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, November 1998.

[50] A. K. Jain and F. Farrokhnia. Unsupervised texture segmentation using Gabor filters. *Pattern Recognition*, 24(12):1167–1186, December 1991.

[51] T. Kadir and M. Brady. Saliency, scale and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

[52] T. Kadir and M. Brady. An affine invariant salient region detector. In *European Conference on Computer Vision*, volume 1, pages 228–241, 2004.

[53] J.-K. Kamarainen, V. Kyrki, and H. Kälviäinen. Fundamental frequency Gabor filters for object recognition. In *Proceedings of the 16th International Conference on Pattern Recognition*, pages 628–631, 2002.

[54] S. Kastner and L. G. Ungerleider. Mechanisms of visual attention in the human cortex. *Annual Review of Neuroscience*, 23(1):315–341, 2000. Available from: `http://arjournals.annualreviews.org/doi/abs/10.1146/annurev.neuro.23.1.315`.

[55] Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 511–517, 2004.

[56] J. Koenderink and A. van Doorn. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.

[57] J. Koenderink and A. van Doorn. Representation of local geometry in the visual system. *Biological Cybernetics*, 55:367–375, 1987.

[58] J. Koenderink and A. van Doorn. Generic neighborhood operators. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(6):597 – 605, June 1992.

[59] V. Kyrki, J.-K. Kamarainen, and H. Kälviäinen. Simple Gabor feature space for invariant object recognition. *Pattern Recognition Letters*, 25:311–318, 2004.

[60] M. Lades, J.C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R.P. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42(3):300–311, March 1993.

[61] T. Lee. Image representation using 2D Gabor wavelets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(10):959–971, October 1996.

[62] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *Workshop on Statistical Learning in Computer Vision, ECCV*, pages 17–32, 2004.

[63] T.K. Leung, M.C. Burl, and P. Perona. Finding faces in cluttered scenes using random labeled graph matching. In *Proceedings Fifth International Conference on Computer Vision*, pages 637 – 644, 1995.

[64] T. Lindeberg. On scale selection for differential operators. In *Proceedings 8th. Conference on Image Analysis*, pages 857–866, May 1993.

[65] T. Lindeberg. Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):117–154, November 1998.

[66] T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79–116, November 1998.

[67] T. Lindeberg and J. Gårding. Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure. *Image and vision computing*, 15:415–434, 1997.

[68] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Image Processing*, 11(4):467 – 476, April 2002.

[69] D. Lowe. Object recognition from local scale-invariant features. In *Proceedings IEEE conference on Computer Vision and Pattern Recognition*, pages 1150–1157, 1999.

[70] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[71] D. G. Lowe. The viewpoint consistency constraint. *International Journal of Computer Vision*, 1(1):57–72, 1987.

[72] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pages 674–679, April 1981.

[73] S.G. Mallat. *A Wavelet Tour of Signal Processing, Second Edition (Wavelet Analysis & Its Applications)*. Academic Press, September 1999.

[74] S.G. Mallat and Z. Zhifeng. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, Dec 1993.

[75] B.S. Manjunath, R. Chellappa, and C. von der Malsburg. A feature based approach to face recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 373 – 378, June 1992.

[76] B.S. Manjunath, C. Shekhar, and R. Chellappa. A new approach to image feature detection with applications. *Pattern Recognition*, 29(4):627–640, 1996.

[77] A.M. Martinez and R. Benavente. The AR face database. Technical Report 24, CVC, June 1998.

[78] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761 – 767, 2004.

[79] K. Mikolajczyk, B. Leibe, and B. Schiele. Multiple object class detection with a generative model. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 26–36, 2006.

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

[81] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *European Conference on Computer Vision*, pages 128–142. Springer, 2002.

[82] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 1(60):63–86, 2004.

[83] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, November 2005.

[84] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005.

[85] K. Mikolajczyk, A. Zisserman, and C. Schmid. Shape recognition with edge-based features. In *Proceedings of the British Machine Vision Conference*, volume 2, pages 779–788, 2003.

[86] P. Moreels, M. Maire, and P. Perona. Recognition by probabilistic hypothesis construction. In *Proceedings of European Conference on Computer Vision*, volume 1, pages 55–68, 2004.

[87] P. Moreno, A. Bernardino, and J. Santos-Victor. Appearance based salient point detection with intrinsic scale-frequency descriptor. In *Proceedings of Visualization Imaging and Image Processing*, pages 332–337, 2005.

[88] P. Moreno, A. Bernardino, and J. Santos-Victor. Gabor parameter selection for local feature detection. In *Proceedings of the Iberian Conference on Pattern Recognition and Image Analysis*, pages 128–142, 2005.

[89] P. Moreno, A. Bernardino, and J. Santos-Victor. Model based selection and classification of local features for recognition using Gabor filters. In *Proceedings of the International Conference on Image Analysis and Recognition*, volume 2, pages 181–192, September 2006.

[90] P. Moreno, A. Bernardino, and J. Santos-Victor. Improving the SIFT descriptor with smooth derivative filters. *Accepted for publication to Pattern Recognition Letters*, 2008.

[91] P. Moreno, M. Marín-Jiménez, A. Bernardino, J. Santos-Victor, and N. Pérez de la Blanca. A comparative study of local descriptors for object category recognition: Sift vs hmax. In *Proceedings of the 3rd Iberian Conference on Pattern Recognition and Image Analysis*, volume 1, pages 515–522, 2007.

[92] H. Murase and S. Nayar. Visual learning and recognition of 3-D objects from appearance. *International Journal of Computer Vision*, 14:5–24, 1995.

[93] K.R. Namuduri, R. Mehrotra, and N. Ranganathan. Edge detection models based on Gabor filters. In *Proceedings of the 11th International Conference on Pattern Recognition*, volume 3, pages 729–732, 1992.

[94] A. S. Ogale and Y. Aloimonos. Robust contrast invariant stereo correspondence. In *Proceedings IEEE conference on Robotics and Automation (ICRA)*, pages 819– 824, April 2005.

[95] A. Oliva and A. Torralba. Building the gist of a scene: The role of global image features in recognition. In *Visual Perception*, volume 155, pages 23–36. Elsevier, 2006.

[96] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Weak hypotheses and boosting for generic object detection and recognition. In *Proceedings of the 8th European Conference on Computer Vision*, volume 2, pages 71–84, 2004.

[97] A. Opelt, A. Pinz, M.Fussenegger, and P.Auer. Generic object recognition with boosting. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 28(3):416–431, 2006.

[98] E. Osuna, R. Freund, and F. Girosi. Support Vector Machines: training and applications. Technical Report AI-Memo 1602, MIT, March 1997.

[99] M. Ouali, C. Laurgeaua, and D. Ziou. Dense disparity estimation using Gabor filters and image derivatives. In *Proceedings of Second International Conference on 3-D Imaging and Modeling*, pages 483–489, 1999.

[100] M. Pontil, S. Rogai, and A. Verri. Recognizing 3-D objects with linear support vector machines. In *Proceedings of the 5th European Conference on Computer Vision*, pages 469–483, 1998.

[101] M. Porat and Y. Zeevi. The generalized Gabor scheme of image representation in biological and machine vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):452–468, July 1988.

[102] R.P.N. Rao and D.H. Ballard. An active vision architecture based on iconic representations. *Artificial Intelligence*, 78(1-2):461–505, 1995.

[103] M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2(11):1019–1025, 1999.

[104] C. Rothwell, D. Forsyth, A. Zisserman, and J. Mundy. Extracting projective structure from single perspective views of 3D point sets. In *Proceedings of the 4th International Conference on Computer Vision*, pages 573–582, 1993.

[105] W. Rucklidge. *Efficient Visual Recognition Using the Hausdorff Distance*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1996.

[106] B. Schiele and J. L. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1):31–50, 2000.

[107] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997.

[108] S. Se, D. Lowe, and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, 21(8):735–758, 2002.

[109] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Object recognition with cortex-like mechanisms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):411–426, 2007.

[110] T. Serre, L. Wolf, and T. Poggio. Object recognition with features inspired by visual cortex. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 994–1000, June 2005.

[111] S. J. Shi and C. Tomasi. Good features to track. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 593–600, June 1994.

[112] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their localization in images. In *Proceedings of the Tenth IEEE International Conference on Computer Vision*, volume 1, pages 370–377, 2005.

[113] F. Smeraldi and J. Bigun. Retinal vision applied to facial features detection and face authentication. *Pattern Recognition Letters*, 23:463–475, 2002.

[114] A. Torralba, K. P. Murphy, , and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 762–769, 2004.

[115] A. Torralba, K.P. Murphy, and W.T. Freeman. Sharing visual features for multiclass and multiview object detection. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 29(5):854–869, 2007.

[116] B. Triggs. Detecting keypoints with stable position, orientation, and scale under illumination changes. In *European Conference on Computer Vision*, volume 3, pages 100–113, 2004.

[117] N. Vasconcelos. Feature selection by maximum marginal diversity. In *Advances in Neural Information Processing Systems*, pages 1351–1358. MIT Press, 2002.

[118] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings IEEE conference on Computer Vision and Pattern Recognition*, volume 1, pages 511–518, 2001.

[119] C. Wallraven, B. Caputo, and A. Graf. Recognition with local features: The kernel recipe. In *Proceedings of Ninth IEEE International Conference on Computer Vision*, volume 1, pages 257– 264, October 2003.

[120] D. Walther, U. Rutishauser, C. Koch, and P. Perona. Selective visual attention enables learning and recognition of multiple objects in cluttered scenes. *Computer Vision and Image Understanding*, 100(1-2):41–63, 2005.

[121] M. Weber, M. Welling, and P. Perona. Towards automatic discovery of object categories. In *Proceedings IEEE conference on Computer Vision and Pattern Recognition*, volume 2, pages 101–108, 2000.

[122] L. Wiskott, J.-M. Fellous, N. Kuiger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):775–779, July 1997.

[123] I.T. Young, L.J. van Vliet, and M. van Ginkel. Recursive Gabor filtering. *IEEE Transactions on Signal Processing*, 50(11):2798–2805, November 2002.

[124] Z. Zhu, H. Lu, and Y. Zhao. Multi-scale analysis of odd Gabor transform for edge detection. In *Proceedings of the First International Conference on Innovative Computing, Information and Control*, volume 2, pages 578–581, 2006.