



**HAL**  
open science

# Selection of Discriminative Regions and Local Descriptors for Generic Object Class Recognition

Gyuri Dorkó

► **To cite this version:**

Gyuri Dorkó. Selection of Discriminative Regions and Local Descriptors for Generic Object Class Recognition. Human-Computer Interaction [cs.HC]. Institut National Polytechnique de Grenoble - INPG, 2006. English. NNT: . tel-00555064

**HAL Id: tel-00555064**

**<https://theses.hal.science/tel-00555064>**

Submitted on 12 Jan 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

*N° attribué par la bibliothèque*

--	--	--	--	--	--	--	--	--	--

**THÈSE**

pour obtenir le grade de

**DOCTEUR DE L'INPG**

**Spécialité : Mathématiques et Informatique**

préparée au laboratoire GRAVIR – IMAG, projet LEAR,  
dans le cadre de **l'Ecole Doctorale Mathématiques, Sciences et Technologie  
de l'Information**

présentée et soutenue publiquement

par

**Gyuri DORKÓ**

le 9 juin 2006

**Selection of Discriminative Regions and Local  
Descriptors for Generic Object Class Recognition**

---

**Directeur de thèse : Dr. Cordelia SCHMID**

---

**JURY**

Prof. Roger MOHR,	Président
Prof. Bernt SCHIELE,	Rapporteur
Prof. Andrew ZISSERMAN,	Rapporteur
Dr. Cordelia SCHMID,	Directeur de thèse
Dr. Tinne TUYTELAARS,	Examineur



TO MY PARENTS  
SZÜLEIMNEK





# SELECTION OF DISCRIMINATIVE REGIONS AND LOCAL DESCRIPTORS FOR GENERIC OBJECT CLASS RECOGNITION

Gyuri DORKÓ, Ph.D. dissertation

Institut National Polytechnique de Grenoble, 9 June 2006

Object category recognition is one of the most difficult problems in computer vision. It involves recognizing objects despite intra-class variations, viewpoint changes and background clutter. The goal of this thesis is to investigate robust invariant local image description and the selection of discriminative features. We show that class-discriminative scale-invariant features achieve excellent results for image-level categorization and object localization. We present solutions for two key problems: (i) we improve the quality of the image description based on a novel scale-invariant keypoint detection method and (ii) we integrate feature filtering techniques into our object models.

Our novel scale-invariant detector is based on the idea of a “maximally stable description”, i.e., the descriptor should be stable even in the presence of minor variations of the detector. The technique performs scale selection based on a region descriptor, here SIFT, and chooses regions for which this descriptor is maximally stable, i.e., the difference between descriptors extracted for consecutive scales reaches a minimum. This scale selection technique is applied to multi-scale Harris and Laplacian points. Experimental results evaluate the performance of our detector and show that it outperforms existing ones in the context of image matching, category and texture classification, as well as object localization.

To construct object models based on discriminative features, we first cluster the scale-invariant descriptors and obtain a set of “visual words”. We then estimate the discriminative information of these clusters based on different feature selection techniques—several of which are traditionally used in text retrieval. We discuss their properties—feature frequency, discriminative power, and redundancy—and analyze their performance in the context of image classification and object localization. We show that each task has different requirements, and indicate which selection techniques are the most appropriate. Experimental results for recognition on challenging large datasets demonstrate the performance of the approach.



# SÉLECTION DE RÉGIONS SIGNIFICATIVES LOCALES ET DE LEURS DESCRIPTEURS POUR LA RECONNAISSANCE DE CLASSES GÉNÉRIQUES D'OBJETS

Gyuri DORKÓ

Institut National Polytechnique de Grenoble, 9 June 2006

La catégorisation d'objets est l'un des problèmes les plus difficiles en vision par ordinateur. Le but est de reconnaître des objets visuels malgré des variations intra-classe, des changements de point de vue et un fort bruit de fond. L'objectif de cette thèse est d'investiguer un descripteur local d'image et une méthode de sélection de caractéristiques discriminatives. Nous montrons que des descripteurs discriminatifs invariants par échelle donnent d'excellent résultats en catégorisation et en localisation d'objet. Des solutions sont apportées aux deux problèmes fondamentaux suivants: (i) nous améliorons la qualité de la description des images grâce à un nouveau détecteur de points d'intérêts invariant par échelle et (ii) nous intégrons des techniques de filtrage de descripteurs dans nos modèles d'objets.

Notre nouveau détecteur invariant par échelle est basé sur l'idée de "région stable maximale", c'est-à-dire le fait que la position du point d'intérêt est stable même en présence de variations mineures du détecteur. La méthode sélectionne une échelle à partir d'un descripteur local — dans notre cas SIFT — et choisit les régions pour lesquelles la stabilité du descripteur est maximale, c'est-à-dire la différence entre les descripteurs à deux échelles consécutives atteint un minimum. Cette technique de sélection d'échelle est appliquée au détecteur de Harris multi-échelle et les points de Laplace. Des résultats expérimentaux permettent d'évaluer les performances de notre détecteur et montrent qu'il améliore les résultats de mise en correspondance d'image, de classification d'objets et de texture et la localisation d'objets.

Afin de construire des modèles d'objets basés sur des facteurs discriminatifs, les descripteurs invariants par échelle sont classés dans des clusters et donne un ensemble de "mots visuels". Ensuite, nous estimons l'information discriminative contenue dans ces clusters en utilisant différentes techniques de sélection discriminatives — Plusieurs d'entre elles sont traditionnellement utilisées en recherche d'information textuelle. Nous discutons leurs propriétés — fréquence, pouvoir discriminatif et redondance — et analysons leur performances dans le contexte de classification et de localisation d'objet. Nous montrons que chaque tâche a ses particularités et indiquons quelle technique de sélection est la plus appropriée. Des résultats expérimentaux de reconnaissance d'objets sur des jeux de données difficiles montrent les bonnes performances de la méthodologie proposée.



## ACKNOWLEDGEMENTS

I would like to thank all people that have contributed to the completion of this thesis. My sincerest thanks go to my advisor Cordelia Schmid for her guidance, many suggestions, original ideas, feedbacks, and helpful criticism throughout this thesis. I am grateful to Prof. Bernt Schiele and Prof. Andrew Zisserman for their interest in my work, for being the reporters of this dissertation, and also to Prof. Roger Mohr and Tinne Tuytelaars for being the co-examiners at my defense.

I would also like to thank Bill Triggs, Frédéric Jurie, and my friend Guillaume Bouchard for their many useful comments and discussions that helped me understand the sometimes difficult corners of computer vision and machine learning.

I am grateful to my fellow researchers from the LEAR group, Eric Nowak, Navneet Dalal, Ankur Agarwal, Jianguo Zhang, Diane Larlus-Larrondo, Peter Carbonetto, Caroline Pantofaru, Marcin Marszałek, and Joost Van de Weijer, for their support, and for making INRIA a fun and motivating place to work.

I would like to thank the support for the European project LAVA (IST-2001-34405), including all the partners, and the European PASCAL network of excellence.

I am thankful for all researcher that have contributed by making their code available to help my research, especially for Prof. David Lowe, Krystian Mikolajczyk, Michael Sdika, and Matthijs Douze. I am also grateful for Barbara Caputo, Prof. Dietrich Paulus, Prof. Laszlo Csink, Laszlo Kutor, and Mária Dudás, without whom I would not have started my PhD.

My special thanks goes to my friends Stan, Marlen, Carla, and Bram, for the many joyful moments in Grenoble, as well as to my Hungarian friends Kriszta, Andi, and Gábor who have not forgotten about me even that I am being so far from home.

Last, but not least, I would like to thank my family for their love, emotional support, and encouragement. Without them, I would not have made it.



---

---

# Table of Contents

---

<b>1</b>	<b>Introduction</b>	<b>13</b>
1.1	Context . . . . .	13
1.2	Our Approach . . . . .	14
1.3	Contributions . . . . .	16
1.4	Applications . . . . .	17
1.5	Overview . . . . .	19
<b>2</b>	<b>Local Image Representation</b>	<b>21</b>
2.1	Background . . . . .	24
2.1.1	Interest Point Detectors . . . . .	24
2.1.2	Local Description: Scale-Invariant Feature Transform . . . . .	29
2.2	Scale Selection by Maximally Stable Local Description . . . . .	30
2.3	Evaluation for image matching . . . . .	34
2.3.1	Viewpoint Changes . . . . .	36
2.3.2	Changes in Illumination . . . . .	39
2.3.3	Overall Performance . . . . .	39
2.4	Evaluation for image categorization . . . . .	41
2.5	Implementation Details . . . . .	44
2.6	Conclusions . . . . .	47
<b>3</b>	<b>Discriminative Feature Selection for Object Class Appearance</b>	<b>49</b>
3.1	Probabilistic Interpretation . . . . .	52
3.2	Feature Scoring Techniques . . . . .	54
3.3	Selection for Local Features . . . . .	65
3.3.1	Visual Words . . . . .	65
3.3.2	Retrieving Object Features . . . . .	67
3.4	Discussion . . . . .	75



<b>4</b>	<b>Classification and Localization of Object Classes</b>	<b>79</b>
4.1	Object Class Classification with Discriminative Features . . . . .	81
4.1.1	Classifier for Objects Presence . . . . .	82
4.1.2	Experimental Set-Up . . . . .	83
4.1.3	Experiments: Image classification . . . . .	85
4.2	Object Localization with Discriminative Features . . . . .	91
4.2.1	The Localization Approach . . . . .	91
4.2.2	Evaluation of Different Parameters . . . . .	95
4.2.3	Additional Results: PASCAL Challenge, Butterfiles . . . . .	101
4.3	Implementation Details . . . . .	104
4.4	Discussion . . . . .	107
<b>5</b>	<b>Conclusion and Future Work</b>	<b>109</b>
	<b>Appendix: Influence of the number of interest points</b>	<b>115</b>

---

# Introduction

---

OBJECT recognition is a challenge that computer vision researchers, psychologists and researchers from other fields have been trying to understand for more than 40 years. After many years of research artificial vision is still far behind human vision. People are able to see, to recognize, and to categorize objects in the world. However, for computers this is not an easy task. The ability, for example, to see a chair from all different viewpoints and to understand and know that it is the same chair are extremely complicated tasks. The 2-D appearance of the same object can be very different when the viewpoint changes. Furthermore, due to our generalization capability, people are capable of finding a chair, even if they have not seen that particular instance before. Creating categories, finding shared properties, generalizing appearance are challenging tasks for computers, mainly due to a potentially high intra-class variance across object instances.

## 1.1 Context

While object recognition is a large field, in this thesis we focus on visual object class categorization and localization. Figure 1.1 illustrates some of the difficulties of recognizing object categories. *Intra-class variations* among instances of a class is only one



Figure 1.1: Five different bicycles illustrate the challenge for object class recognition. Different viewpoints, occlusion, noise, and cluttered background make it hard to recognize the objects. Intra-class variation (shape and color) across the different bicycles challenges the generalization capabilities of computer vision systems.



Figure 1.2: Examples of wildcats.

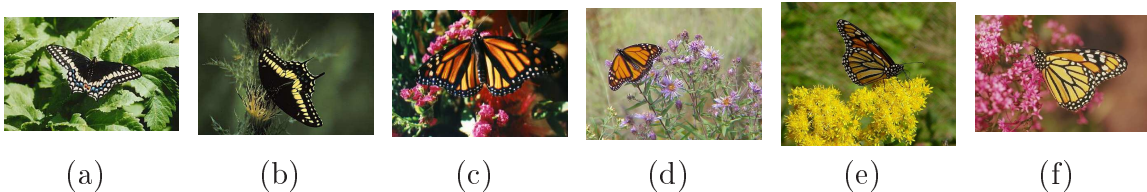


Figure 1.3: Examples of butterflies.

of the challenges: object parts can have different geometrical structure, color or can be completely missing. In Figure 1.1 bicycles (a) and (e) are different in color, while bicycle (b) has different geometrical proportions. Many applications require objects to be found in predefined pose and orientation, such as recognizing profiles of faces, or side-views of cars. Others, like the bicycle example, are less restricted and therefore more difficult: bicycles (d) and (e) are viewed from *different viewpoints*, and (a) and (b) are imaged at *different scales* (magnification). Robustness to *occlusions* and missing parts are usually additional requirements for state-of-the-art applications; e.g., bicycle (a) has a missing (covered) seat. Occlusions may be caused by the environment, or even by the object itself: the spokes of the first tire are occluded on (d). Everyday objects, such as bicycles, often appear together with other objects or on *cluttered background*. This additional data, so called *context*, can distract our system and needs in general to be discarded. Note that it can also help to recognize the object class. An example is a traffic control system detecting cars. In such a system the recognition of roads is probably useless because they occur in all images. However, the shadow of the car (on the road) is probably a useful discovery.

## 1.2 Our Approach

Instances of an object category often share some visual appearance, and our main goal is to find these common features. The examples in Figure 1.2 and Figure 1.3 show two different object categories. The selection of common discriminative object parts is relatively easy, because almost any set of features (of adequate size) separates wildcats from butterflies. However, if Figure 1.2 itself are defined to contain two categories—cheetahs (a),(b), and jaguars (c),(d)—discriminative features are much

harder to find. Furthermore, if we assume that examples in Figure 1.3 are from two categories, then butterfly experts would immediately notice that (a) and (b) are *black swallowtails*, while (c)-(f) are *monarchs*. Those who have less experience with insects would probably say that (a)-(d) are open while (e)-(f) are closed butterflies. So we see that common features are not always discriminative, and according to the task the useful features are different. To discover discriminative object parts we use

- **local** or semi-local **representations of images** to describe object parts,
- a way to measure their usefulness, and *select discriminative features*.

Sparse **local representations** are typically computed on a set of interest point locations. Their aim is to describe the regions by keeping distinctive information, and at the same time providing robustness to small translations and noise. Local representation of images offer a solution to deal with occlusion and cluttered background: individual descriptors only store information of the local content, and therefore they are not distracted by other parts of the image. The influential work of Schmid and Mohr (1997) is the first that uses **interest points** for content based object recognition. Interest points are automatically detected image locations, such as corners or centers of blobs. They allow to create a sparse local representation of images by selecting regions which keep distinctive information, and at the same time provide robustness to small translations and noise. In the last few years these points became invariant to various image transformations, like changes in viewpoint and scale. At the time of writing at least a dozen of these detectors exist all selecting regions by different criteria. The combination of interest points detectors and local descriptors allows *sparse and robust representation of object, scenes, or textures*. Rotated objects, scenes from different viewpoints or with illumination changes are challenges that can be solved already at representation level, i.e., there is no need to learn those by examples.

State of the art methods provide relatively good solutions for recognizing specific objects, such as a given bicycle or car, by matching local appearance. However, detection of object categories requires additional *generalization* capabilities to deal with intra-class variability. **Discriminative feature selection** methods can guide object recognition to find category-discriminative object parts and to discard unnecessary background features. These methods are recent tools in computer vision adopted from the text literature. Local representation of images and standard learning techniques, such as vector quantization, have built a bridge between computer vision and text recognition. Our images become visual documents and the quantized local descriptors became visual words. Owing to a huge availability of documents, the text community has early realized the need for discriminative feature selection. For example, to index news directories or web pages, relevant information has to be selected to train classifiers to recognize different categories. In the last few years, the growing number of examples (Internet) directed researchers to improve classification efficiency and accuracy. One essential topic of this research is feature selection. In this thesis we apply

these techniques to computer vision. In object category recognition, local representation and feature selection together help to develop high performance automatic tools for object and texture recognition, categorization and detection, for scene analysis, and for image indexing.

### 1.3 Contributions

In this thesis we discuss and offer solutions for recent problems of image representation and object detection. The key contributions are the following:

#### **Interest Point Detection by Maximally Stable Local Image Representation**

Many interest point detectors and local descriptors have been developed during the last few years. Their quality depends on the task. For example, some perform well for image matching while others are better for object recognition. Their behavior can be explained by the different ways they select image regions and incorporate various feature properties. As an examples, image classification or image retrieval may only match the local regions purely by appearance, i.e., ignoring their scales, locations, and spatial organization. For other applications, such as image matching or camera calibration, these properties are very important, and many times their estimation is unstable or noisy. Consequently, the quality of interest point detectors is not straightforward to measure, since different methods should be used depending on the context. Our experience has shown that one of the weakest properties of scale-invariant detectors is the scale estimation. This thesis proposes a novel method to determine (select) the characteristic scales for interest point detectors. Our idea is to use an appropriately chosen descriptor to select regions for which this descriptor is maximally stable. Experimental results show that our new criterion improves performance for image matching in challenging environments, such as variation in illumination conditions. Due to a more stable appearance-based representation, texture categorization on popular sets shows 3 – 10% improvement with the new detectors.

#### **Feature Selection for Local descriptors**

In this thesis we adapt and compare several techniques from the text literature, most of which are new in vision. We analyze several feature properties including feature frequency, i.e., how often a feature appears, discriminative power to separate object from background, and redundancy. Different trade-offs between properties are pointed out, and selection methods are distinguished (grouped) accordingly. By the correct combination of these properties, i.e., by choosing the selection method wisely for a given task, we show how to achieve good recognition performance with many or just a sparse set of features. Our experiments evaluate class-discriminative feature selection for invariant local features.

### Improved Object Class Recognition via Feature Ranking and Selection

We have chosen object category classification and localization to demonstrate the performance of discriminative feature selection. A simple classification framework demonstrates that discovering discriminative features can directly be used for object recognition. Selection methods on different types of features are compared and discussed for three different tasks: Object feature retrieval tries to recall features providing the best object coverage, while keeping the background featureless or very sparse. Appearance-based object classification uses discriminative features to decide about the presence of an object class in images. Object class localization aims to determine the exact position of unseen object instances in test images. For localization we extend an existing state-of-the-art method by incorporating feature ranks. This leads to a faster system with improved performance. We additionally extend the framework for rotation invariant training and detection.

## 1.4 Applications

Advances such as discriminative feature selection and scale-invariant local representations, discussed in this thesis, help to analyze and improve state-of-the-art image representation and object recognition techniques. In the following we list a few examples among a wide range of possible applications.

### Surveillance and Security

One of the most useful applications of object recognition are surveillance systems. Recent security systems based on photography or CCTV (Closed Circuit Television) use computer vision to match digital images taken from cameras with images stored in a database. Discriminative feature selection may help to determine an important subset of features in advance, and therefore increase the system quality and performance.

### Manufacturing Processes and Quality Control

Improved feature extraction and local description of images can help industrial application to support manufacturing processes. Many quality control methods employ computer vision. They are based on statistical analysis of detected features, and aim to reduce the amount of faulty products, in order to meet customer requirements.

### Autonomous Vehicles

Even though autonomous driving cars are not yet available for the market, manufacturers have already demonstrated preliminary prototypes and driving systems. Learning and rapid discovery of useful features, such as parts of other cars or obstacles, can guide or help the drivers increasing their safety. UAVs (unmanned aerial vehicles) first

were used for surveillance, and nowadays, almost all major military have them. They are also used to monitor traffic, detect certain events, such as forest fires. Robust local image representation and focus of attention mechanism (feature selection) help those vehicles for better motion planning, navigation, scene analysis (to detect where it is), or improved SLAM techniques<sup>1</sup>.

## Web Search and Content Based Image Retrieval

Did you know that the verb GOOGLE<sup>2</sup> has been added to the New Oxford American Dictionary? The Internet search engines have become a part of our everyday life. Researchers from the text domain have implemented discriminative feature selection so successfully that search engines generate around 85% of the total web traffic. Now it is our turn to index images. Many recent search engines, such as Google, MSN, Lycos, Yahoo, Altavista, and A9 support search for images. However their algorithm is based on purely textual information, such as filenames, image meta-data, and surrounding HTML content. While many times this is sufficient, indexing by image content would improve current performance, as well as open new possibilities:

- visual similarity between images helps to reject incorrect matches, and increase the recall by discovering new correct ones,
- queries can be based on images instead of text; e.g., we can look for a certain car by its picture, or find our copyright protected images and identify fraud,
- given an image or images of someone or something, e.g., a famous building or an actress, we can recover its identity, such as its place and name,
- mixed text and image queries can provide a richer way of looking for information.

In order to efficiently index and rank images, the correct features have to be generated and selected. Discriminative feature selection may help to develop domain specific search engines, as well as to find the most informative features in general.

## Video Indexing

Digital videos are now available not only for professionals but also for everyday people. DVD players and recorders, recent digital cameras, and high speed Internet connections made indexing for videos as important as for images. Videos can be seen as a sequence of images, and therefore many techniques from images can be applied without

---

<sup>1</sup>In Simultaneous Localization And Mapping (SLAM), the quality of the iteratively built map can be refined and therefore improved by matching discriminative local features over time.

<sup>2</sup>goo·gle |ˈɡoʊɡəl| (also Goo·gle) · verb informal [ intrans. ] use an Internet search engine, particularly Google.com: *she spent the afternoon googling aimlessly*. · [ trans. ] search for the name of (someone) on the Internet to find out information about them: *you meet someone, swap numbers, fix a date, then Google them through 1,346,966,000 Web pages*. ORIGIN: from Google, the proprietary name of a popular Internet search engine.

major modification. However, adding temporal information to the feature space opens new perspectives, such as searching for certain actions. Presently only preliminary versions of video web search are available on major sites (Google, Yahoo, Altavista, A9) and similarly to images, their indices are built on textual information only. Discriminative feature selection could help to build domain specific search, e.g., looking for the appearance of an actor in a movie, or to determine the difference between actions. Scene analysis can guide professionals when editing movies, or can identify viewers preferences (e.g., improve TiVo suggestions).

## 1.5 Overview

The manuscript is organized as follows. Chapter 2 introduces a sparse local image representation with interest point detectors and local descriptors. In Section 2.2 we describe our new scale selection method. Evaluation and comparison with existing techniques are carried out for image matching (Section 2.3), object and texture classification (Section 2.4 and Section 4.1.3), and object localization (Section 4.2.2).

Chapter 3 introduces different selection and ranking techniques. In Section 3.3 we build the link between image representation and features by creating *visual words*, and experimentally compare the introduced selection techniques for object feature retrieval. Chapter 4 integrates feature selection into a framework for object recognition. First we show an application to recognize the presence or absence of objects in images (image classification), and compare the results of different features and selection methods. In Section 4.2 we show how to improve object localization by class-discriminative feature ranking.





---

# Local Image Representation

## Scale Selection via Maximally Stable Local Description

---

LOCAL photometric descriptors computed at keypoints have demonstrated excellent results in many vision applications, including object recognition (Fergus *et al.*, 2003; Opelt *et al.*, 2004), image matching (Schaffalitzky and Zisserman, 2002), and sparse texture representation (Lazebnik *et al.*, 2003). Recent work has concentrated on making these descriptors invariant to image transformations. This requires constructing invariant image regions which are then used as support regions to compute invariant descriptors. In most cases a detected region is described by an independently chosen descriptor. It would, however, be advantageous to use a description adapted to the region. For example, for blob-like detectors which extract regions surrounded by edges, a natural choice would be a descriptor based on those edges. However, those adapted representations may not provide enough discriminative information for the region, and consequently, a general purpose descriptor (e.g. wavelets, shape-context, SIFT, etc.) might be a better choice. Many times this leads to better performance, yet less stable representations: small changes in scale or location can alter the descriptors significantly. Our experiments have shown that the most sensitive component of keypoint-based scale-invariant detectors is the scale selection. This motivated us to develop a novel detector which uses the descriptor chosen for the given task to select the characteristic scales. Our feature detection approach consists of two steps. We first apply an interest point detector on multiple scales to determine informative and repeatable locations. For each position we then apply a scale selection algorithm to identify maximally stable representations, i.e., a scale for which a local descriptor is the most stable. The local description can be any measure that can be computed on a pixel neighborhood, such as color histograms, steerable filters and wavelets. For our experiments we chose the Scale-Invariant Feature Transform (SIFT) (Lowe, 2004), which has proven excellent performance for object representation and image matching (Mikolajczyk and Schmid, 2004a).

Our new method for scale-invariant keypoint detection and image representation has the following properties:

- Our scale selection method guarantees more stable descriptors than state-of-the-art techniques by explicitly using descriptors during keypoint detection. The stability criterion is developed to minimize the variation of the descriptor for a small change in scale.
- Repeatable locations are provided by interest point detectors (e.g. Harris), and therefore they have rich and salient neighborhoods. This consequently helps to choose repeatable and characteristic scales. We verify this experimentally, and show that our selection competes favorably with the best available detectors.
- The detector takes advantage of the properties of the local descriptor. This can include invariance to illumination or rotation as well as robustness to noise. Our experiments show that the local invariant image representation extracted by our algorithm leads to significant improvement for object and texture recognition.

## Related Work

For selecting local invariant regions, many different scale- and affine-invariant detectors exist in the literature. Harris-Laplace (Mikolajczyk and Schmid, 2004b) detects multi-scale keypoint locations with the Harris detector (Harris and Stephens, 1988) and the characteristic scales are then determined by the Laplacian operator. Locations based on Harris points are very accurate. However, scale estimation is often unstable on corner-like structures, because it depends on the exact corner location, i.e., shifts by one pixel may modify the selected scale significantly. The scale-invariant Laplacian detector (Lindeberg and Garding, 1994) (LoG) selects the extremal values in location-scale space. The Difference of Gaussian (DoG) detector developed by Lowe (2004) approximates the Laplacian, and therefore it similarly selects scale-space maxima to find blob-like structures. Blobs are well localized structures, but due to their homogeneity, the information content is often poor in the center of the region. Triggs' detector (Triggs, 2004) extends the Förstner-Harris approach to general motion models and robust template matching by finding regions which can be accurately self-matched under various similarity or affine transformations. This detector extracts fewer but very stable keypoints. For instance, the rotation invariant detection rejects point-like structures, since they cannot be well-localized (self-matched) under image rotation, i.e., they have no characteristic orientation. The method of Kadir *et al.* (2004) extracts circular or elliptical regions in the image as maxima of the entropy scale-space of region histograms. This is also a blob detector, but has been shown to provide a more robust appearance based representation for some object categories (Kadir *et al.*, 2004). Mikolajczyk *et al.* (2005b) showed that it performs poorly for image matching, which might be due to the sparsity of their scale quantization. Presumably performance issues prohibit them for more extensive search in scale-space. The Intensity-Based Region detector (Tuytelaars and Van Gool, 2004) selects multi-scale locations at extremal intensity values and determines the corresponding neighborhood by discovering sudden

nearby intensity changes. The edge-based region detector (Tuytelaars and Van Gool, 2004) finds quadrangular segments with a corner detected by the multi-scale Harris operator and sides determined by near edges. The object-part detector of Jurie *et al.* (Jurie and Schmid, 2004) selects circular regions with the most salient convex arrangement of local edges extracted by the Canny-Deriche operator. Since the detected regions are surrounded by edges, they proposed a local image representation based on this structure. These descriptors are however not as discriminative as other available representations, since it only encodes information of the surrounding edges. Due to the homogeneity of the selected regions it suffers from the same problems as other blob-like methods. The Maximally Stable Extremal Regions (MSER) detector (Matas *et al.*, 2002) defines extremal regions as image segments where each inner-pixel intensity value is less/greater than a certain threshold  $t$ , and all intensities around the boundary are greater/less than the same  $t$ . An extremal region is *maximally stable* when the area (or the boundary length) of the segment changes the least with respect to  $t$ . This detector works particularly well on images with well defined edges, but is less robust to noise and not adapted to texture-like structures. It usually selects relatively few regions.

Viewpoint invariance is sometimes required to achieve reliable image matching, object or texture recognition. Affine-invariant detectors (Kadir *et al.*, 2004; Matas *et al.*, 2002; Mikolajczyk and Schmid, 2004b; Tuytelaars and Van Gool, 2004) explicitly estimate the affine shape of the regions to allow pre-normalization of the patch prior to the descriptor computation. The affine extension of Harris-Laplace (Mikolajczyk and Schmid, 2004b) is similar to the one first used by Lindeberg and Garding (1997) for shape-from-texture. It applies the affine kernel only to fixed points to reduce the complexity of the entire affine-space. This is one of the most widely used approaches; Lazebnik *et al.* (2003) use a similar technique for the LoG detector to perform texture classification under affine transformations. However, note, that their adaptation procedure is a post-processing step of the scale-invariant detection based on the scatter matrix of image gradients at keypoint locations.

Mikolajczyk *et al.* (2005b) evaluated several affine-invariant detectors. MSER (Matas *et al.*, 2002) performed best, closely followed by Hessian- and Harris-Laplace. Moreels and Perona (2005) also find that Harris- and Hessian-Laplace perform best for object recognition. Their study shows poor performance of the MSER detector for 3D environments. Mikolajczyk *et al.* (2005a) experimentally compared the performance of recently proposed detectors and descriptors for category recognition, and found Hessian-Laplace (Mikolajczyk and Schmid, 2004b) and the entropy detector (Kadir *et al.*, 2004) to be the most suitable.

## Overview

This chapter is organized as follows. In Section 2.1 we present the interest point detectors and local descriptors that are used in this chapter. Section 2.2 presents our

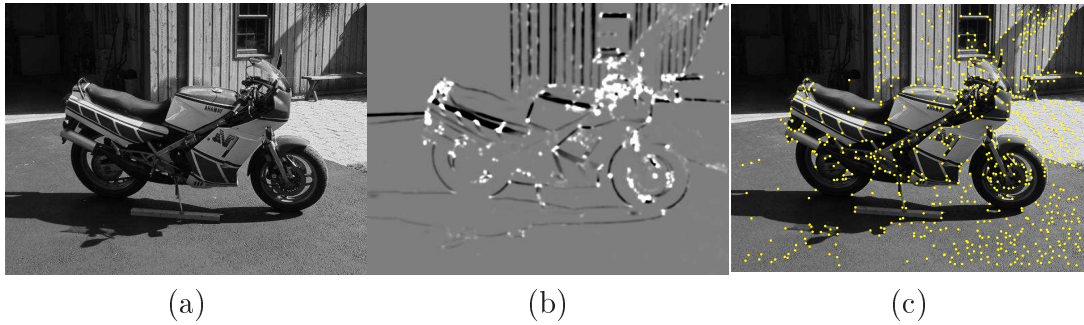


Figure 2.1: Harris corner detection. (a) the original image, (b) the Harris image, (c) the local maxima of the Harris image marked on the original image.

new scale selection technique *Maximally Stable Local SIFT Description* and introduces two new detectors, Harris-MSLSD and Laplacian-MSLSD. We then compare their performance to Harris-Laplace and the Laplacian detectors. In Section 2.3 we evaluate the performance for image matching using a publicly available framework. Section 2.4 reports results for object-category and texture classification. Finally, in Section 2.6 we conclude.

## 2.1 Background

This section provides a detailed description of the interest point detectors of (Mikolajczyk and Schmid, 2004b; Lowe, 2004; Triggs, 2004; Lindeberg, 1998; Matas *et al.*, 2002), and the Scale-Invariant Feature Transform descriptor (Lowe, 2004). Our aim is not to cover the full theory of scale-invariant detectors and local representation, but to provide sufficient background information for the techniques that are used later in this chapter. Our experiments will compare our scale selection to several existing techniques in the literature.

### 2.1.1 Interest Point Detectors

#### Harris Points — a corner detector

The *scatter matrix* (or second moment matrix) of local image gradients,  $\int \nabla \mathbf{I}^T \nabla \mathbf{I} dx$ , is often used for feature detection, and it is given as

$$\mu(\mathbf{x}, \sigma_I, \sigma_D) = \sigma_D^2 g(\sigma_I) * \begin{bmatrix} \mathbf{I}_x^2(\mathbf{x}, \sigma_D) & \mathbf{I}_x \mathbf{I}_y(\mathbf{x}, \sigma_D) \\ \mathbf{I}_x \mathbf{I}_y(\mathbf{x}, \sigma_D) & \mathbf{I}_y^2(\mathbf{x}, \sigma_D) \end{bmatrix}. \quad (2.1)$$

Image derivatives  $\mathbf{I}_x$  and  $\mathbf{I}_y$  are computed by convolution of Gaussian filters with scale  $\sigma_D$  (derivation scale), and locally averaged by Gaussian smoothing with scale  $\sigma_I$  (integration scale). The eigenvalues of this matrix represent the two principal

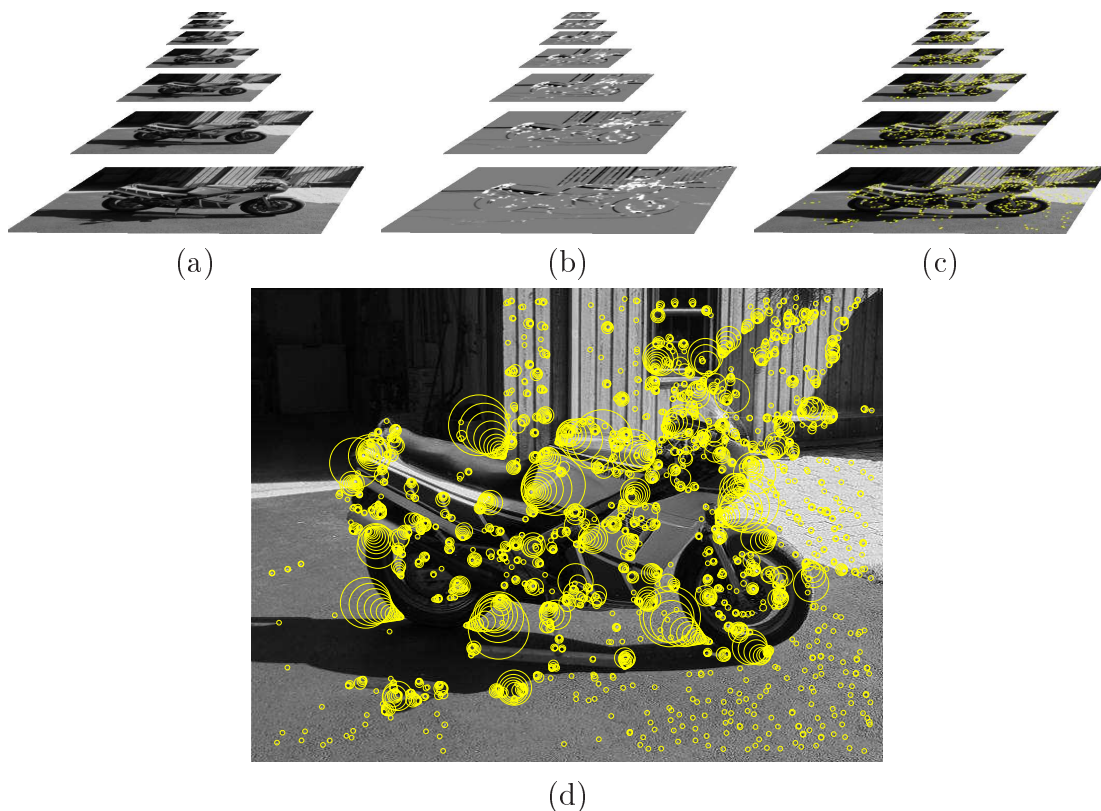


Figure 2.2: Extraction of multi-scale Harris points. (a) shows the multi-scale image pyramid, (b) the computed Harris images at each scale, and (c) the image pyramid with the multi-scale Harris points. (d) shows the detections projected back to the original image. The radii of the circles correspond to the scale  $(2\sigma)$ .

curvatures of a point  $\mathbf{x}$ . Corner-like structures can be extracted at points where both of these curvatures are significant in orthogonal directions. The Harris detector (Harris and Stephens, 1988) is based on this principle. The Harris *cornerness* combines the determinant and trace of this matrix and defined by

$$\det(\mu(\mathbf{x}, \sigma_I, \sigma_D)) - \alpha \text{trace}^2(\mu(\mathbf{x}, \sigma_I, \sigma_D)). \quad (2.2)$$

The keypoints are determined as local maxima of this value. Figure 2.1 shows a Harris image, i.e., the *cornerness* for each point, and the keypoints on an example image. Schmid *et al.* (2000) show that the Harris detector is superior to other methods (Cottier, 1994; Heitger *et al.*, 1992; Horaud *et al.*, 1990).

### Multi-Scale Interest Points

A multi-scale representation of images is crucial for many applications. A typical example is matching scenes or objects with different scales. Many state-of-the-art methods

are based on the Gaussian kernel. A multi-scale representation consists of a set of images at different discrete levels of scale (Witkin, 1983). Koenderink (1984) showed that scale-space satisfies the diffusion equation for which the solution is a convolution with a unique Gaussian kernel (Babaud *et al.*, 1986; Lindeberg, 1990; Florack *et al.*, 1992). Images on coarse scales are obtained by smoothing images on finer scales with an appropriate Gaussian kernel. An implementation can sample the coarser scale image by the corresponding scale factor to accelerate the computation and this representation is often referred as the scale-space image pyramid.

When an interest point operator is applied on multiple scales we call the detections *multi-scale interest points*. Even though they are called *points*, they can be interpreted as regions—points and their neighborhood—as they are parameterized by a location  $\mathbf{x}$ , and a scale  $\sigma$ .<sup>1</sup> As for the Harris operator, Dufournaud *et al.* (2000) proposed a scale adaptive extension, where the points are detected at the local maxima of the Harris images computed at different scales. Figure 2.2 illustrates the multi-scale Harris interest points. Figure 2.2(a) shows the original image pyramid, and (b) the corresponding Harris images. Figure 2.2(c) marks the detections, i.e., the maxima of (b) on the original images (a), and finally on (d) we show all the detections with circles corresponding to the detection scale. Note, that for illustration purposes, we omit some scale levels from the pyramids (a), (b), and (c).

### Scale-Invariant Interest Points

Instead of extracting interest points for every scale level, automatic scale-selection techniques determine one or a few characteristic scales at each location. These detections are called scale-invariant interest points because they mark the same points  $(\mathbf{x}, \sigma)$  on images taken at different resolutions. There are two main advantages of selecting scales. First, the number of interest points is reduced by *intelligent* rejection of unnecessary scales, and second, the scale becomes a new characteristic property of the detection. Many applications, such as the one in Section 4.2, rely on this property to perform scale-invariant learning and recognition.

One of the first scale-invariant interest point detectors is the Laplacian-of-Gaussian (*LoG*) developed by Lindeberg (1998). It is based on the Gaussian scale-space (successive smoothing with Gaussian kernels), and it selects 3D local extrema of the Laplacian filtered images. Detections are obtained on blob-like image structures. Figure 2.3(b) shows an example detection of LoG. To demonstrate the multi-scale behavior, i.e., LoG without scale selection, Figure 2.3(a) shows the local extrema of the Laplacian

---

<sup>1</sup>In several multi-scale detectors that are based on second moment matrix computations, we distinguish between two scale parameters, the derivation scale ( $\sigma_D$ ) and the integration scale ( $\sigma_I$ ) (cf. Section 2.1.1). Usually, a constant factor is used between  $\sigma_D$  and  $\sigma_I$  to balance the size of the area used to calculate the statistics of local gradient variations.

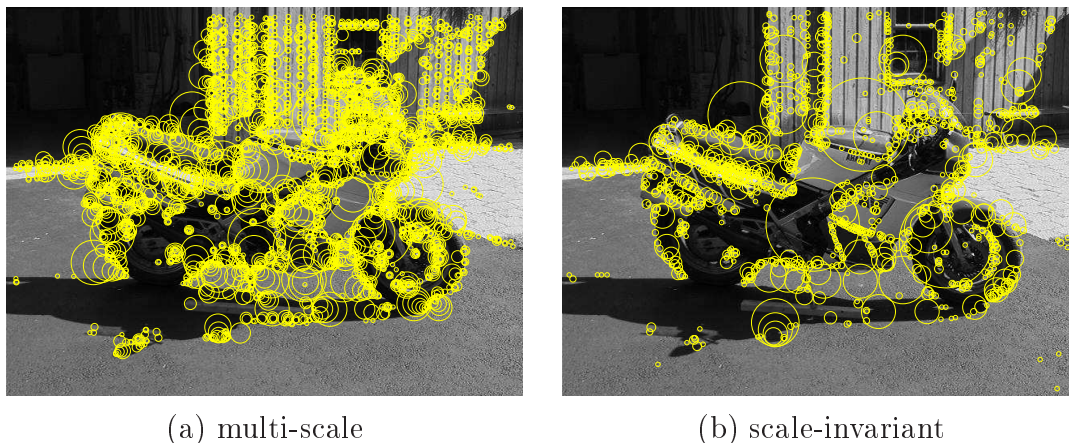


Figure 2.3: The LoG detector. (a) shows all extrema of the 2D LoG function on multiple scales. (b) LoG 3D maxima in location-scale space. Note for illustration purposes we omit some scales from (a).

on each scale. As before, the radii of the circles indicate the scale. We can observe that while the LoG (Figure 2.3(b)) detector selects only blob-like features, the 2D LoG maxima (Figure 2.3(a)) includes also detections near corners and edges.

Mikolajczyk and Schmid (2001) evaluate different scale selection criteria for scale-invariant image matching environments. Apart from the Laplacian they study the squared image gradients, the Difference-of-Gaussians (Lowe, 2004) (the difference of the Gaussian filter responses between two consecutive scales), and the Harris function (2.2). Their evaluation shows that the Laplacian function selects the highest percentage of correct characteristic scales, and as a result they introduce the scale-invariant *Harris-Laplace (H-Lap)* detector, which combines the stable Harris detector with the Laplacian scale-selection. Unfortunately, their evaluation of scale selection functions are carried out in general, i.e., for each pixel in the image. While it is a reasonable assumption to transfer the results to Harris points, they did not verify the quality of scale selection specifically on keypoint locations. Even though, they did not search for the Harris maxima in scale space, we find it interesting to investigate the Harris scale selection on Harris points, and include the *Harris-Harris (H-Har)* detector in our experiments.

Triggs (2004) generalizes the Förstner-Harris approach to general motion models and offers a new characteristic scale selection technique. Including scale as a (non-translational) motion parameter forces the detections to be accurately self-matched not only in location but also in scale-space. Since this is a more generalized Harris detector, we call it *Harris-Gen (H-Gen)* in our experiments. Notice the difference between *Harris-Harris* and *Harris-Gen*. The former computes the 2D Harris images for stable locations and chooses the maxima of cornerness in scale-space, while *Harris-Gen* optimizes the Harris keypoints for matching precision in higher dimensional (not only translational)



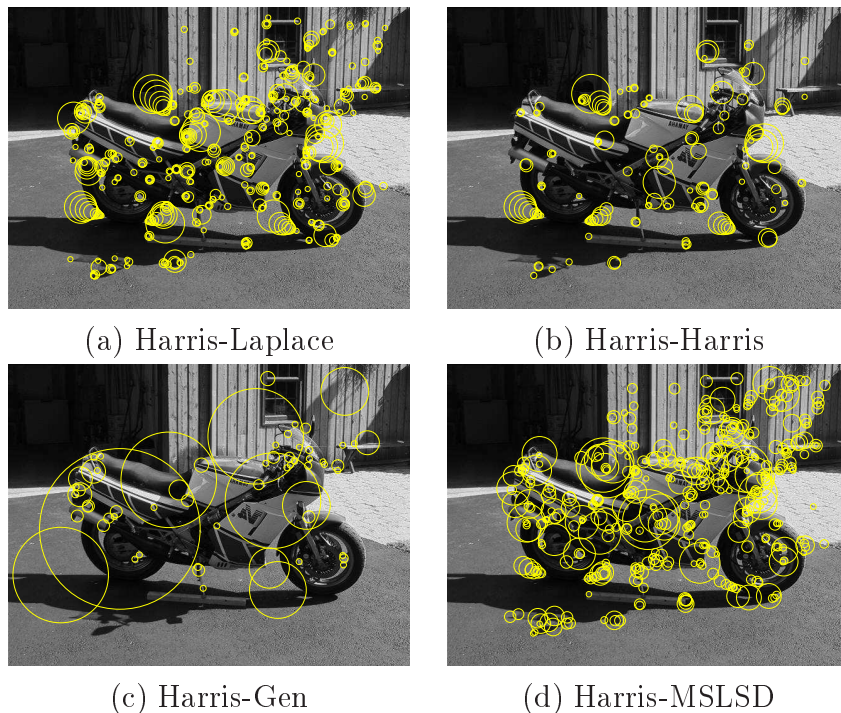


Figure 2.4: Scale-Invariant Harris points. The example shows the points with their characteristic scales for each scale selection method. For illustration we omitted detections with  $\sigma < 2$ .

space. In our experiments Harris-Gen is used with rotation stability enabled, so the motion model actually includes 4 parameters<sup>2</sup> (location+scale+rotation). Example detections for the various Harris-based detectors can be found in Figure 2.4. Figure 2.4 (d) also shows results of our scale selection approach introduced in Section 2.2.

Maximally Stable Extremal Regions (*MSER*) (Matas *et al.*, 2002) directly optimizes the region shape for stability. The algorithm determines a small subset of all regions, the so-called extremal regions, where each inner-pixel intensity value is less/greater than a certain threshold  $t$ , and all intensities around the boundary is greater/less than  $t$ . Among these extremal regions they select the ones that are the most stable in shape. Stability is measured by the change in region area (or boundary length) with respect to  $t$ . The MSER detector has been shown to perform well (Mikolajczyk and Schmid, 2004b) for matching scenes with significant viewpoint changes.

---

<sup>2</sup>In our experiments we do not include other stability properties, e.g., affine transformations, illumination, etc, into H-Gen; the detector is consistently used with the same criteria. Note, that we have tried to add other parameters, but the results were always inferior to using location+scale+rotation.

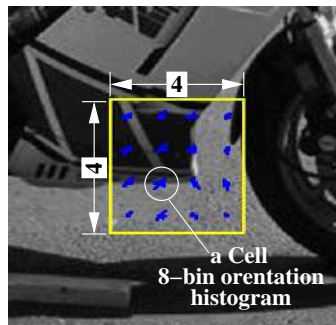


Figure 2.5: The SIFT descriptor computed on a  $4 \times 4$  grid with 8-bin orientation histograms.

### 2.1.2 Local Description: Scale-Invariant Feature Transform

Local image representations are typically a set of vectors computed on image patches at various locations. Possible choices of image descriptors are raw image intensities, color histograms (Swain and Ballard, 1991), wavelets (Grossmann and Morlet, 1984), steerable filters (Freeman and Adelson, 1991), moment invariants (Van Gool *et al.*, 1996), differential invariants (Koenderink and van Doom, 1987), complex filters (Schaffalitzky and Zisserman, 2002), shape context (Belongie *et al.*, 2002), spin images (Lazebnik *et al.*, 2003), scale-invariant feature transform (SIFT) (Lowe, 2004), and its variants (Ke and Sukthankar, 2004; Lazebnik *et al.*, 2005; Mikolajczyk and Schmid, 2004a). Mikolajczyk and Schmid (2004a) compared some of these descriptors and show that SIFT (Lowe, 2004) features performs better than others. Evaluation of Moreels and Perona (2005) also found SIFT and shape-context to perform best for object recognition. Based on their results we always use SIFT as a local image representation.

Figure 2.5 illustrates the computation of SIFT on an image patch centered on keypoint locations ( $\mathbf{x}$ ) and using a window size related to its scale ( $\sigma$ ). The patch is divided by an  $IS \times IS$  grid, where  $IS$  is the *index size*, and is set to 4. For each cell an  $OS$ -bin histogram of local orientations (weighted by the gradient magnitudes) is computed ( $OS = 8$ ), leading to a concatenated,  $4 * 4 * 8 = 128$  dimensional real vector. These parameters were suggested by Lowe (2004), and are fixed for our experiments. For robust description, histograms are computed with a Gaussian weighting function ( $\sigma = \text{half window size}$ ) and a trilinear interpolation is used to distribute the value of each gradient sample into adjacent histogram bins (each orientation falls to  $2^3 = 8$  bins). The SIFT descriptor is normalized to unit length, providing invariance to scalar changes in image contrast. Since the descriptor is based on gradients, it is also invariant to additive constant changes in brightness. SIFT was originally proposed to be rotation invariant, which is achieved by an efficient dominant gradient computation, which can directly be used to normalize the gradients for the orientation histograms.

Practically, many times scale-invariant interest point detections are followed by a normalization to obtain a *regular region* before the computation of the descriptors. This may include an elliptical or an irregular shape normalization to unit square or a rotation of patches to a pre-computed characteristic orientation. In our experiments we also follow this principle, however, rotation invariance is only applied when indicated, i.e., in general the SIFT descriptors are computed in a non-rotation invariant way.

## 2.2 Scale Selection by Maximally Stable Local Description

In this section we propose a new method for selecting characteristic scales for keypoint detectors and discuss the advantages and properties of the new approach. We address two key features of interest point detectors: repeatability and description stability. *Repeatability* determines how well the detector selects the same region under various image transformations, and is important for image matching. In practice, due to noise and object variations, the corresponding regions are never exactly the same but their underlying descriptions are expected to be similar. This is what we call the *description stability*, and it is important for image representation and appearance based recognition.

The two properties, *repeatability* and *descriptor stability*, are in theory contradictory. A homogeneous region provides the most stable description, whereas its shape is in general not stable. On the other hand, if the region shape is stable, for example using edges as region boundaries, small errors in localization will often cause significant changes of the descriptor. Our solution is to apply the Maximally Stable Local Description algorithm to interest point locations only. These points have repeatable locations and informative neighborhoods. Our algorithm adjusts their scale parameters to stabilize the descriptions and rejects locations where the required stability cannot be achieved. The combination of repeatable location selection and descriptor stabilized scale selection provides a balanced solution. In Section 2.3 we show that our new method provide comparable performance to Harris-Laplace and LoG for image matching. Moreover, due to additional robustness (which is discussed later in this section) they outperform their counterparts.

### Scale-invariant MSLSD detectors

To select characteristic locations with high repeatability we first apply an interest point detector at multiple scales. We chose two widely used complementary methods, Harris (Harris and Stephens, 1988) and the Laplacian (Blstein and Ahuja, 1989; Lindeberg, 1998) detectors. The second step of our approach selects the characteristic scales for each keypoint location. We use *description stability* as criterion for scale selection: the scale for each location is chosen such that the corresponding representation (in our case SIFT (Lowe, 2004)) *changes the least* with respect to scale. Figure 2.6 illustrates our selection method for two Harris points. The two graphs show how the

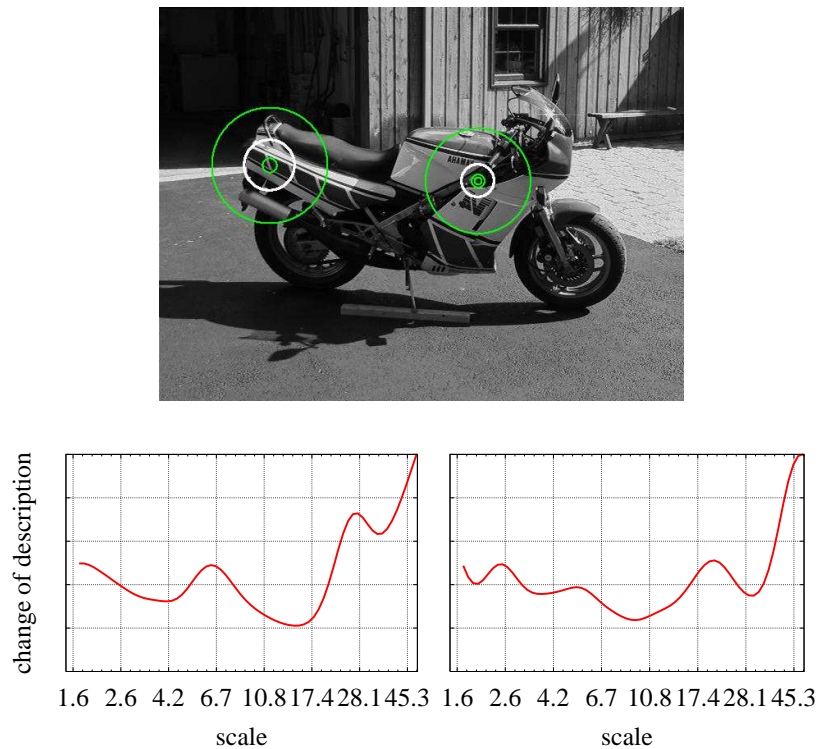


Figure 2.6: Two examples of scale selection. The left and right graphs show the change of the local description as a function of scale for the left and right points respectively. The scales for which the functions have local minima are shown in the image. The bright thick circles corresponds to the global minima.

descriptors change as we increase the scale (the radius of the region) for the two keypoints. To measure the difference between SIFT descriptions we use the Euclidean distance as in (Lowe, 2004). The minima of the functions determine the scales where the descriptions are the most stable; their corresponding regions are depicted by circles in the image. Our algorithm selects the *absolute minimum* (shown as bright thick circles) for each point, yet in cases of extreme scale changes we recommend choosing all minima and discovering multiple sparse selections of scales per keypoint locations. Multi-scale points which correspond to the same image structure often have the same absolute minimum, i.e., result in the same region. In this case only one of them is kept in our implementation. To limit the number of selected regions an additional threshold can be used to reject unstable keypoints, i.e., if the minimum change of description is above a certain value the keypoint location is rejected. For each point we use a percentage of the maximum change over scales at the point location, set to 50% in our experiments.

Our algorithm is in the following referred to as *Maximally Stable Local SIFT Description* (MSLSD). Depending on the location detector we add the prefix H for Harris

and L for Laplacian, i.e., H-MSLSD and L-MSLSD.

### Illumination and Rotation Invariance

Our new detectors are robust to illumination changes, as our scale selection is based on the SIFT descriptor. Recall, that the SIFT descriptor is invariant to affine illumination changes.

Many applications require representations that are invariant to similarity transformations including rotation. This is either achieved by a rotation invariant descriptor (Lazebnik *et al.*, 2003), or, as we discussed when we introduced SIFT, by the extraction of a dominant orientation. In case of SIFT, if detected keypoints have poorly defined orientations, the resulting descriptions may become unstable and noisy. (This is not the case if the detected regions have a centered circular texture or they are completely homogenous.) In our algorithm, we orient the patch in the dominant direction prior to the descriptor computation for each scale. Maximal description stability is then found for locations with well defined local gradients. In our experiments a *-R* suffix indicates rotation invariance. Experimental results in Section 2.4 show that our integrated estimation of the dominant orientation can significantly improve results, in contrast to other detectors lacking this type of stability.

### Affine invariance

The affine extension of our detector is based on the affine adaptation in (Lindeberg and Garding, 1994; Baumberg, 2000), where the shape of the elliptical region is determined by the second moment matrix of the intensity gradient. However, unlike other detectors (Lazebnik *et al.*, 2003; Mikolajczyk and Schmid, 2004b), we do not use this estimation as a post-processing step after scale selection, but estimate the elliptical region prior to the descriptor computation for each scale. When the affine adaptation is unstable, i.e., sensitive to small changes of the initial scale, the descriptor changes significantly and the region is rejected. This improves the robustness of our affine-invariant representation. In our experiments an *-Aff* suffix indicates affine invariance. Full affine invariance requires rotation invariance, as the shape of each elliptical region is transformed into a circle reducing the affine ambiguity to a rotational one. Rotation normalization of the patch is, therefore, always included when affine invariance is used in our experiments.

### Illustration of Scale Selection

Table 2.1 shows the number of extracted interest points for the motorbike image from Figure 2.6 (640x480). On the left, Harris and Laplacian interest points are extracted on each scale. Note that the number of multi-scale detections depends on the multiplier between neighboring scales of the image pyramid (1.2 in our case). On the right, we show the reduced number of points by the characteristic scale selection. The first

Detector	# of points	Scale-invariant detector	# of points
Multi-Scale Harris	2228	Harris-Laplace	1011
Multi-Scale Laplacian	4893	Harris-Harris	283
		Harris-Gen	66
		Our H-MSLSD	1225
		LoG	2862
		Our L-MSLSD	1261

Table 2.1: The number of interest points extracted for the image in Figure 2.6. On the left we shows multi-scale points with 1.2 multiplier between scales. On the right we show the results after scale selection with Harris-Laplace and Harris-Harris, Harris-Gen, our new H-MSLSD, and for LoG and our new L-MSLSD. See text for details.

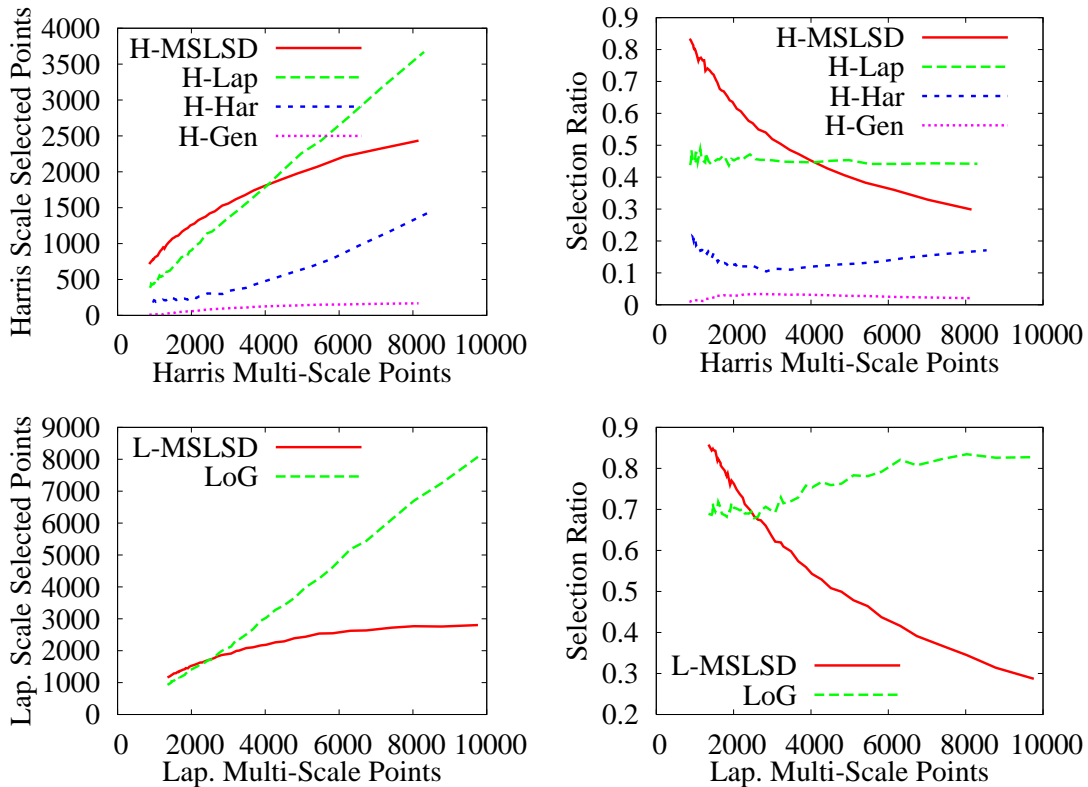


Figure 2.7: Number of selected points with gradually increased multi-scale points. Selection Ratio is define in (2.3) See text for discussion.

line shows the Harris-Laplace detector (Mikolajczyk and Schmid, 2001) followed by the other Harris-based detectors in the next three rows. The last two rows show scale selections on Laplacian points. In practice, to further limit the number of selected regions an additional threshold can be used to reject unstable keypoints. Apart from

LoG and Harris-Harris detectors, two separate thresholds can be set, one for the location and one for the scale function. Please also note that rotation invariance, which is enabled in these examples, further reduced the numbers of points found by Harris-Gen, H-MSLSD, and L-MSLSD.

Using a fixed image pyramid we define the *scale selection ratio* as

$$\text{Selection Ratio} = \frac{\text{Scale Invariant Points}}{\text{Multi Scale Points}} \quad (2.3)$$

Table 2.1 shows that H-Lap, H-MSLSD, LoG and L-MSLSD provide sufficient amount of detections, yet at the same time, their scale selection ratio is relative high, i.e., they keep many of the multi-scale points.

Figure 2.7 analyzes how much the detected number of points depends on the scale-space pyramid. We gradually change the scale multiplier between 1.5 and 1.03 and plot the number of scale-invariant points as a function of multi-scale points. Since the absolute number of points for each detector may easily be altered by a threshold, the interesting part of the curves are their shapes. One would expect that after a certain level adding intermediate new layers in the pyramid should not increase the number of detections. Surprisingly, the H-Lap detector (almost straight line) always selects a certain ratio of multi-scale points. This could be caused by noise or imprecise Laplacian scale selection on Harris points. The selection ratio of H-Har detector begins as expected, but after 3000 multi-scale points it actually starts to increase. H-Gen and H-MSLSD both demonstrate the expected descending shape. In case of the Laplacian-based detectors (Figure 2.7 second line), we draw similar conclusions, MSLSD stops increasing the number of detections after a certain limit. The expected behavior of our MSLSD implementation is probably due the smoothing factor introduced in our implementation during the computation of descriptor differences. It explicitly removes high frequency noise from the scale selection function. Also note that our scale selection always uses a finer scale-step then the multi-scale initialization.

## 2.3 Evaluation for image matching

This section evaluates the performance of the new detectors for image matching based on the evaluation framework in (Mikolajczyk *et al.*, 2005b).<sup>3</sup> We compare our results to H-Lap, H-Har, H-Gen and LoG respectively. The two main evaluation criteria of the framework we also applied are repeatability and matching rates.

The repeatability rate measures how well the detector selects the same scene region under various image transformations. Each sequence has one reference image and five images with known homographies to the reference image. Regions are detected for the images and their accuracy is measured by the amount of overlap between the

---

<sup>3</sup>The evaluation script may be downloaded from <http://www.robots.ox.ac.uk/~vgg/research/affine/evaluation.html>.



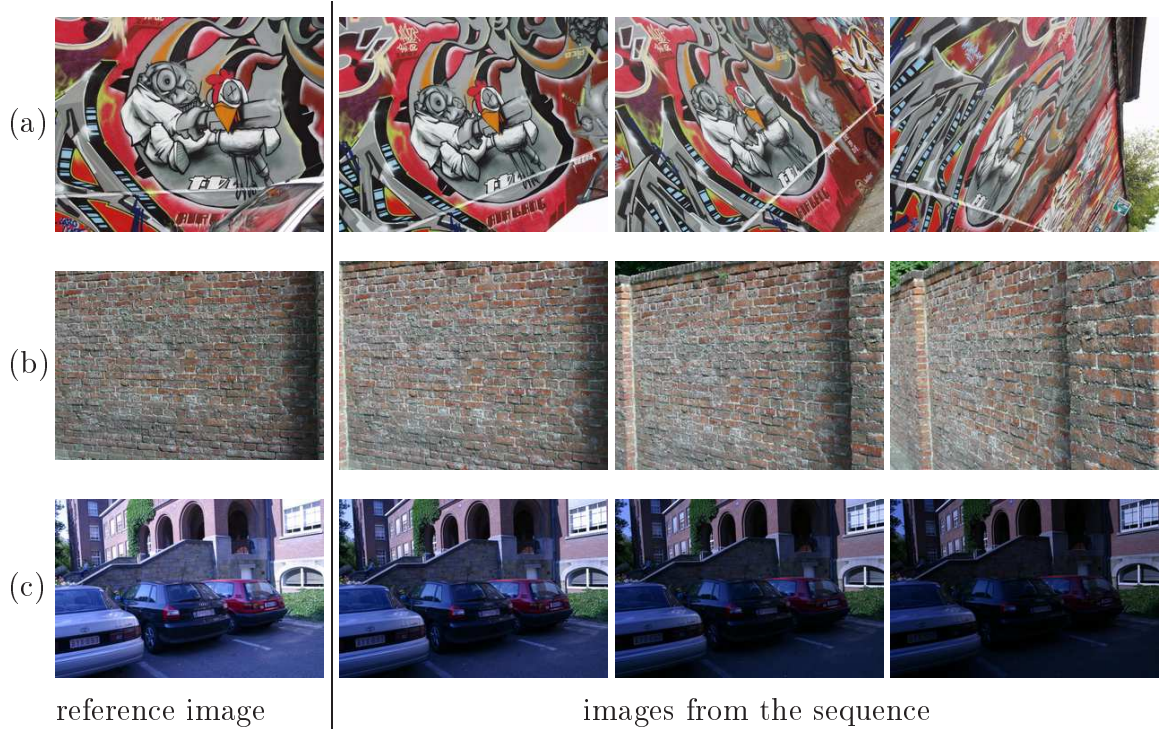


Figure 2.8: Image sequences used in the matching experiments. (a) and (b) are sequences with viewpoint change, while (c) contains illumination change. The first column shows the reference image, the other images are examples which homography is known to the reference. These sequences may be downloaded from <http://www.robots.ox.ac.uk/~vgg/research/affine/index.html>.

detected region and the corresponding region projected from the reference image with the known homography. Two regions are matched if their *overlap error* is sufficiently small:

$$1 - \frac{R_{\mu_a} \cap R_{(H^T \mu_b H)}}{R_{\mu_a} \cup R_{(H^T \mu_b H)}} < \epsilon_O$$

where  $R_{\mu}$  is the elliptic or circular region extracted by the detector and  $H$  is the homography between the two images. The union ( $R_{\mu_a} \cup R_{(H^T \mu_b H)}$ ) and the intersection ( $R_{\mu_a} \cap R_{(H^T \mu_b H)}$ ) of the detected and projected regions are computed numerically. As in (Mikolajczyk *et al.*, 2005b) the maximum possible overlap error  $\epsilon_O$  is set to 40% in our experiments. The *repeatability score* is the ratio between the correct matches and the smaller number of detected regions in the pair of images.

The second criterion, the matching score, measures the discriminative power of the detected regions. Each descriptor is matched to its nearest neighbor in the second image. This match is marked as correct if it corresponds to a region match with maximum overlap error 40%. The matching score is the ratio between the correct matches and



the smaller number of detected regions in the pair of images. See (Mikolajczyk *et al.*, 2005b) for more detailed discussion of the procedure.

### 2.3.1 Viewpoint Changes

The performance of our detectors for viewpoint changes is evaluated on two different image sequences with viewpoint changes from 20 to 60 degrees. Figure 2.8(a) shows sample images of the graffiti sequence. This sequence has well defined edges, whereas the wall sequence (Figure 2.8(b)) is more texture-like.

Figure 2.9 shows the repeatability rate and the matching scores as well as the number of matches for different affine-invariant detectors. The ordering of the detectors is very similar for the criteria repeatability rate and matching score, as expected. In the following we focus on the comparison of H-MSLSD-Aff to the other Harris based detectors, and L-MSLSD-Aff to LoG-Aff respectively. On the *graffiti sequence* (Figure 2.9, first row) the original Harris-Laplace (H-Lap-Aff) detector performs better than the other Harris detectors. On this sequence the new H-MSLSD-Aff are outperformed by H-Lap-Aff and H-Har-Aff. On the wall sequence, a more natural scene, results for H-MSLSD-Aff are slightly better than for H-L-Aff. This shows that the Laplacian scale selection provides good repeatability mainly in the presence of well defined edges. In case of the Laplacian our detector (L-MSLSD-Aff) outperforms the original one (LoG) for both sequences. This can be explained by the fact that LoG-Aff detects a large number of unstable (poorly repeatable) regions for nearly parallel edges, see Figure 2.10. A small shift or scale change of the initial regions can lead to completely different affine parameters of LoG-Aff. These regions are rejected by L-MSLSD-Aff, as the varying affine parameters cause large changes in the local description over consecutive scale parameters. Note that in case of affine divergence all detectors reject the points. This example clearly shows that description stability may lead to more repeatable regions. In case of natural scenes, as for example the wall sequence, this advantage is even more apparent, i.e., the difference between L-MSLSD-Aff over LoG-Aff is higher than for the graffiti sequence.

We can observe that we obtain a significantly higher number of correct matches with our L-MSLSD. This is due to a larger number of detected regions. This could increase the probability of accidental matches. To ensure that this did not bias our results—and to evaluate the effect of the detected region density—we compared the performance for different Laplacian thresholds for the L-MSLSD detector. Note that the Laplacian threshold determines the number of detections in location space, whereas the scale threshold rejects unstable locations and remains fixed throughout the thesis. Figure 2.11 shows that as the number of correct matches gradually decrease, the quality of the descriptors (matching score) stays the same. Consequently, we can conclude that the quality of the detections does not depend on the density of the extracted regions.

Figure 2.12 shows that in case of small viewpoint changes the scale-invariant versions of the detectors perform better than the ones with affine invariance. It also allows

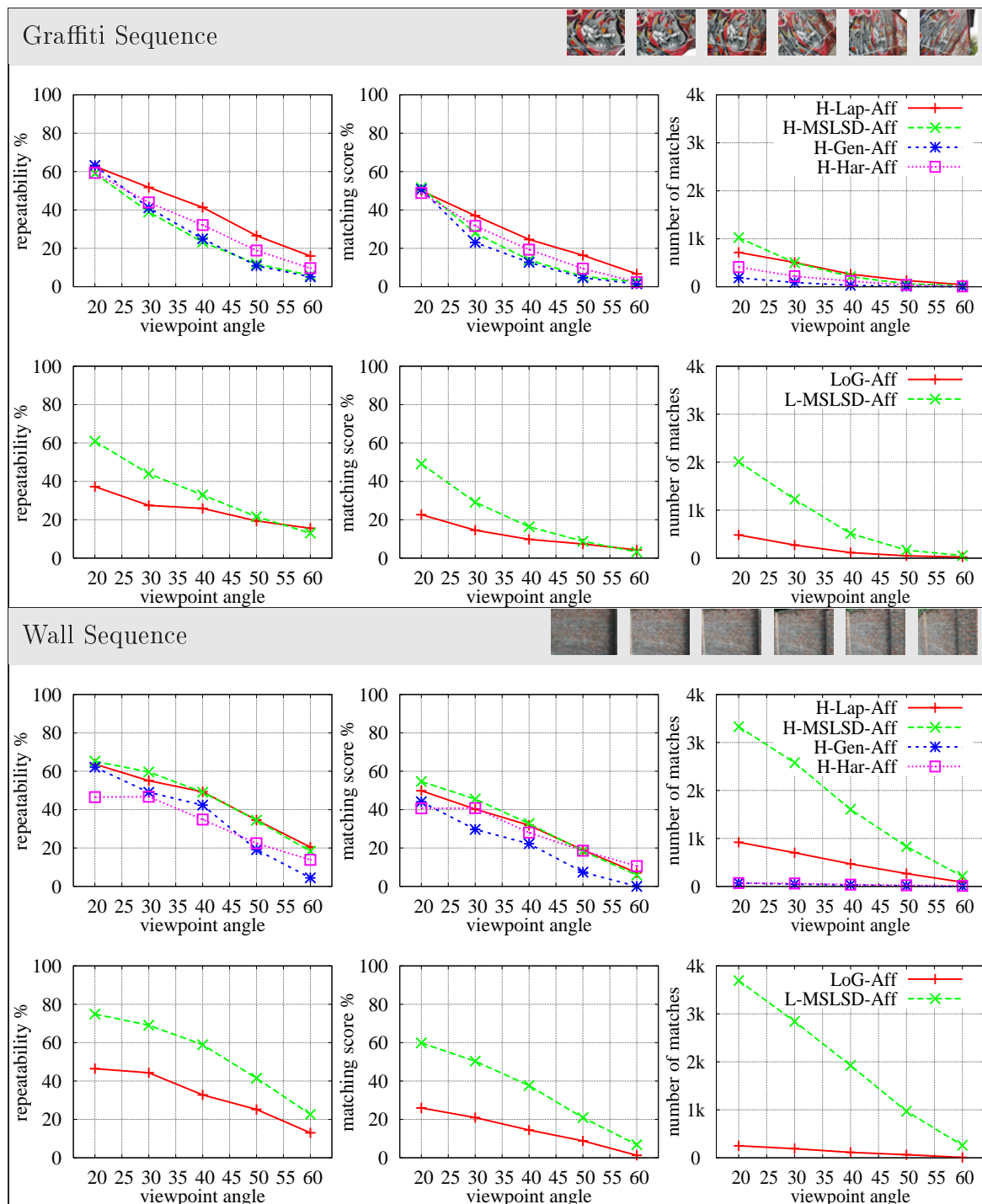


Figure 2.9: Comparison of detectors on viewpoint invariant sequences. The repeatabilities, matching scores and the number of matches are computed on the graffiti (first row) and on the wall (second row) sequences. See text for discussion.

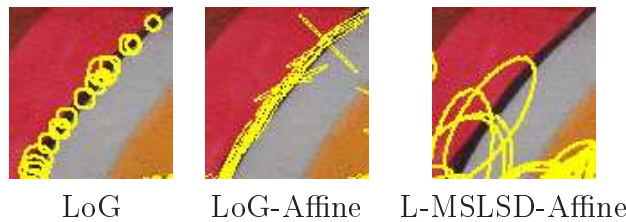


Figure 2.10: Output of LoG detection on a part of a graffiti image. On the left, the output of the standard LoG detector which is at the same time the input (initialization) of the affine adapted LoG (middle). On the right, the output of the new L-MSLSD-Affine. See text for discussion.

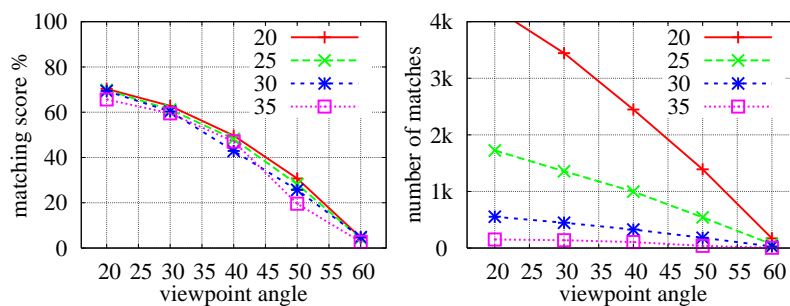


Figure 2.11: L-MSLSD results on the wall sequence while the threshold of the detector is gradually increased (20, 25, 30, 35). Higher threshold implies fewer detection and consequently a smaller number of absolute matches (second column).

to compare the scale-invariant detectors. On the graffiti images the original H-Lap and H-Gen performs better than its affine adapted version until  $30^\circ$  of viewpoint change. For our detector this transition occurs later around  $40^\circ$ . In the case of L-MSLSD and LoG the curves cross around  $35^\circ$  and  $40^\circ$  respectively. Interestingly, H-Har-Aff performs better on this sequence than H-Har. On the wall sequence it is almost never helpful to use the affine adaptation, scale invariance is sufficient until  $55 - 60^\circ$ . We can conclude that the use of affine invariance is not necessary unless the viewpoint changes are significant, and that it is more helpful in case of structured scenes. We can also observe that the scale-invariant versions H-Lap and H-MSLSD give comparable results for the graffiti sequence, whereas in the case of affine invariance H-Lap-Aff outperforms H-MSLSD-Aff. In the other cases, our scale-invariant detectors outperform their standard versions. In addition, the improvement of our detectors over the standard versions is more significant for scale invariance than for affine invariance, in particular for the Laplacian and the wall sequence.

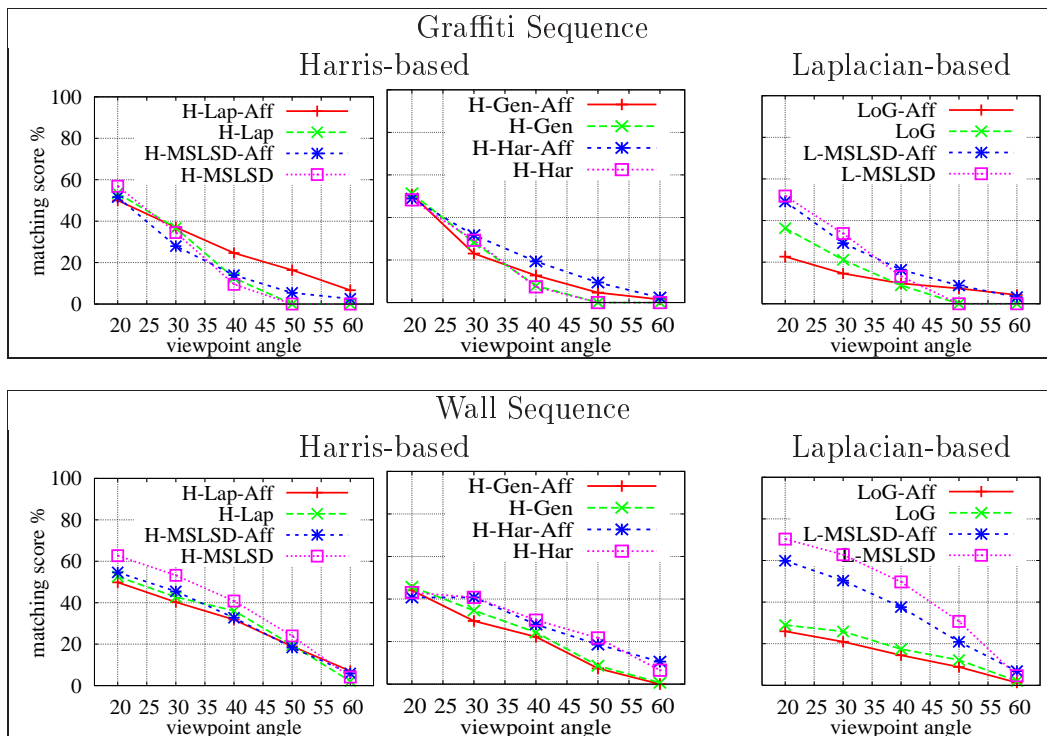


Figure 2.12: Comparison of using invariant detectors with and without affine estimation on the *graffiti* (first row) and the *wall* (second row) sequences. First column show results with the Harris, while the second is with the Laplacian-based detectors. See text for discussion.

### 2.3.2 Changes in Illumination

Section 2.2 motivated that our scale selection method offers robustness to properties provided by the underlying representation, in this case to illumination changes by SIFT. In this section, experiments are carried out for the Leuven sequence (Figure 2.8 (c)), i.e., images of the same scene under gradually reduced camera aperture. Figure 2.13 shows that the repeatability rate and matching score are significantly higher for our Harris- and Laplacian-based detectors than for the other Harris-based and LoG detectors respectively. This confirms that our scale selection is robust to lighting conditions as it is based on the SIFT descriptor which, recall, is invariant to affine illumination changes.

### 2.3.3 Overall Performance

Mikolajczyk *et al.* (Mikolajczyk *et al.*, 2005b) reported MSER (Maximally Stable Extremal Regions (Matas *et al.*, 2002)) as the best affine-invariant detector on the three image sequences used here. Figure 2.14 compares the matching score of our detectors

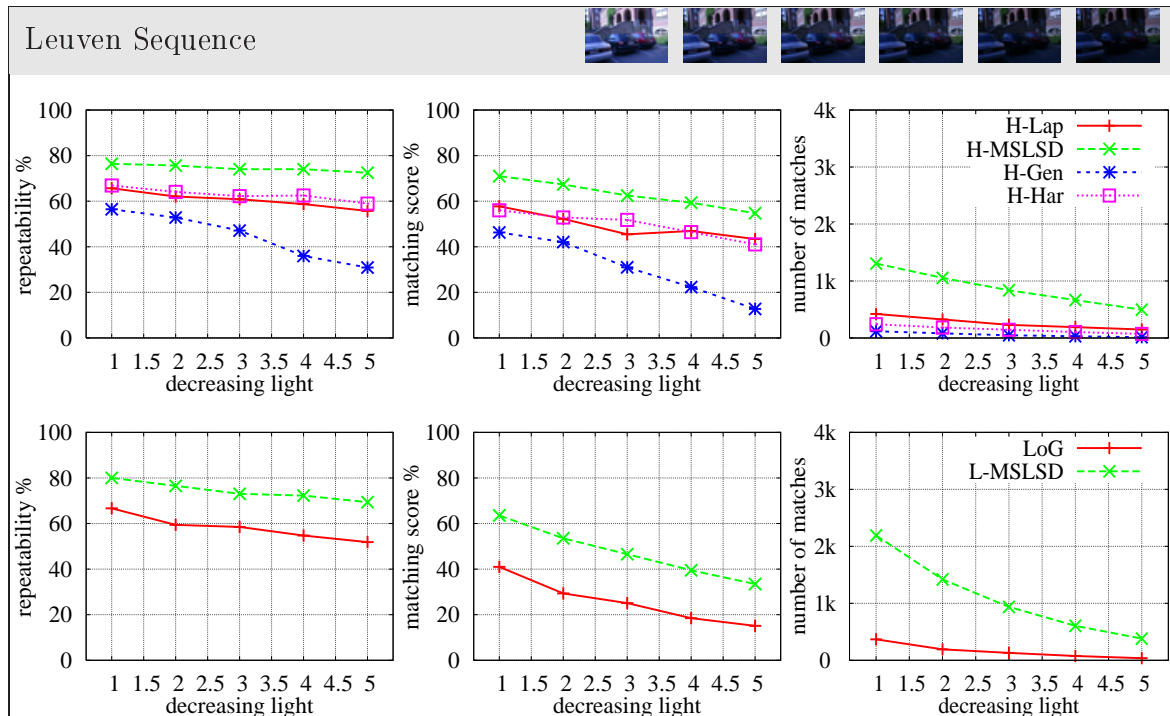


Figure 2.13: Detector performance on the *Leuven* sequence (illumination change). See text for discussion.

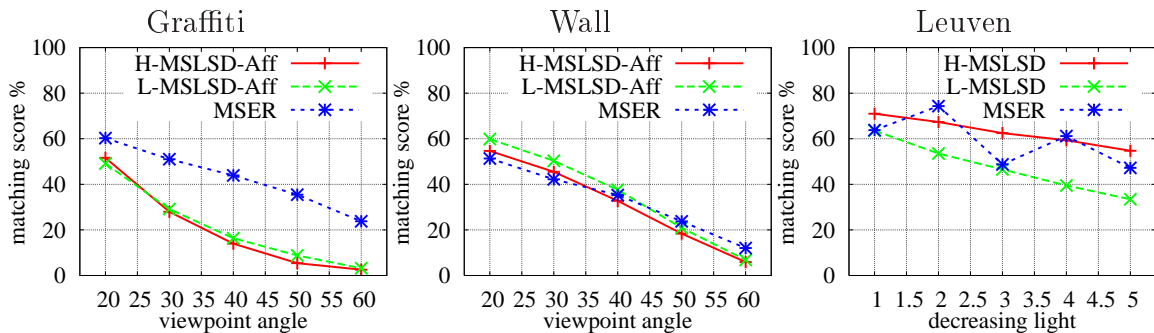


Figure 2.14: Comparison of the matching scores obtained for our detectors, H-MSLSD-Aff and L-MSLSD-Aff, and MSER.

to the performance of MSER on these sequences. Note that our results are directly comparable to the other detectors reported in (Mikolajczyk *et al.*, 2005b), as we use the same dataset and evaluation criteria. We can observe that L-MSLSD outperforms MSER on the wall sequence and that H-MSLSD performs better than MSER on the Leuven sequence. MSER gives better results than other detectors on the graffiti images. Note that due to the image structure of the graffiti scenes MSER selects significantly fewer keypoints than the other detectors.

Detector	Caltech databases		TUGraz1 databases	
	Motorbikes	Airplanes	Bicycles	People
H-Lap	98.25	97.75	92.0	<b>86.0</b>
H-Har	97.25	97.75	86.0	78.0
H-Gen	97.75	97.00	88.0	72.0
H-MSLSD	98.5	<b>99.25</b>	<b>94.0</b>	<b>86.0</b>
LoG	<b>98.75</b>	98.75	90.0	78.0
L-MSLSD	<b>98.75</b>	98.75	92.0	80.0
MSER	98.5	91.5	84.0	72.0
Fergus	96.0	94.0	<i>n.a.</i>	<i>n.a.</i>
Opelt	92.2	90.2	86.5	80.8

Table 2.2: Object class recognition results using seven different features sets and four different databases. Classification rates are reported at EER and compared to [Fergus \*et al.\* \(2003\)](#); [Opelt \*et al.\* \(2004\)](#).

## 2.4 Evaluation for image categorization

In this section we evaluate our new detectors for object and texture categorization. In both cases we perform image classification based on the bag-of-kepoints approach ([Csurka \*et al.\*, 2004](#)). Images are represented as histograms of visual word occurrences, where the visual words are clusters of local descriptors. The histograms of the training images are used to train a linear SVM classifier. In the case of object categorization the output of the SVM determines the presence or absence of a category in a test image. For multi-class texture classification we use the 1-vs-1 strategy. Vocabularies are constructed by the K-Means algorithm separately for each class. The number of clusters is fixed for each category, i.e., does not depend on the detector (400 for motorbikes and airplanes, 200 for bicycles, 100 for people, 1120 for Brodatz, and 1000 for KTH-TIPS). In all experiments we compare H-L to H-MSLSD and LoG to L-MSLSD and our representation is always SIFT.

### Evaluation for category classification

The experiments are performed for four different datasets. Motorbikes and airplanes of the CalTech dataset ([Fergus \*et al.\*, 2003](#)) contain 800 images of objects and 900 images of background. Half of the sets are used for training and the other half for testing. The split of the positive sets is exactly the same as ([Fergus \*et al.\*, 2003](#)). The TUGRAZ-1 dataset ([Opelt \*et al.\*, 2004](#)) contains people, bicycles, and a background class. We use the same training and test sets for two-class classification as ([Opelt \*et al.\*, 2004](#)).



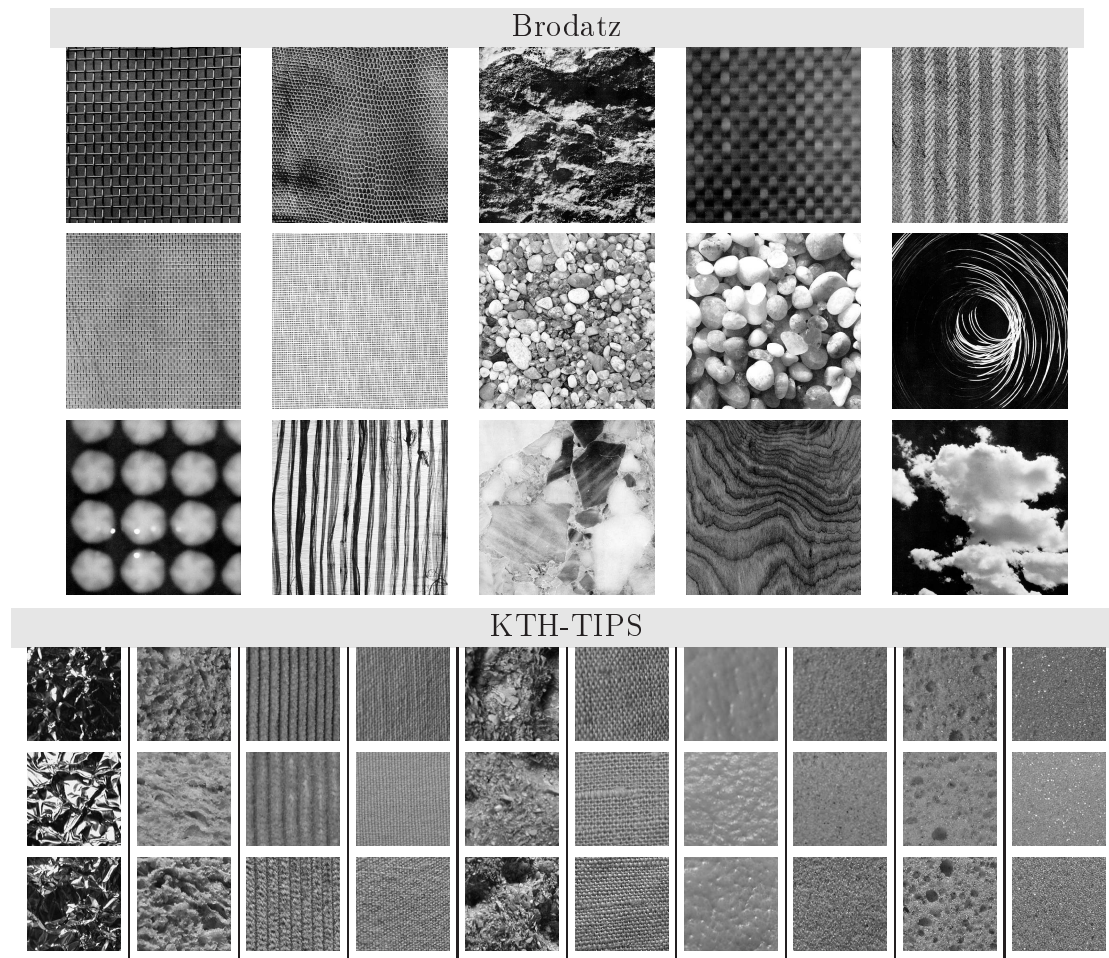


Figure 2.15: Example images from Brodatz and KTH-TIPS databases

Table 2.2 reports the classification rate at the EER<sup>4</sup> for four databases and seven different detectors. The last rows columns give results from the literature. We can observe that in most cases our detectors give better results when compared to their standard versions. In the remaining cases the results are exactly the same. This demonstrates that the local description based on our detectors is more stable and representative of the data. Comparison to other detectors are reported in Section 4.1.3.

Database	H-Lap-R	H-Har-R	H-Gen-R	H-MSLSD-R	LoG-R	L-MSLSD-R
Brodatz	88.3 $\pm$ 0.6	34.5 $\pm$ 1.0	81.6 $\pm$ 0.5	<b>92.0<math>\pm</math>0.5</b>	90.5 $\pm$ 0.5	<b>95.8<math>\pm</math>0.4</b>
KTH-TIPS	83.9 $\pm$ 1.1	42.5 $\pm$ 2.2	52.5 $\pm$ 1.2	<b>88.4<math>\pm</math>0.9</b>	71.2 $\pm$ 1.5	<b>81.1<math>\pm</math>1.2</b>

Table 2.3: Multi-class texture classification for two different datasets. Columns show results for different detectors, here their rotation invariant versions. Random class-assignment would give 10% on KTH-TIPS (10 classes) and 0.9% on Brodatz (112 classes). See text for more discussion.

### Evaluation for texture classification

Experiments are carried out on two different texture databases: Brodatz (Brodatz, 1966) and KTH-TIPS (Hayman *et al.*, 2004). The Brodatz dataset consists of 112 different texture images, each of which is divided into 9 non-overlapping sub-images. The KTH-TIPS texture dataset contains 10 texture classes with 81 images per class. Images are captured at 9 scales, viewed under three different illumination directions and three different poses. Our training set contains 3 sub-images per class for Brodatz and 40 images per class for KTH-TIPS. Each experiment is repeated 400 times using different random splits and results are reported as the average accuracy on the folds with their standard deviation over the 400 runs. Table 2.3 compares the results of our detectors H-MSLSD-R and L-MSLSD-R to H-Lap-R, H-Har-R, H-Gen-R and LoG-R. Note that we use the rotation invariant version here, as rotation invariance allows to group similar texture structures. We can observe that our scale selection technique, MSLSD, improves the results significantly in all cases. The poor performance of H-Har is due to the small number of detected features, for example on the Brodatz dataset H-Har did not detect any points in 285 images from 46 classes. This agrees with the conclusion of Mikolajczyk (2002, p.52 and p.58) that the Harris function rarely attains maxima in scale space.

Table 2.3 show result for a rotation invariant SIFT representation, for which the patch is rotated in the direction of the gradient orientation. Depending on the database, rotation invariance may help to group similar structures together and improve the classification accuracy. On the other hand making descriptors more similar, i.e., impose additional invariance, may result in performance drop. Consequently the following set of experiments, results reported in Table 2.4, analyzes the influence of rotation invariance on the representation. Results for the state-of-the-art detectors are, with one exception (LoG on the Brodatz dataset), better *without*, whereas results for our detectors are always better *with* rotation invariance. Notice that our improvement

---

<sup>4</sup>The Equal-Error-Rate is a standard way to compare recognition results of Receiver Operation Characteristic curves. It corresponds to the point where the classification errors on the positive and negative examples are equal, i.e.,  $p(\text{TruePositives}) = 1 - p(\text{FalsePositives})$ .



Brodatz			KTH-TIPS		
Detector	no rot.inv.	rot.inv.(-R)	Detector	no rot.inv.	rot.inv.(-R)
H-Lap	89.2 $\pm$ 0.6	$\leftarrow$ 88.3 $\pm$ 0.6	H-Lap	85.8 $\pm$ 1.1	$\leftarrow$ 83.9 $\pm$ 1.1
H-Har	36.9 $\pm$ 1.0	$\leftarrow$ 34.5 $\pm$ 1.0	H-Har	43.8 $\pm$ 3.0	$\leftarrow$ 42.5 $\pm$ 2.2
H-Gen	84.0 $\pm$ 0.5	$\leftarrow$ 81.6 $\pm$ 0.5	H-Gen	61.3 $\pm$ 1.3	$\leftarrow$ 52.5 $\pm$ 1.2
H-MSLSD	91.5 $\pm$ 0.6	$\rightarrow$ 92.0 $\pm$ 0.5	H-MSLSD	88.1 $\pm$ 1.2	$\rightarrow$ <b>88.4<math>\pm</math>0.9</b>
LoG	90.1 $\pm$ 0.5	$\rightarrow$ 90.5 $\pm$ 0.5	LoG	73.1 $\pm$ 1.5	$\leftarrow$ 71.2 $\pm$ 1.5
L-MSLSD	94.2 $\pm$ 0.5	$\rightarrow$ <b>95.8<math>\pm</math>0.4</b>	L-MSLSD	80.9 $\pm$ 1.3	$\rightarrow$ 81.1 $\pm$ 1.2

(a)

(b)

Table 2.4: Classification accuracy with and without rotation invariance. Results for (a) Brodatz and (b) KTH-TIPS datasets and different detectors.

may also be achieved on databases, such as Brodatz, where textures are not rotated. In our opinion, the poor performance of the existing detectors is due to an unstable estimation of the orientation leading to significant errors/noise in the descriptions. Note, that the orientation of the patch is estimated after the region detection. In our MSLSD method rotation estimation is integrated into the scale selection criterion (cf. Section 2.2) which implies that only regions with stable dominant gradients are selected, and it therefore improves the quality of the image representation.

Table 2.4 shows the results for both Brodatz and KTH-TIPS texture datasets. Surprisingly, results for the state-of-the-art detectors are in general better *without* rotation invariance, whereas results for our method are improved by the additional normalization. The only exception is LoG on the Brodatz dataset, that shows a small improvement using rotation invariant description. The poor performance of the existing detectors is due to unstable orientation estimation which leads to significant errors/noise in the descriptor. In our MSLSD method rotation estimation is included into the scale selection criterion which implies that only regions with stable dominant gradients are selected, and therefore it improves the quality of the texture representations. Notice that improvement may also be achieved on databases, such as Brodatz, where textures are not rotated.

## 2.5 Implementation Details

In this section we present implementation details and the parameters used in our experiments.

### Interest Point Detectors

Harris-Gen and MSER. In the case of these two detectors, we use the implementation provided by their authors (Triggs, 2004; Matas *et al.*, 2002) with default

parameters. For the Harris-Gen detector we use the location+scale+rotation (4D) stability criteria.

**Harris-Laplace.** Our implementation is based on the PhD thesis of [Mikolajczyk \(2002\)](#).

First, we build a *multi-scale pyramid*, with a scale factor of 1.3, and apply the Harris corner detector with a threshold of 300 for every scale. Then, a scale selection algorithm verifies if the Laplacian function has a local extremum on each detected Harris points. If the scale selection criterium is not fulfilled, or the absolute value of the Laplacian is below a certain threshold (3.0), the keypoint is rejected.

**Harris-Harris.** This detector is implemented very similarly to Harris-Laplace. We build a multi-scale pyramid using the same parameters, i.e., a scale factor of 1.3, and compute Harris images for each scale. The difference is that here we use the Harris functions for scale selection. For each keypoint point we ensure that it has a maximum cornerness w.r.t. it neighbors both in location and scale space. For the Harris function we additionally used a threshold of 300, similarly to Harris-Laplace.

**LoG.** The implementation of this keypoint detector starts similarly as the two previous detectors, but for each image in the multi-scale pyramid we compute 2D Laplacian images. Keypoints in location space are selected by thresholding these images with the value of 15. For each candidate locations, we then verify if its Laplacian has an extrema on scale space. Otherwise, the keypoint is rejected.

**H-MSLSD and L-MSLSD.** We start by building a multi-scale pyramid. For each scale, we compute Harris or Laplacian images accordingly. The potential keypoint locations are selected on each scale using the same criteria as for Harris-Laplace or LoG (see above): the thresholds are also the same, 300 for Harris, and 15 for the Laplacian. For each candidate locations, we optimize the descriptor stability criterion in function of scale. During this optimization the change of description is computed for a denser scale-space with a scale factor of 1.05, in contrast to 1.3, which is used only in the initialization phase, i.e., determine possible characteristic locations. The absolute minimum of descriptor change on the smoothed (Gaussian  $\sigma = 3$ ) descriptor-change function is then the selected characteristic scale. It may happen that candidates that are close by, or that on different scales correspond to the same image structure, have the same absolute minimum in our final selection, i.e., after optimization they result in the same region. In this case only one of them is kept in our implementation. To limit the number of selected regions, and to impose higher stability, an additional threshold is used to reject unstable keypoints. In our implementation this is a threshold relative to the maximum change in description, 50% in our experiments.

In general all keypoint detectors are applied to the images without any preprocessing. In our experiments all points detected below scale 2 are omitted, as they have too little information to compute appearance descriptors.

### Invariances

*Rotation invariance* is achieved by estimating the dominant orientation, and by pre-rotating the patch before descriptor computation. For the dominant orientation we use the gradient direction in the center of the patch estimated with the appropriate Gaussian kernel according to the detection scale.

The *affine adaptation* for all detectors is based on the second moment matrix of the intensity gradient, and it is identical for all the above detectors, but MSER. Rotation invariance is always included when affine invariance is used. For the MSER detector an ellipse is fitted on the detected region to determine the affine shape of the region. The keypoint location is the center of the ellipse.

### Local Descriptors

Before computing the descriptors, regions are normalized to obtain the required invariance. We map each neighborhood to a standard circular region, with smoothing in the case of downscaling. For all experiments in this thesis we use the SIFT descriptor with a 4x4 grid (index size), and with 8 bin orientation histograms. The resulting dimension of the descriptor is 128. The descriptor is first normalized to unit length, then bin values larger than 0.2 are truncated, and the vector is renormalized. Scales of keypoints detected by Harris-Laplace, Harris-Harris, Harris-Gen and LoG are multiplied by a factor of 2 prior to descriptor computation.

### Image Matching Experiments

These experiments are carried out with the publicly available evaluation framework of Mikolajczyk *et al.* (2005b). We use affine invariant detectors (cf. -Aff) for viewpoint changes, and rotation and scale invariant ones otherwise.

### Image Categorization Experiments

For each image database the two class categorization experiments use separate vocabularies built by kmeans. The number of clusters, and therefore the number of bins in the histogram is 400 for motorbikes, 200 for bicycles, 100 for people, 1120 for Brodatz textures, and 1000 for KTH textures. These numbers are chosen manually, according to the size of the database. For the classification we use linear SVM (SVM<sup>light</sup> (Joachims, 1999) implementation) with the trade-off between training error and margin,  $c = 0.005$ .

## 2.6 Conclusions

This chapter has introduced an approach for selecting characteristic scales based on the stability of the local description. We experimentally evaluated this technique for the SIFT descriptor, i.e., Maximally Stable Local SIFT Description (MSLSD). A new key property for interest points detectors, *local description stability*, has been introduced and discussed. We also demonstrated how a stable estimate of affine regions and orientation can be integrated in our method. Results for MSLSD versions of Harris and Laplacian points outperformed in many cases their corresponding state-of-the-art versions with respect to repeatability and matching, in particular under challenging conditions such as highly textured scenes and under different lighting conditions. For object category classification MSLSD achieved better or similar results for four datasets. In the context of texture classification our approach always outperformed the standard versions of the detectors.



# Discriminative Feature Selection for Object Class Appearance

---

THE selection of discriminative features is typically used to either improve classification performance or to reduce the size of the feature set. If the goal is a higher recognition rate, appropriate selection methods eliminate unimportant features, thus reducing the noise prior to classification. On the other hand, if the method is used to reduce the size of the feature set, the constructed sparse representation can significantly decrease processing time as well as required resources.

Due to the recent popularity of local image representation and the increasing size of datasets, feature selection has become important in computer vision. Many learning methods are unable to handle the huge feature sets produced by dense multi-scale representations. Although scale-invariant interest point detectors dramatically reduce this amount, discovering discriminative features can further improve the feature set. Selecting discriminative features helps to separate objects from background, and therefore can be used directly for classification or to support and improve more complex learning methods.

Figure 3.1 illustrates the importance of discriminative feature selection. The two scale-invariant regions in Figure 3.1(a) have very similar appearance. However, one of them lies on the background and the other on the object (bicycle). This region is therefore not discriminative for the bicycle class. Non-discriminative descriptors typically occur with small tubular or transparent parts, and with “donut-like” patches. Figure 3.1(b) shows discriminative features of the bicycle class determined by one of our selection methods.

## Related Work

In the following we present a state-of-the-art on discriminative feature selection. Many of the methods were originally developed and used in text classification. In document categorization, the challenge raised by the large number of features, i.e., the num-



Figure 3.1: Illustration of feature selection. (a) Two similar regions which cannot be used in a purely appearance based system to distinguish between the bicycle and the background. (b) The most discriminative features of the bicycle determined by our method.

ber of *words*, has made experts realize the need for feature selection. Techniques to choose discriminative features, i.e., features which are particular for a given class, has been extensively studied for document retrieval. Many of the methods discussed in this chapter are motivated by applications from text classification; some have already been used, while others are applied here for the first time for computer vision. In text categorization among many available methods the most basic techniques include domain specific *stop word removal* to avoid uninformative features, *stemming*<sup>1</sup>, and the *exclusion of overly common words*. A number of feature scoring methods have been used in *filters*, among which mutual information and odds ratios are the most popular. Filtering methods compute a score for each feature according to a chosen selection metric, then take the best  $n$  features as a final representation. Recent studies (Forman, 2003; Mladenić *et al.*, 2004) show that standard classifiers, such as naïve Bayes or k-Nearest-Neighbor, can explicitly profit from such selections: using a subset of features significantly improves their classification performance, particularly for classes with limited training examples. Linear Support Vector Machines implicitly select the useful features, therefore, they are more robust to insignificant data. Experiments of Mladenić *et al.* (2004) show that the SVM does not improve with feature selection, but it yields better performance when the reduced feature space allows a larger set of training examples. Joachims (1998) states that SVMs eliminate the need for feature selection and experimentally shows that classifiers built on the low-ranked features still perform better than random. Findings of Gabrilovich and Markovitch (2004) are similar to Joachims (1998) with respect to the latter, however, they found those low-utility features redundant rather than irrelevant. They show that using *out-*

<sup>1</sup>Stemming algorithms, or stemmers, have been developed to reduce a word to its stem or root form. This linguistic normalization is commonly used to reduce the number of words (e.g., in search engines), however, it is considered feature engineering rather than selection.

---

*lier count*—a measure estimating feature redundancy by outlier analysis—for ordering datasets reflects the degree to which a dataset can be described by only a few features. They also define a class of problems where feature selection can significantly improve the accuracy of linear SVMs. Forman (2003) introduces Bi-Normal Separation as a feature scoring method and shows improvement with SVM host classifiers.

Our study mainly focuses on filtering techniques, however there also exists another large group of selection methods, the wrappers (John *et al.*, 1994). Examples of wrappers are sequential forward and backward selection or genetic search. Wrapper methods evaluate all possible subsets of the features by repetitively calling the induction algorithm (classifier) as a black-box, and choose the subset with the highest performance. Comparisons find that wrapper methods are superior to filters (Kohavi and John, 1997), although those studies are limited to lower dimensional representation. For large scale problem these NP-hard methods are impractical, and filter methods are used instead. A valuable empirical study of filter methods for text classification is written by Forman (2003).

In the domain of text classification feature selection is typically applied on documents represented as *bag of words* (Sebastiani, 2002), i.e., by histograms built on occurrences of words. The construction of feature sets (visual vocabularies) are more complex in computer vision. They are two widely used approaches: feature sets are a set of descriptors computed on local regions (Viola and Jones, 2001; Opelt *et al.*, 2004), or are the result of a vector quantization algorithm applied on the descriptor space (Agarwal *et al.*, 2004; Weber *et al.*, 2000b; Willamowski *et al.*, 2004). In the latter case feature extraction (construction) is usually achieved by a clustering algorithm. Cluster centers can be interpreted as visual words (Sivic and Zisserman, 2003), and image representations based on occurrence histograms are called *bag of features* or *bag of keypoints* (Willamowski *et al.*, 2004). Some recent methods combine feature selection and local representation for object recognition. Viola and Jones (2001) extract rectangular Haar-like features to represent local parts of faces. They build a fast and reliable face recognition framework by the linear combination of classifiers based on individual features using Adaboost. Chen *et al.* (2001) also use boosting to construct components by local non-negative matrix factorization. Opelt *et al.* (2004) apply Adaboost for individual local descriptors to learn a local feature-classifier for determining the presence or absence of objects in images. Torralba *et al.* (2004) develop a framework for sharing features between object classes. They use multi-class boosting to efficiently select the common features to improve generalization, as well as to reduce the final computation cost. Mahamud and Hebert (2003) select discriminative object parts and develop an optimal distance measure for nearest neighbor search. Rikert *et al.* (1999) use a mixture model that retains only discriminative clusters, and Schmid (2001) selects significant texture descriptors in a weakly supervised framework. Both Rikert *et al.* (1999) and Schmid (2001) select features based on their discriminative score. The former uses the *term strength* (class conditional probability), while the latter uses the normalized likelihood ratio computed on the training



images. Ullman *et al.* (2001) use image fragments and combine them with a linear discriminative type classification rule. Their selection algorithm is based on mutual information. Fleuret (2004) uses conditional mutual information to select discriminative edge-based features for face recognition. He discusses and compares his method with  $k$ -NN, naïve Bayes, and SVM host classifiers. The study of Vidal-Naquet and Ullman (2003) shows that linear classifiers can be learned using only a small set of features if they are informative. Their method uses a greedy integrative algorithm based on mutual information to select features for classification of cars. There are a few recent applications using linear SVM based selection. Jurie and Triggs (2005) has experimentally evaluated visual vocabularies created by different clustering algorithms together with ranking methods based on linear SVM, mutual information and odds ratio. They observed that the SVM selection is superior to the others. Fan and Lu (2005) integrate SVM discriminative feature selection in a multi-class framework to efficiently classify faces from different viewpoints. They have shown significant speed-up in recognition without major degradation in classification performance.

## Overview

This chapter studies discriminative local feature selection for computer vision. In Section 3.1 we first introduce a probabilistic notation and then in Section 3.2 describe different scoring techniques as well as discuss their various properties. Section 3.3 shows how to build a visual vocabulary, as well as demonstrates and compares selection techniques on real images. Experiments in Section 3.3.2 evaluate individual feature classification, which decides whether a feature lies on the object or not. In Chapter 4 we integrate the rankings into an object detection and localization framework.

## 3.1 Probabilistic Interpretation

This section defines the probabilistic notation used in Section 3.2 to introduce different scoring techniques. Our notation is based on a given set of features  $\mathbb{F} = \{f_1, f_2, \dots, f_k\}$ , and a set of measurements  $\mathbf{x}_j$ . In our experiments the features are based on *visual words* (cf. Section 3.3.1), and the measurements are local invariant descriptors presented in the previous chapter.  $f_i$  is a binary variable indicating the existence of visual word  $i$ .  $\mathbf{x}^\oplus$  and  $\mathbf{x}^\ominus$  are positively and negatively labeled local descriptors (image patches). In the case of weakly supervised data, as in Section 3.3.2, positive labels may not necessary mean positive descriptors, but instead *unlabelled* descriptors from positive images. Patches from negative images always have negative ( $\ominus$ ) labels. This section does not detail the generation of the feature set  $\mathbb{F}$ ; it assumes the probabilities  $P(f_i|\mathbf{x}_j)$  are given for all features  $f_i$  and descriptor  $\mathbf{x}_j$ .

Let  $N^\oplus$  and  $N^\ominus$  be the total number of positively and negatively labeled  $\mathbf{x}_j$ . In our experiments they correspond to the number of descriptors extracted from positive and negative images respectively. We can then introduce the following notations:

$P(\oplus)$  is the probability that a randomly drawn  $\mathbf{x}_j$  is from a positively labeled image.

$$P(\oplus) = \frac{N^\oplus}{N^\oplus + N^\ominus}.$$

$P(\ominus)$  is the probability that a randomly drawn  $\mathbf{x}_j$  is from a negatively labeled image.

$$P(\ominus) = \frac{N^\ominus}{N^\oplus + N^\ominus}.$$

$P(f_i)$  is the average probability that a randomly drawn  $\mathbf{x}_j$  belongs to feature  $f_i$ , and can be estimated by

$$P(f_i) = \frac{\sum_j P(f_i|\mathbf{x}_j)}{N^\oplus + N^\ominus}.$$

$P(\bar{f}_i)$  is the average probability that a randomly drawn  $\mathbf{x}_j$  does not belong to feature  $f_i$ .

$$P(\bar{f}_i) = \frac{\sum_j P(\bar{f}_i|\mathbf{x}_j)}{N^\oplus + N^\ominus}.$$

$P(f_i, \oplus)$  is the joint probability that a descriptor belongs to feature  $f_i$  and is in a positively labeled image:

$$P(f_i, \oplus) = \frac{\sum_{j=1}^{N^\oplus} P(f_i|\mathbf{x}_j^\oplus)}{N^\oplus + N^\ominus}.$$

It can be interpreted as the *probability of true positives* on our measurement set  $\{\mathbf{x}_j\}$ .

Joint probabilities  $P(f_i, \ominus)$ ,  $P(\bar{f}_i, \oplus)$ , and  $P(\bar{f}_i, \ominus)$  are defined similarly, and they correspond to the probability of *false positives*, *false negatives*, and *true negatives* respectively.

$P(f_i|\oplus)$  is the conditional probability that a descriptor from a positively labeled image belongs to feature  $f_i$ . It is interpreted as the *true positive rate* and estimated by

$$P(f_i|\oplus) = \frac{\sum_{j=1}^{N^\oplus} P(f_i|\mathbf{x}_j^\oplus)}{N^\oplus}.$$

Conditional probabilities of *false positive rate* ( $P(f_i|\ominus)$ ), *false negative rate* ( $P(\bar{f}_i|\oplus)$ ), and *true negative rate* ( $P(\bar{f}_i|\ominus)$ ) are defined similarly.

Several selection criteria are based on how many descriptors are assigned to a given feature on object and background images. For visualization each feature can be represented as a point in a 2D frequency diagram, see Figure 3.2 for an example. The

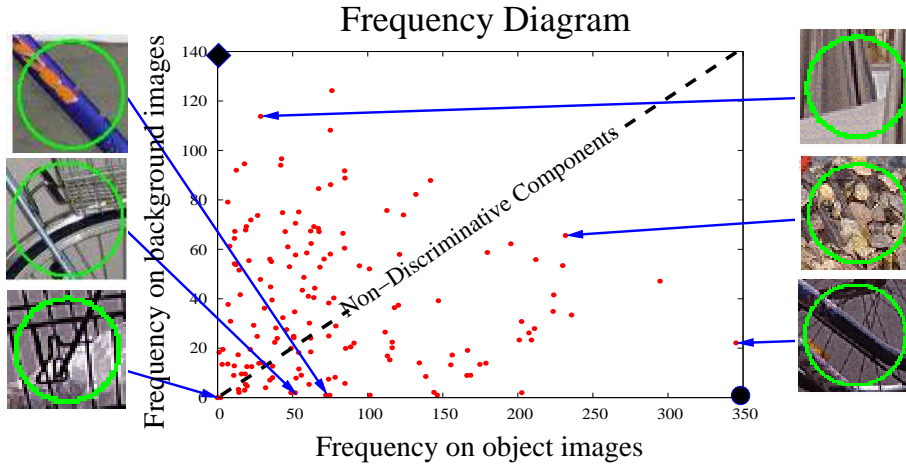


Figure 3.2: 2D frequency diagram for the features of the bicycles dataset. There are 200 features, regions are detected with ENTR detector. We show a representative region for six example clusters.

$x$  and  $y$  axes indicate respectively the frequency that descriptors of a feature appears in object and background images. Note that normalizing the axes by constant factor  $N^\oplus$  and  $N^\ominus$ , delivers exactly the same diagram with  $P(f_i|\oplus)$  and  $P(f_i|\ominus)$  on the  $x$  and  $y$  axes respectively. Descriptors of non-discriminative features are equally frequent in positive and negative images, and therefore they lie close to the dashed diagonal line. Features close to the bottom-right corner are discriminative for the objects, while those close to the top-left are good for the background.

### 3.2 Feature Scoring Techniques

In this section we introduce and discuss possible scoring techniques for feature ranking. We also show theoretical frequency diagrams for each method to demonstrate which features they tend to select.

**Freq:** *Frequency* is one of the simplest methods; it measures how many times a feature appears in the data set  $\{\mathbf{x}_j\}_{j=1}^{N^\oplus+N^\ominus}$ . More frequent features have a higher chance to appear on the unseen images, and thus, a sparse representation should find them useful. Frequency rank is defined as

$$\mathcal{R}^{(Freq)} = P(f_i) = P(f_i, \oplus) + P(f_i, \ominus).$$

The frequency diagram for  $\mathcal{R}^{(Freq)}$  is shown in Figure 3.3. We can observe that this selection method does not take into account the discriminative power. The frequency score peaks at the most frequent and the least discriminative top-right corner. However,  $\mathcal{R}^{(Freq)}$  is often used to reject rare features. For example

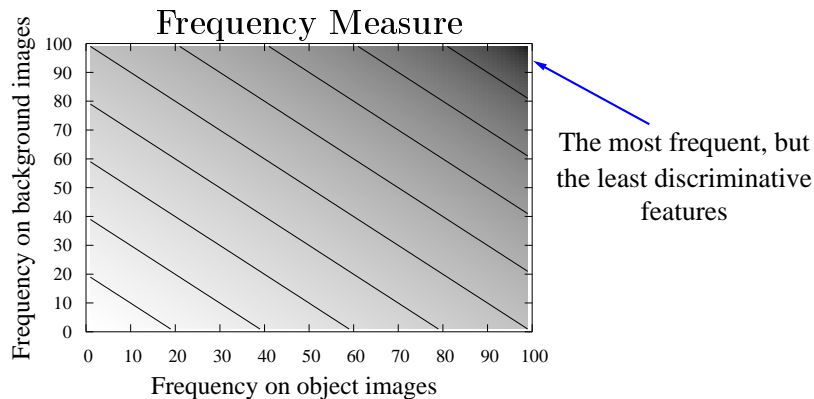


Figure 3.3: Selection by Frequency ( $\mathcal{R}^{(Freq)}$ ). Darker regions correspond to higher scores. Isocontours indicate the same value across the plot. See the text for discussion.

Weber *et al.* (2000b) ignore small clusters (i.e., rare object parts) to reduce the computational complexity of their method for learning a joint spatial model.

More sophisticated selection algorithms that combine frequency and discriminative information (such as *Mutual Information* or *Chi-Square*, see below), are shown to be superior to Frequency (Yang and Pedersen, 1997)<sup>2</sup>.

**OR:** The *Odds Ratio* is one of the most popular scoring methods. It is defined as the odds that a feature is labeled as positive normalized by the odds that it is labeled as negative.

$$\mathcal{R}^{(OR)} = \frac{P(f_i|\oplus) (1 - P(f_i|\ominus))}{(1 - P(f_i|\oplus)) P(f_i|\ominus)} = \frac{P(f_i, \oplus) P(\bar{f}_i, \ominus)}{P(f_i, \ominus) P(\bar{f}_i, \oplus)}.$$

The corresponding diagram can be found in Figure 3.5 (b). This measure is widely used in text classification (Caropreso *et al.*, 2001; Mladenić *et al.*, 2004; Ruiz and Srinivasan, 1999) for relevance ranking. Mladenić and Grobelnik (1999) report the best performance with *Odds Ratio* for multinomial naïve Bayes. The significant improvement—according to them—was due to the fact that this selection method is “compatible” with the classification algorithm. *Odds Ratio* is a very intuitive method and directly related to the discriminative power of the features, yet surprisingly it is not often used in computer vision. One explanation is that the Likelihood Ratio ( $\mathcal{R}^{(LIK)}$ ) is better motivated by probabilities and offers similar properties as *Odds Ratio*.

**LIK:** The *Likelihood ratio* (or probability ratio (Forman, 2003)) is basically the ratio (odds) of the probabilities if a random patch which belongs to feature  $f_i$  being

<sup>2</sup>What we call Mutual Information is called Information Gain by Yang and Pedersen (1997).

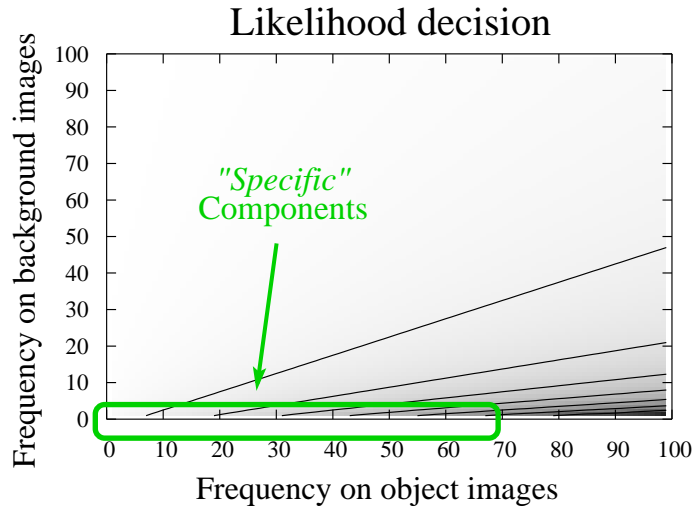


Figure 3.4: Likelihood scores for the 2D frequency diagram. Darker regions correspond to higher scores. Isocontours indicate the same value across the plot. “Specific” features have low frequency on the object images as well as close to zero frequency on the background images.

labeled positive or negative. It is defined as

$$\mathcal{R}^{(LIK)} = \frac{P(f_i|\oplus)}{P(f_i|\ominus)}.$$

In our earlier work (Dorkó and Schmid, 2003) as well as in (Schmid, 2001) it demonstrated very good performance. Intuitively, ranking by likelihood ratio is well suited for classification and detection purposes because it performs selection based on the classification rate. This is confirmed by our experiments in Section 3.3.2 and in Section 4.1. This method is robust to changes in parameter settings and overfitting of the estimated pdf of the data. On the other hand,  $\mathcal{R}^{(LIK)}$  and  $\mathcal{R}^{(OR)}$  typically prefer very “specific” features with near-zero values in the denominator. Even though these rare parts have individually low recall rates, combinations of them can provide sufficient recall with excellent precision. Figure 3.4 shows the likelihood scores for the 2D frequency diagram. Darker regions indicate higher likelihood scores. Features in the bottom right corner receive the highest values, since they are the most discriminative object features. We can also observe that “specific” features, located at the bottom of the diagram, also have high scores.

The computed scores (feature ranks) can be used, in many cases, within a recognition framework. It is often necessary to bound or to provide probabilistic explanation for these values. Since the likelihood ratio is the ratio of the correct- and mis-classification rates, the often used relationship between odds and prob-

abilities allow us to modify  $\mathcal{R}^{(LIK)}$  as

$$\widehat{\mathcal{R}}^{(LIK)} = \frac{\mathcal{R}^{(LIK)}}{1 + \mathcal{R}^{(LIK)}} = \frac{P(f_i|\oplus)}{P(f_i|\oplus) + P(f_i|\ominus)},$$

which is now bounded between 0 and 1. Notice, that in case of equal priors,  $\widehat{\mathcal{R}}^{(LIK)}$  is the posterior of a positively labeled image given the feature  $f_i$ . Schmid (2001) uses this measure to determine the significance of clusters. Our experiments in Section 4.2 use  $\widehat{\mathcal{R}}^{(LIK)}$  scores for object localization. Notice, that  $\widehat{\mathcal{R}}^{(LIK)}$  provides the same ordering of the features as  $\mathcal{R}^{(LIK)}$ .

**CHI:** *Chi-Square* is a well-known statistical test measuring the divergence from an expected distribution, or in our case, the lack of independence between the feature and the class label. Since we have binary variables, the formulation of Chi-Square has only four terms, as a typical case for problems set out in a fourfold table:

$$\widehat{\mathcal{R}}^{(CHI)} = t(P(f_i, \oplus), P(f_i)P(\oplus)) + t(P(\bar{f}_i, \oplus), P(\bar{f}_i)P(\oplus)) + \\ + t(P(f_i, \ominus), P(f_i)P(\ominus)) + t(P(\bar{f}_i, \ominus), P(\bar{f}_i)P(\ominus)),$$

where  $t(x, y) = \frac{(x - y)^2}{y}$ . After some basic algebra, the simplified computation (which is often used in scientific calculators) is

$$\widehat{\mathcal{R}}^{(CHI)} = \frac{[P(f_i, \oplus) P(\bar{f}_i, \ominus) - P(f_i, \ominus) P(\bar{f}_i, \oplus)]^2}{P(f_i) P(\bar{f}_i) P(\oplus) P(\ominus)}.$$

Note, that we used probabilities in the previous formulation. In order to retrieve the Chi-Square value in terms of feature frequency, similarly to the probabilities, a constant multiplier, the total number of descriptors can be applied:

$$\mathcal{R}^{(CHI)} = \frac{(N^\oplus + N^\ominus)[P(f_i, \oplus) P(\bar{f}_i, \ominus) - P(f_i, \ominus) P(\bar{f}_i, \oplus)]^2}{P(f_i) P(\bar{f}_i) P(\oplus) P(\ominus)}.$$

$\mathcal{R}^{(CHI)}$  has a value of zero when the feature and the class labels are independent. Figure 3.5 (g) shows the  $\mathcal{R}^{(CHI)}$  values in the 2D frequency diagram.  $\mathcal{R}^{(CHI)}$  uses both the frequency and the discriminative power of the features.

Comparative experiments of Yang and Pedersen (1997) have reported  $\mathcal{R}^{(CHI)}$  to be one of the most effective selection functions for text recognition. Motivated by their study many others (Galavotti *et al.*, 2000; Sebastiani, 2002; Zheng *et al.*, 2004) adapted it, and as a consequence it has become very popular in that domain.

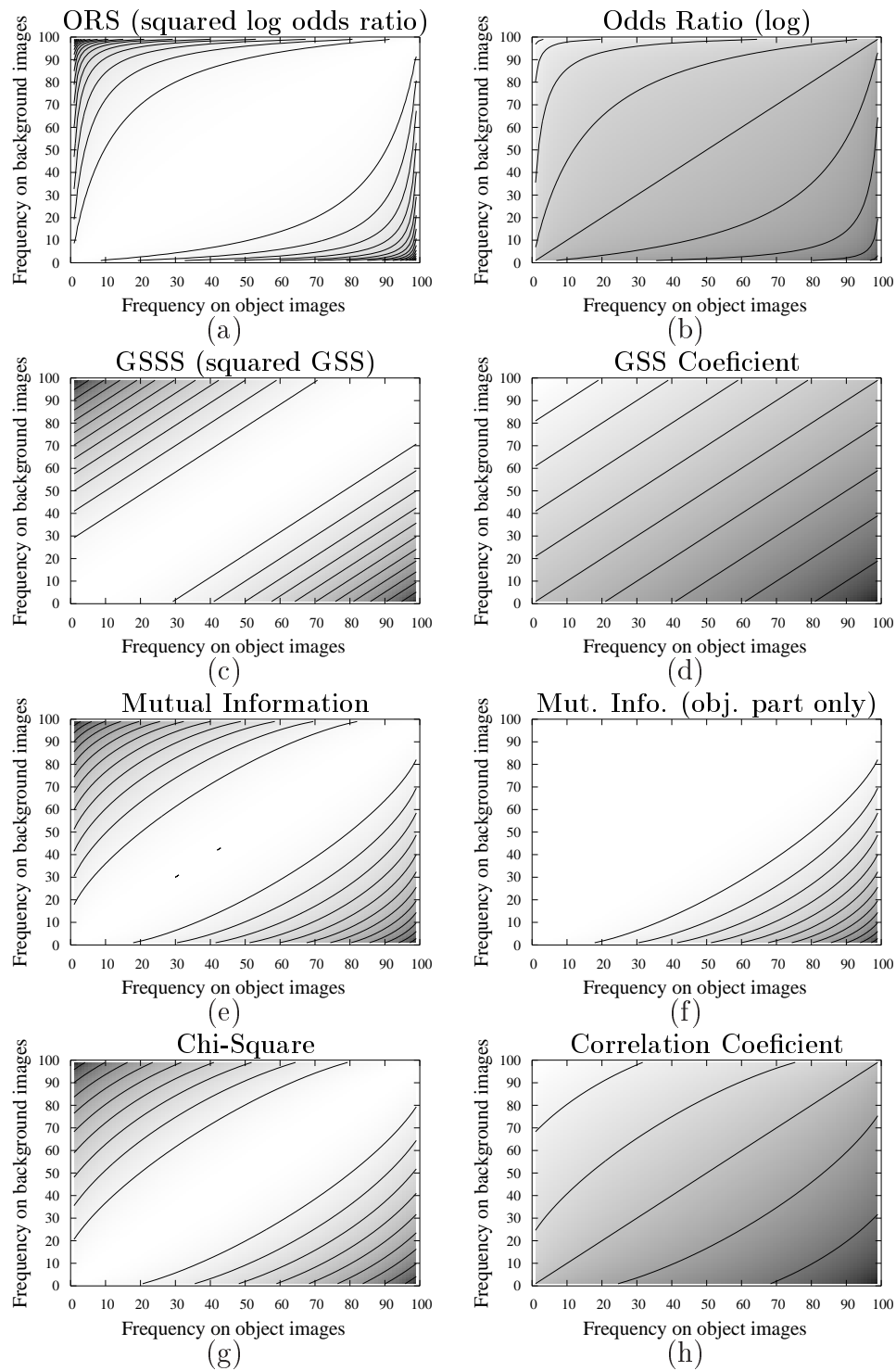


Figure 3.5: The selection scores for various ranking methods. Darker regions correspond to higher scores. Isocontours indicate the same value within the plot. See the text for discussion. Symmetric measures on the right; their corresponding “positive only” equivalent on the left.

**CC:** The *Correlation Coefficient* (also called NGL coefficient) was given by Ng *et al.* (1997) as

$$\mathcal{R}^{(CC)} = \frac{\sqrt{N^{\oplus} + N^{\ominus}} [P(f_i, \oplus) P(\bar{f}_i, \ominus) - P(f_i, \ominus) P(\bar{f}_i, \oplus)]}{\sqrt{P(f_i) P(\bar{f}_i) P(\oplus) P(\ominus)}}.$$

As Ng *et al.* (1997) point out and we demonstrate on Figure 3.5 (g),  $\mathcal{R}^{(CHI)}$  is a symmetric measure giving equal importance to the positive and negative features. While for a typical two class problem this can be useful, in many tasks it can be a drawback. If one of the classes is under-represented, e.g., in the case of an object-background problem, we cannot expect representative selection of its features. While  $\mathcal{R}^{(CC)^2} = \mathcal{R}^{(CHI)}$ , i.e., it keeps many properties of  $\mathcal{R}^{(CHI)}$ ,  $\mathcal{R}^{(CC)}$  is defined to prefer positive correlation between the feature and the positive class label. Experiments of Ng *et al.* (1997) show that  $\mathcal{R}^{(CC)}$  is superior to  $\mathcal{R}^{(CHI)}$ . Scores of  $\mathcal{R}^{(CC)}$  are shown in Figure 3.5 (h).

**MI:** *Mutual Information*

If the main purpose of our system is to produce a sparse object class representation, it is best to select a few discriminative and “general” features. Besides  $\mathcal{R}^{(CHI)}$  our other option is to use the mutual information (Papoulis, 1991) criterion, which ranks features based on their information content for separating the negative from the positive class. The mutual information between the label set  $\mathcal{L} = \{\oplus, \ominus\}$  and feature  $\mathcal{F}_i = \{f_i, \bar{f}_i\}$  (as two random variables) is defined as

$$I(\mathcal{F}_i; \mathcal{L}) = H(\mathcal{L}) - H(\mathcal{L}|\mathcal{F}_i),$$

where  $H(\cdot)$  and  $H(\cdot|\cdot)$  are Shannon’s entropy (Shannon, 1948) and conditional entropy respectively. Using our notation the  $\mathcal{R}^{(MI)}$  rank is defined as

$$\mathcal{R}^{(MI)} = \sum_{l \in \{\oplus, \ominus\}} \sum_{f \in \{f_i, \bar{f}_i\}} P(f, l) \cdot \log \frac{P(f, l)}{P(f) P(l)}.$$

Figure 3.5 (e) shows the decision surface for  $\mathcal{R}^{(MI)}$ . We can observe that  $\mathcal{R}^{(CHI)}$  (see Figure 3.5 (e)) and  $\mathcal{R}^{(MI)}$  have similar patterns of scores, and likewise in our experiments (Section 3.3.2) they show similar behavior, yet we found that  $\mathcal{R}^{(MI)}$  usually outperforms  $\mathcal{R}^{(CHI)}$ . The positive non-specific but discriminative features are located in the rightmost part of the lower triangle in the diagrams. The score pattern of  $\mathcal{R}^{(MI)}$  and  $\mathcal{R}^{(CHI)}$  indicate that more features are chosen from that area compared to  $\mathcal{R}^{(OR)}$  (Figure 3.5 (d)) and  $\mathcal{R}^{(LIK)}$ . This clearly displays the preference for more “general”, i.e., frequent, features.

**GSS:** The *GSS coefficient* is a scoring method motivated by  $\mathcal{R}^{(CC)}$ . Galavotti *et al.* (2000) suggested removing the constant factor  $\sqrt{N^{\oplus} + N^{\ominus}}$  as well as the denominator leading to

$$\mathcal{R}^{GSS} = P(f_i, \oplus) P(\bar{f}_i, \ominus) - P(f_i, \ominus) P(\bar{f}_i, \oplus).$$



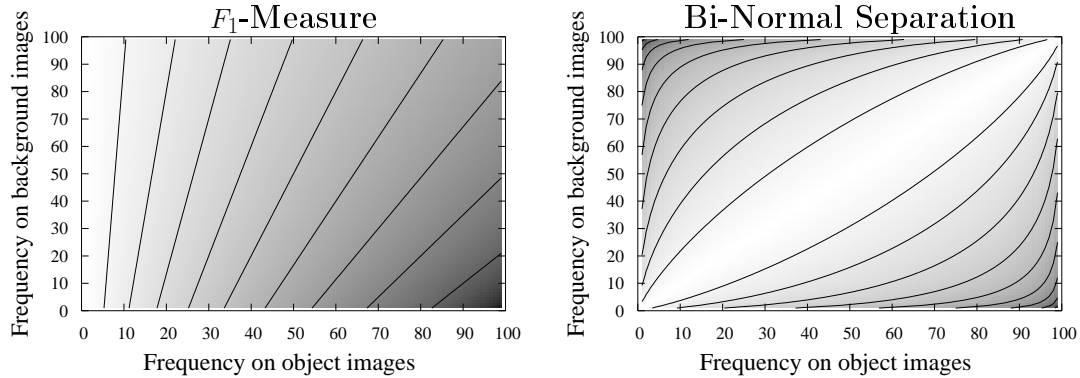


Figure 3.6: Selection by (a)  $F_1$ -measure and (b) Bi-normal separation. Darker regions correspond to higher scores. Isocontours indicate the same value across the plot. See the text for discussion.

In  $\mathcal{R}^{(CC)}$  the  $\sqrt{P(f_i)P(\bar{f}_i)}$  in the denominator emphasizes rare features, and therefore  $\mathcal{R}^{(GSS)}$  prefers even more frequent features than  $\mathcal{R}^{(CC)}$ . While this can be an advantage when the number of features is very large, it is not necessarily useful for present computer vision applications. The frequency diagram in Figure 3.5 (d) shows the modified surface.

**F1:** The  $F_1$ -measure (van Rijsbergen, 1979) is defined as a harmonic mean of precision and recall. It is often used to compare recall-precision curves, and therefore, it is one of the most used measures in detection frameworks and information retrieval. This motivates the direct application of this measure to feature selection. Using probabilistic notations  $\mathcal{R}^{(F_1)}$  is defined as

$$\mathcal{R}^{(F_1)} = \frac{2 P(f_i|\oplus) P(f_i, \oplus)}{P(f_i|\oplus) P(f_i) + P(f_i, \oplus)} = \frac{2 P(f_i, \oplus)}{2 P(f_i, \oplus) + P(f_i, \ominus) + P(\bar{f}_i, \oplus)}.$$

Its corresponding frequency diagram is shown on Figure 3.6 (left).

**BNS:** *Bi-normal separation* is defined by Forman (2003) as

$$\mathcal{R}^{(BNS)} = |F^{-1}(P(f_i|\oplus)) - F^{-1}(P(f_i|\ominus))|,$$

where  $F$  is the Normal c.d.f. An alternative interpretation of  $\mathcal{R}^{(BNS)}$  is motivated by ROC threshold analysis. It measures the separation between two standard *Normal* curves where their relative positions—the center of the curves—are prescribed by  $P(f_i|\oplus)$  and  $P(f_i|\ominus)$ . Their study shows improvement for SVM host classifiers using the  $\mathcal{R}^{(BNS)}$  feature selection. Its frequency diagram shows that  $\mathcal{R}^{(BNS)}$  neither cuts off features in the top-right and bottom-left corners as drastically as  $\mathcal{R}^{(MI)}$ , nor keeps the overly specific features like  $\mathcal{R}^{(OR)}$  and  $\mathcal{R}^{(LIK)}$ .

All filter methods that we have introduced so far rank features according to their individual power. Selection based on these rankings can lead to redundant and thus less informative sets if we limit the number of selected features. The three additional methods that we discuss in the following addresses this problem and select features *conditionally* on the others.

**AB:** *Adaboost* (Freund and Schapire, 1996a,b) combines several classifiers (*weak learners*) into an accurate (*strong*) classifier by an incremental voting procedure. In our framework the strong classifier labels an image  $x$  and is defined as

$$s(x) = \sum_{t=1}^T \mathcal{R}_t^{(AB)} h_t(x),$$

a linear combination of weak learners  $h_t(x)$ .  $h_t(x)$  is defined as a presence (+1) or absence (-1) of feature  $\widehat{f}_i$ .  $\widehat{f}_i$  is a binary feature and in our case it is present on an image if there are at least  $\theta_i$  descriptors of feature  $f_i$ , where  $\theta_i$  is set during training to maximize the mutual information between the feature  $\widehat{f}_i$  and the image labels  $\mathcal{Y} \in \{-1, 1\}$ . The weight of each weak learner is  $\mathcal{R}_t^{(AB)}$  and can be used as a rank for the features.  $T$  is the number of iterations. At each step  $t$  the Adaboost algorithm selects a weak learner that minimizes the weighted error  $\epsilon_j$ :

$$h_t(x) = \arg \min_{h_j \in \mathcal{H}} \epsilon_j = \sum_{i=1}^K D_t(i) [y_i \neq h_j(x_i)].$$

$D_t(i)$  are the weights for each training image in step  $t$ , which are initialized to  $1/K$ , where  $K$  is the number of features. The weight for the chosen classifier is then set to

$$\mathcal{R}_t^{(AB)} = \frac{1}{2} \log \left( \frac{1 + r_t}{1 - r_t} \right),$$

$$r_t = \sum_{i=1}^K D_t(i) h_t(x_i) y_i,$$

and the weights for the data are updated

$$D_{t+1}(i) = \frac{D_t(i) \exp(-\mathcal{R}_t^{(AB)} y_i h_t(x_i))}{Z_t}.$$

$Z_t$  is a normalization factor chosen so that  $D_{t+1}$  is a distribution.

Even though the purpose of Adaboost is to build a strong classifier, given as the sign of  $s(x)$ , it can be seen as a feature selection criterion as well, since weak learners (features) with higher weight have more influence the output of the strong classifier. In addition, we have not used the aggregation of the weak learners  $s(x)$ , only the ranks provided by the algorithm.

**CMIM:** *Conditional Mutual Information Maximization* is used to select a small subset of features that carries as much information as possible. The mutual information for  $\mathcal{L}$  and  $\widehat{\mathcal{F}}_i$  conditioned on  $\widehat{\mathcal{F}}_j$  is given as

$$I(\mathcal{L}; \widehat{\mathcal{F}}_i | \widehat{\mathcal{F}}_j) = H(\mathcal{L}) - H(\mathcal{L} | \widehat{\mathcal{F}}_i, \widehat{\mathcal{F}}_j),$$

where  $\widehat{\mathcal{F}}_i = \{\widehat{f}_i, \widetilde{f}_i\}$  is a binary random variable. Our feature set  $\{\widehat{\mathcal{F}}\}$  is constructed in the same way as in the previous section (as for  $\mathcal{R}^{(AB)}$ ). The ideal selection would minimize the entropy conditioned on the selected subset of features

$$\widehat{H}(\mathcal{L} | \widehat{\mathcal{F}}_{\nu(1)}, \dots, \widehat{\mathcal{F}}_{\nu(n)}).$$

However, as it is based on the joint entropy estimation as in (Yang and Moody, 1999), it is intractable with realistic sizes of training sets. Fleuret (2004) proposes an iterative solution where a new feature  $\widehat{f}_i^*$  is selected only if  $\widehat{I}(\mathcal{L}; \widehat{f}_i^* | \mathcal{F})$  is large for all  $\widehat{\mathcal{F}}$  that have been selected before, i.e.,  $\widehat{f}_i^*$  carries information about the labels  $\mathcal{L}$ . Formally

$$\nu(1) = \arg \max_n \widehat{I}(\mathcal{L}; \widehat{\mathcal{F}}_n),$$

i.e., the first feature is chosen by the original  $\mathcal{R}^{(MI)}$  criteria, and all the following features  $k + 1$  are selected by

$$\nu(k + 1) = \arg \max \left\{ \min_{l < k} \widehat{I}(\mathcal{L}; \widehat{\mathcal{F}}_n | \widehat{\mathcal{F}}_{\nu(l)}) \right\}. \quad (3.1)$$

This solution does not solve the problem, but offers a trade-off between individual power and independence and reduces the computational time by two orders of magnitude. An efficient implementation is given in Fleuret (2004). Even though we abbreviate this approach as  $\mathcal{R}^{(CMIM)}$ , this method rather than assigning rank (consequently decreasing score) to each feature, only orders the feature set, and thus the value  $\mathcal{R}^{(CMIM)}$  itself cannot be used as a rank. Fleuret (2004) uses  $\mathcal{R}^{(CMIM)}$  with naïve Bayes classifier for face recognition. Vidal-Naquet and Ullman (2003) proposes the criterion

$$\nu(k + 1) = \arg \max \left\{ \min_{l < k} I(\widehat{\mathcal{F}}_{\nu(n)}, \widehat{\mathcal{F}}_{\nu(l)} | \mathcal{L}) - I(\widehat{\mathcal{F}}_{\nu(l)}; \mathcal{L}) \right\}$$

to iteratively select features which is equivalent to (3.1).

**SVM:** *Linear Support Vector Machines (SVM)* were first used for feature selection by Sindhvani *et al.* (2001), then later by Brank *et al.* (2002). Mladenić *et al.* (2004) generalized the idea for linear classifiers. SVM (Vapnik, 1995) is a classifier that

finds a maximal margin hyperplane separating two classes of data. The predicted label for an unseen  $\mathbf{x}$  is given by

$$H(\mathbf{x}) = \text{sgn}[b + \sum_i \alpha_i K(\mathbf{x}, \mathbf{x}_i)],$$

which can be rewritten for the linear kernel  $K(\cdot, \cdot)$  as

$$H(\mathbf{x}) = \text{sgn}[b + \mathbf{w}^T \mathbf{x}], \text{ where } \mathbf{w} = \sum_i \alpha_i \mathbf{x}_i.$$

$\mathbf{w} = (w_1, \dots, w_K)$  are the weights (the normal to the separating hyperplane) that are learned during SVM training, and can be accessed directly. Features with higher absolute weights have more influence on the SVM prediction, and therefore can be used for feature selection. [Shih et al. \(2002\)](#) also point out that high  $|w_i|$  are more influential in determining the width of the margin.

To train the SVM and obtain the weights for our features we use the feature set  $\{\widehat{\mathcal{F}}\}$ , as for  $\mathcal{R}^{(CMIM)}$  and  $\mathcal{R}^{(AB)}$ . In our experiments  $\mathcal{R}^{(SVM)}$  indicates the ranking by  $|w_i|$

### Frequent — Discriminative — Redundant

Some of the measures presented in this chapter prefer frequent features while others prefer more discriminative ones.  $\mathcal{R}^{(Freq)}$  does not take discriminative power into account and  $\mathcal{R}^{(LIK)}$  only uses the discriminative power.  $\mathcal{R}^{(MI)}$ ,  $\mathcal{R}^{(CHI)}$ , and  $\mathcal{R}^{(BNS)}$  use both frequency and discriminative power. Choosing the appropriate measure for a given task is not straightforward. First one should decide if it is necessary to prune, i.e., significantly reduce, the input space. In those cases feature frequency should play an important role in the selection. Another alternative for pruning the space is to reject redundant features that do not provide *additional* discriminative information.  $\mathcal{R}^{(SVM)}$ ,  $\mathcal{R}^{(CC)}$  and  $\mathcal{R}^{(CMIM)}$  are examples of such selections. On the other hand, when accuracy is more important, a combination of many rare but discriminative features often gives better results. Furthermore, if the number of training examples are limited, rejection of top ranked features can help to avoid side effects caused by *outliers*. The *compatibility* between the learning framework (classifier) and the feature selection method is also important. [Mladenić et al. \(2004\)](#) analyzed the relationship for methods of naïve Bayes, perceptron, SVM and for selection methods  $\mathcal{R}^{(MI)}$ ,  $\mathcal{R}^{(OR)}$ , and  $\mathcal{R}^{(SVM)}$ . They found  $\mathcal{R}^{(SVM)}$  to be superior to others, even for naïve Bayes, which is the most *compatible* with  $\mathcal{R}^{(OR)}$ . For further discussion on compatibility we refer to [Mladenić et al. \(2004\)](#).

### One-Sided or Two-Sided

Scores obtained for the frequency diagrams indicate that some of the scoring techniques are one-sided while others are two-sided. Two-sided techniques treat positive

and negative features equally while one-sided do not. Among the presented methods  $\mathcal{R}^{(OR)}$ ,  $\mathcal{R}^{(LIK)}$ ,  $\mathcal{R}^{(CC)}$ ,  $\mathcal{R}^{(GSS)}$ , and  $\mathcal{R}^{(F_1)}$  (even when inverted) are one-sided and select only positive features, and  $\mathcal{R}^{(Freq)}$ ,  $\mathcal{R}^{(CHI)}$ ,  $\mathcal{R}^{(MI)}$ ,  $\mathcal{R}^{(BNS)}$ ,  $\mathcal{R}^{(CMIM)}$ ,  $\mathcal{R}^{(AB)}$  and  $\mathcal{R}^{(SVM)}$  are two-sided. In two-class recognition problems with fully supervised training, a two-sided measure can be a natural choice especially when the task is to discriminate between two object classes or two different types of scenes. On the other hand, when one class is the background, or our positive images contain background clutter, selecting negative features could be disadvantageous. Even though the presence of negative features on positive images can be discovered by learning techniques used after feature selection, due to the lack of sufficient background examples the trained systems become less *transferable* to new environments. Those systems might rely on particular background statistics, and therefore may only be used in specific cases. While [Forman \(2003\)](#) shows that all feature selection methods degrade when they are converted to be one-sided, [Ng et al. \(1997\)](#) develops  $\mathcal{R}^{(CC)}$  from  $\mathcal{R}^{(CHI)}$  to have a one-sided measure which then outperforms its two-sided equivalent. Two-sided measures may easily mislead the user when two-class or multi-class experiments do not contain a separate background category. Assigning exclusive labels to images two-sided techniques may lead to excellent performance without actually learning one of the categories. For example, successful training of a linear SVM classifier (implicit two-sided selection by  $\mathcal{R}^{(SVM)}$ ) that separates images of *cars* and *people*, may not be used to detect people on any unseen images. It can easily happen that the trained classifier relies on the absence of *cars* when it labels an image as *people*. We conclude that to choose whether one- or two-sided technique is more appropriate is task dependent. Since our experiments always have a background category, we only use one-sided, object features only, selection. This type of selection is also a key property for the localization experiments in Section 4.2.

Usually one-sided measures can easily be converted into two-sided measures, and visa-versa. [Zheng et al. \(2003\)](#) squares  $\mathcal{R}^{(OR)}$  and  $\mathcal{R}^{(GSS)}$  to introduce the two-sided  $\mathcal{R}^{(ORS)}$  and  $\mathcal{R}^{(GSSS)}$ . We turn symmetric measures, such as  $\mathcal{R}^{(MI)}$ ,  $\mathcal{R}^{(BNS)}$ ,  $\mathcal{R}^{(AB)}$ , and  $\mathcal{R}^{(CMIM)}$  one-sided by requiring

$$P(f_i|\oplus) > P(f_i|\ominus), \quad (3.2)$$

and therefore they only select features informative for the object class and not for the background. We define  $\mathcal{R}^{(SVM+)}$  by using only the weights ( $w_i$ ), i.e. omitting the absolute function ( $|\cdot|$ ) from  $\mathcal{R}^{(SVM)}$ . (Note that all binary features  $\hat{f}_i$  are positive.) Figure 3.5 on the left column shows symmetric measures, and on the right their one-sided equivalents.

[Zheng et al. \(2004\)](#) points out that it is difficult for two-sided measures to obtain the optimal combination of positive and negative features, especially with unbalanced data. [Forman \(2003\)](#) solves this by balancing the training data, while [Zheng et al. \(2004\)](#) select the two kinds of features separately and then explicitly combine them.

With this combination  $\mathcal{R}^{(CHI)}$  performs similarly to  $\mathcal{R}^{(CC)}$  when the feature set is small and the set is highly unbalanced.

### Multi-class

Similar problems appear in multi-class frameworks. [Forman \(2004\)](#) showed that there is a pitfall in feature selection methods performing independent scoring whereby they get distracted from selecting useful features for difficult classes, in the case when there is a supply of strongly predictive features for easier classes. To avoid such problems he proposed a solution inspired by a round-robin scheduling technique. [Fan and Lu \(2005\)](#) integrate linear SVM feature selection into a multi-class framework by appropriately combining the ranks of several one-vs-all problems. Their selection method outperformed traditional kernel space methods for appearance-based face recognition.

### Combination of Different Types of Features

Ranking methods offer an elegant way to combine different features, e.g., the output of different interest point detectors, or different types of region descriptors. If we assume that the source of our features are independently distributed, we can create two separate feature sets. To estimate the ranking score for the features of the different sets, we can adapt equations of different ranking methods to multiple types of features: the conditional and joint probabilities are computed for the corresponding sets, and are zero for the components of the other sets. The normalization factors  $N^\oplus$  and  $N^\ominus$  correspond to the total number of unlabeled and negative features over all types. This provides comparable ranking values for features extracted using different methods.

### Expanding the Feature Set

While most filtering techniques assume independence between features, feature sets can be expanded by constructing conjunctive features or products of features. A feature that is useless on its own can be useful when combined with others (e.g. xor problem, chessboard problem).

## 3.3 Selection for Local Features

In this section we apply the selection techniques introduced in Section 3.2 to images. First, we describe our feature set and how we estimate the probabilities  $P(f_i|\mathbf{x}_j)$ . We then evaluate the scoring methods by experiments which retrieve object features.

### 3.3.1 Visual Words

Our feature set is based on local patches extracted from images. For our experiments, images are represented by local descriptors of interest points (Section 2.1). In this

chapter, our reports mainly use the detector of [Kadir and Brady \(2001\)](#) (ENTR), together with the SIFT (Section 2.1.2) representation. However, results that can be obtained by other detectors would lead to similar conclusions. A performance evaluation of different detectors for object recognition are presented in Chapter 4.

For many vision applications, due to the diversity and high dimensionality of the descriptors it is necessary to quantize them to generate the actual features. In our approach, these features, the *visual words*, are generated as the first step of our training phase by an unsupervised estimation of a Gaussian mixture model (GMM) ([Bishop, 1995](#)) on all descriptors from our training set. We employ a parametric estimation to model the distribution of our local descriptors. Our method is based on a GMM, which is a linear combination of Gaussian densities  $p(\mathbf{x}|C_i)$

$$p(\mathbf{x}) = \sum_{i=1}^K p(\mathbf{x}|C_i)P(C_i), \quad (3.3)$$

where  $K$  is the number of Gaussian components within the mixture,  $P(C_i)$  corresponds to the mixing parameters and  $\sum_i^K P(C_i) = 1$ . The individual Gaussian components are of the form

$$p(\mathbf{x}|C_i) = \mathcal{N}(\boldsymbol{\mu}_i|\boldsymbol{\Sigma}_i), \quad (3.4)$$

where  $\boldsymbol{\mu}_i$  is a  $d$  dimensional mean vector and  $\boldsymbol{\Sigma}_i$  is the  $d \times d$  covariance matrix for component  $C_i$ . In our case  $d = 128$ , corresponding to the dimension of the SIFT descriptors.

The model parameters  $\boldsymbol{\mu}_i$ ,  $\boldsymbol{\Sigma}_i$  and  $P(C_i)$  of (3.3) and (3.4) are computed with the *expectation-maximization* (EM) algorithm ([Bishop, 1995](#)). EM is initialized with the output of  $k$ -means and at each iterative M-step we update the parameters as follows:

$$\boldsymbol{\mu}_i^j = \frac{\sum_{n=1}^N P^{j-1}(C_i|\mathbf{x}^n)\mathbf{x}^n}{\sum_{n=1}^N P^{j-1}(C_i|\mathbf{x}^n)}, \quad (3.5)$$

$$\boldsymbol{\Sigma}_i^j = \frac{\sum_{n=1}^N P^{j-1}(C_i|\mathbf{x}^n)(\mathbf{x}^n - \boldsymbol{\mu}_i^j)(\mathbf{x}^n - \boldsymbol{\mu}_i^j)^T}{\sum_{n=1}^N P^{j-1}(C_i|\mathbf{x}^n)}, \quad (3.6)$$

$$P^j(C_i) = \frac{1}{N} \sum_{n=1}^N P^{j-1}(C_i|\mathbf{x}^n), \quad (3.7)$$

where  $N$  is the number of unlabeled descriptors  $\mathbf{x}^n$ . We limit the number of free parameters in the optimization by using diagonal covariance matrices. This restriction helps to prevent the covariance matrices from becoming singular. The number of Gaussian mixture components  $K$  is chosen manually for each class based on the average number of interest points in the class. Based on our earlier experience, we select the largest possible  $K$  such that each component contains a sufficient number of descriptors

to estimate the parameters. Larger values of  $K$  permit us to represent the distribution more accurately. In our experiments (Section 4.1 and Section 3.3.2) the number of clusters  $K$  was 400 for motorbikes and airplanes, 200 for faces and bicycles, 100 for people and due to the small number of detections only 25 for leaves. The number of images used for clustering is indicated for each class in the last column of Figure 3.7.

Figure 3.7 displays for several object classes two of the ten highest ranked clusters; interest regions are detected with ENTR (Kadir *et al.*, 2004) and ranked with the likelihood ratio described in Section 3.2. We show example image regions which are most likely assigned to each cluster. We can observe that the clusters typically contain representative object parts or textures. In the case of airplanes, the nose has a very characteristic shape as does the tailplane (see Figure 3.7, first row). We also obtained significant clusters on the fuselage containing small passenger windows, and on the wing. In the case of bicycles and motorbikes, tires, wheels and tubular parts are clearly grouped and distinguished. Faces give one of the most impressive results, as left and right eyes, including the eyebrows, are clustered separately. Sometimes, if objects have very characteristic textures, their corresponding descriptors are clustered together as is the case for the wildcats (see sample cluster #1 in Figure 3.7).

During features selection (training), the probabilities  $P(f_i|\mathbf{x}_j)$  introduced in Section 3.2 are determined by their Gaussian component, and are therefore equivalent to  $P(C_i|\mathbf{x}_j)$ . However, to classify a test feature  $\mathbf{y}$ , we use a *hard assignment*.  $\mathbf{y}$  is assigned to the component  $i^*$  of the Gaussian mixture model with the highest probability:

$$i^* = \arg \max_i p(\mathbf{y}|C_i)P(C_i).$$

This rule defines a separation boundary for each component of the mixture model. Figure 3.8 shows four examples of separation boundaries based on a GMM with  $K = 8$  components. Note that the figure is just an illustration, in practice the number of components is much larger and our feature space is high-dimensional ( $d = 128$ ). We mark the  $n$  components with the highest rank as positive and construct a final classifier. A descriptor is classified as positive if its closest component (*Maximum A Posteriori*) is marked positive.

### 3.3.2 Retrieving Object Features

Here, we evaluate how well the descriptors of selected features correspond to the object class on test images.

#### Experimental Set-Up

For the following set of experiments we have used the bicycles category from the Graz1 dataset available at [http://www.emt.tugraz.at/~pinz/data/GRAZ\\_01](http://www.emt.tugraz.at/~pinz/data/GRAZ_01). For the separation of training and test images we have used the same images as Opelt *et al.*












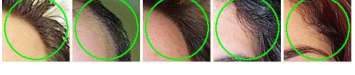




Database	Sample cluster #1	Sample cluster #2	# of images
Airplanes			200
Motorbikes			200
Leaves			46
Wildcats			50
Faces			109
Bicycles			100
People			50

Figure 3.7: Illustration of the clustering. We show 2 of the 10 best clusters for ENTR regions and likelihood ranking (see Section 3.2). The last column indicates the number of images used for the clustering (i.e. the half of the training set).

(2004). Training is weakly-supervised, i.e., training images are annotated as positive or negative, but the objects in the positive images are not marked. All object images contain a large amount of background. We have divided the training set into two halves: the clustering and the ranking set. The GMM is estimated on the clustering set, and the feature selection is performed on the ranking set. The  $n$  top ranked components correspond to positive, while the others to negative features. In the following we evaluate how many of the positively classified points lie on the object. To create the ground truth we use hand-segmented test images. We consider a selected feature as true positive if its center is located on the object.

For  $\mathcal{R}^{(SVM)}$  and  $\mathcal{R}^{(CMIM)}$  we used the implementation of  $SVM^{light}$  (Joachims, 1999) and Fleuret (2004) respectively.

### Performance Evaluation

Figure 3.9 shows the recall-precision curves for the ENTR detector and for different ranking methods. The curves are generated by changing  $n$ , the number of positive

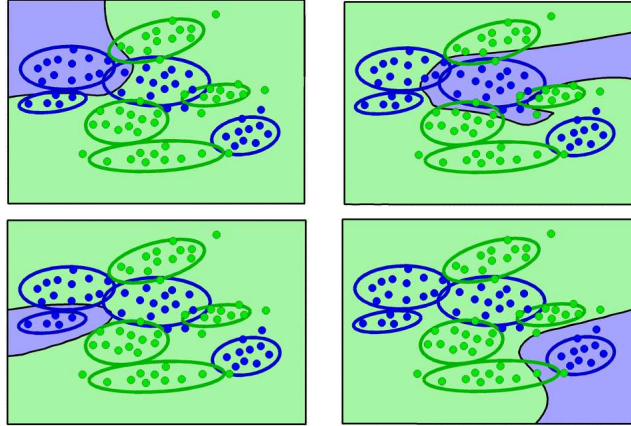


Figure 3.8: Illustration of a GMM model with  $K = 8$  components in 2-dimensions. A classifier is associated with each component. We only show separation boundaries for 4 classifiers.

Ranking	Recall				
	0.2	0.4	0.6	0.8	1.0
$\mathcal{R}^{(LIK)}$	83	75	71	66	53
$\mathcal{R}^{(OR)}$	84	75	70	66	53
$\mathcal{R}^{(BNS+)}$	84	75	71	-	-
$\mathcal{R}^{(CMIM+)}$	80	75	70	-	-
$\mathcal{R}^{(MI+)}$	82	72	70	-	-
$\mathcal{R}^{(CC)}$	82	72	70	66	53
$\mathcal{R}^{(GSS)}$	72	71	70	66	53
$\mathcal{R}^{(F_1)}$	65	66	64	66	53
$\mathcal{R}^{(SVM+)}$	66	64	60	-	-
$\mathcal{R}^{(AB+)}$	75	-	-	-	-
$\mathcal{R}^{(Freq)}$	60	60	58	54	53

Table 3.1: Precision values at selected recall levels for different ranking methods on the bicycle images using ENTR detector.

features. The highest accuracy is achieved by  $\mathcal{R}^{(LIK)}$  and  $\mathcal{R}^{(OR)}$  closely followed by  $\mathcal{R}^{(BNS+)}$ . In the legend the ranking methods are listed in the order of their performance, and Table 3.1 shows the precision at some selected recall rates.  $\mathcal{R}^{(OR)}$  and  $\mathcal{R}^{(LIK)}$  have very similar results, due to their related measures. When the positive and negative set is unbalanced (which is not the case here) we prefer  $\mathcal{R}^{(LIK)}$  because it performs a separate normalization for the two classes. As we expected, the worst result is given by  $\mathcal{R}^{(Freq)}$  performing close to chance, since it does not use the discriminative information at all. Selection methods mixing frequency and discriminative

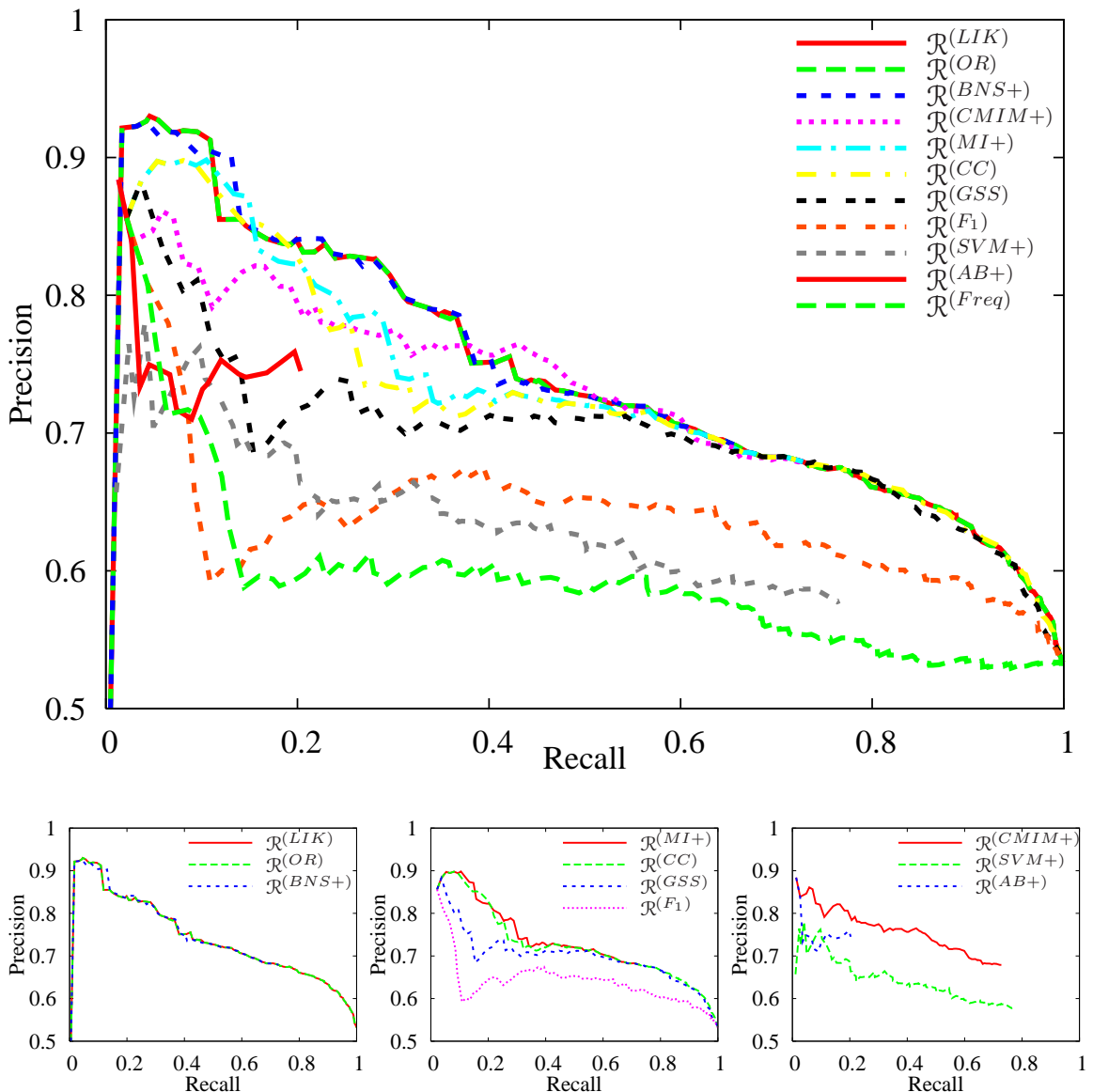


Figure 3.9: Precision-Recall curves for different ranking methods on the bicycle images using ENTR detector. In the legend the ranking methods are ordered by their performance. The row below compares three subsets of methods separately. On the left, the ones that are based mostly on discriminative power; in the middle methods combining discriminative power with frequency; and on the right the three methods that reduce feature redundancy as well.

power, have slightly lower accuracy, due to the rejection of rare “specific” features. For feature selection, more quantitative experiments will be presented in the next chapter. Figure 3.10 shows the frequency diagrams built on the actual bicycle features. The scores are smoothed over the extracted components, and therefore show which parts

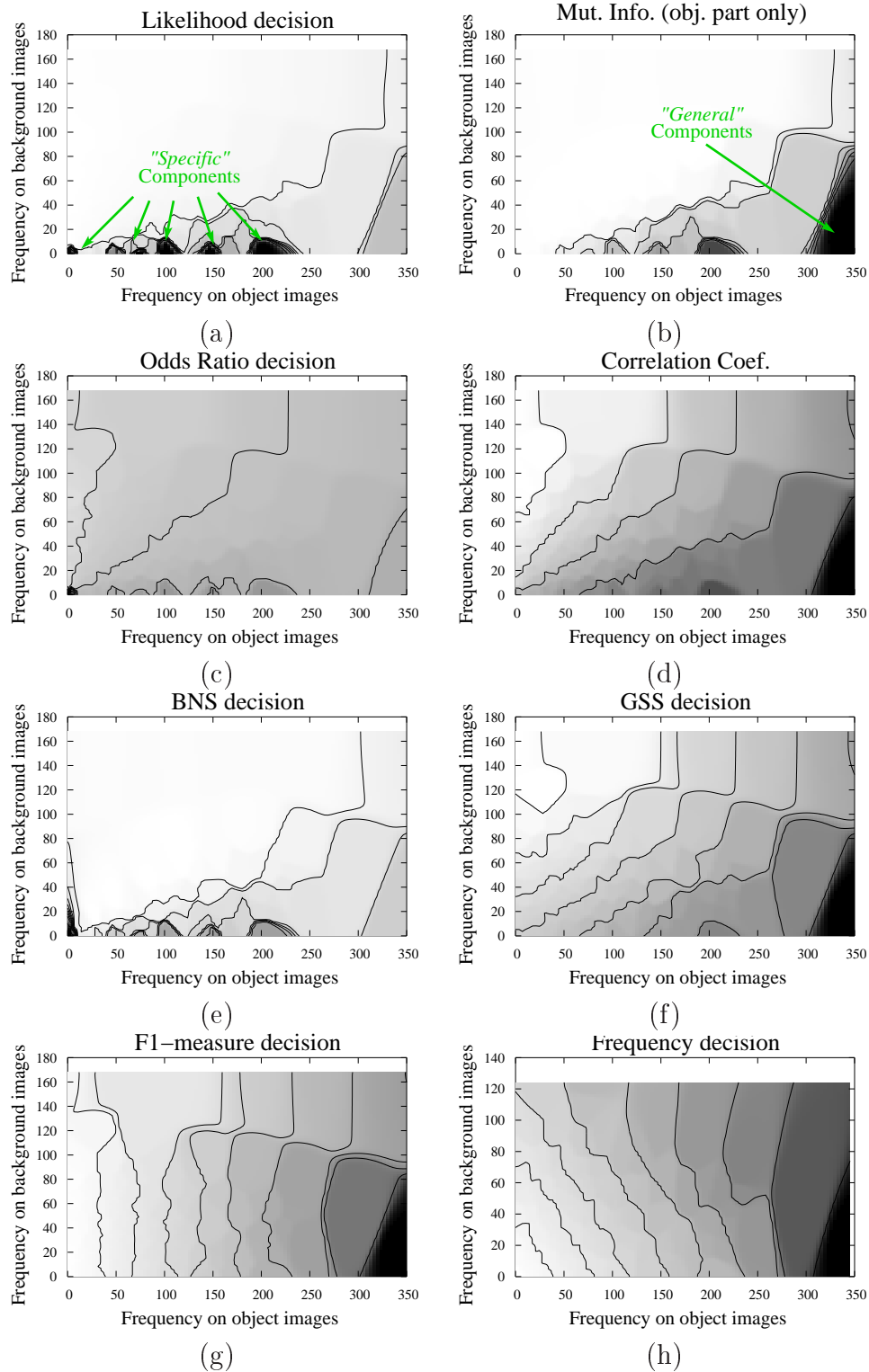


Figure 3.10: The selection scores for various ranking methods on the actual ENTR features on the bicycle dataset. Darker regions correspond to locations of features with high scores. The values are smoothed on the 200 features of the visual vocabulary. Isocontours indicate the same value within the plot. See the text for discussion.

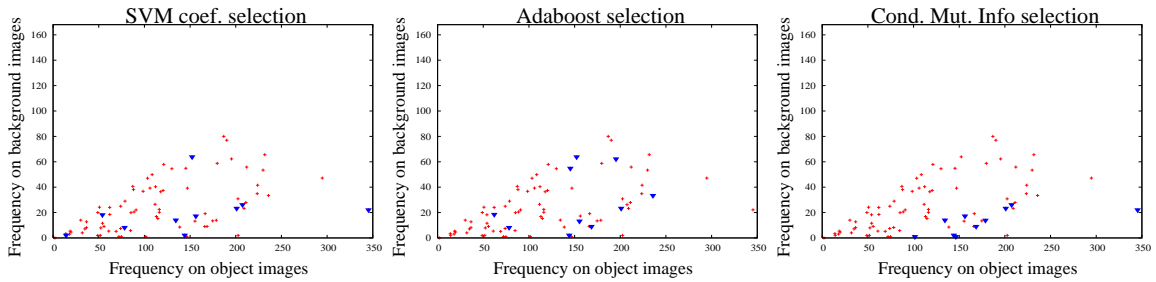


Figure 3.11: The top 10 selected features (triangles) with  $\mathcal{R}^{(SVM+)}$ ,  $\mathcal{R}^{(AB+)}$ , and  $\mathcal{R}^{(CMIM+)}$  on the bicycles dataset with ENTR features and  $K = 200$ . We also show the distribution of all positive features.

of the frequency space have components and which ones have high scores. Peaks along the horizontal axis show that  $\mathcal{R}^{(LIK)}$ ,  $\mathcal{R}^{(OR)}$ , and  $\mathcal{R}^{(BNS+)}$  ((a),(c), and (e)) prefer rare and discriminative features (cf. Figure 3.10 (a)). On the other hand, the peak in the bottom right corner indicates that frequency plays an important role for  $\mathcal{R}^{(MI+)}$ ,  $\mathcal{R}^{(CC)}$ ,  $\mathcal{R}^{(GSS)}$ , and  $\mathcal{R}^{(F_1)}$  ((b), (d), (f), and (g)).

### Performance using Only a Few Features

We have seen that with respect to accuracy of retrieving object features (cf. Figure 3.9)  $\mathcal{R}^{(MI+)}$  and  $\mathcal{R}^{(CC)}$  behave very similarly, while  $\mathcal{R}^{(GSS)}$  and  $\mathcal{R}^{(F_1)}$  are worse. The real advantage of these frequency based methods are illustrated in Figure 3.12, where we show the  $F_1$ -measure as a function of the number of selected features. When our purpose is to build a very sparse representation these methods are preferred. On the bicycles dataset,  $\mathcal{R}^{(GSS)}$  performs the best when only a few components are selected, while above 15 features  $\mathcal{R}^{(CC)}$  and  $\mathcal{R}^{(GSS)}$  swap places several times. Selection methods that reject redundant features have curves below these. The lower object coverage (recall part of  $F_1$ -measure) can indicate redundancy, but on the other hand it can also correspond to poorer features; this is verified in Section 4.1. The top 10 selected features for the bicycle category are shown in Figure 3.11.

Figure 3.13 shows the selected regions for different selection methods and varying number  $n$  of components. As we expect the top  $n$  classifiers select more regions with methods that use frequency ( $\mathcal{R}^{(MI+)}$ ,  $\mathcal{R}^{(CC)}$ ,  $\mathcal{R}^{(GSS)}$  and  $\mathcal{R}^{(F_1)}$ ). This confirms the results obtained in Figure 3.12. Among the methods that conditionally rank features ( $\mathcal{R}^{(SVM+)}$ ,  $\mathcal{R}^{(CMIM+)}$ ,  $\mathcal{R}^{(AB+)}$ ),  $\mathcal{R}^{(CMIM+)}$  selected the most descriptors in total. This again indicates the explicit preference of frequent features using mutual information.

Depending on the dataset and the visual vocabulary, different ranking methods may lead to similar ordering of the features, and therefore similar results. Using a different dataset, the *People* set from Graz1, Figure 3.15 shows a situation when the limited number of training examples fail to provide frequent and sufficiently discriminative

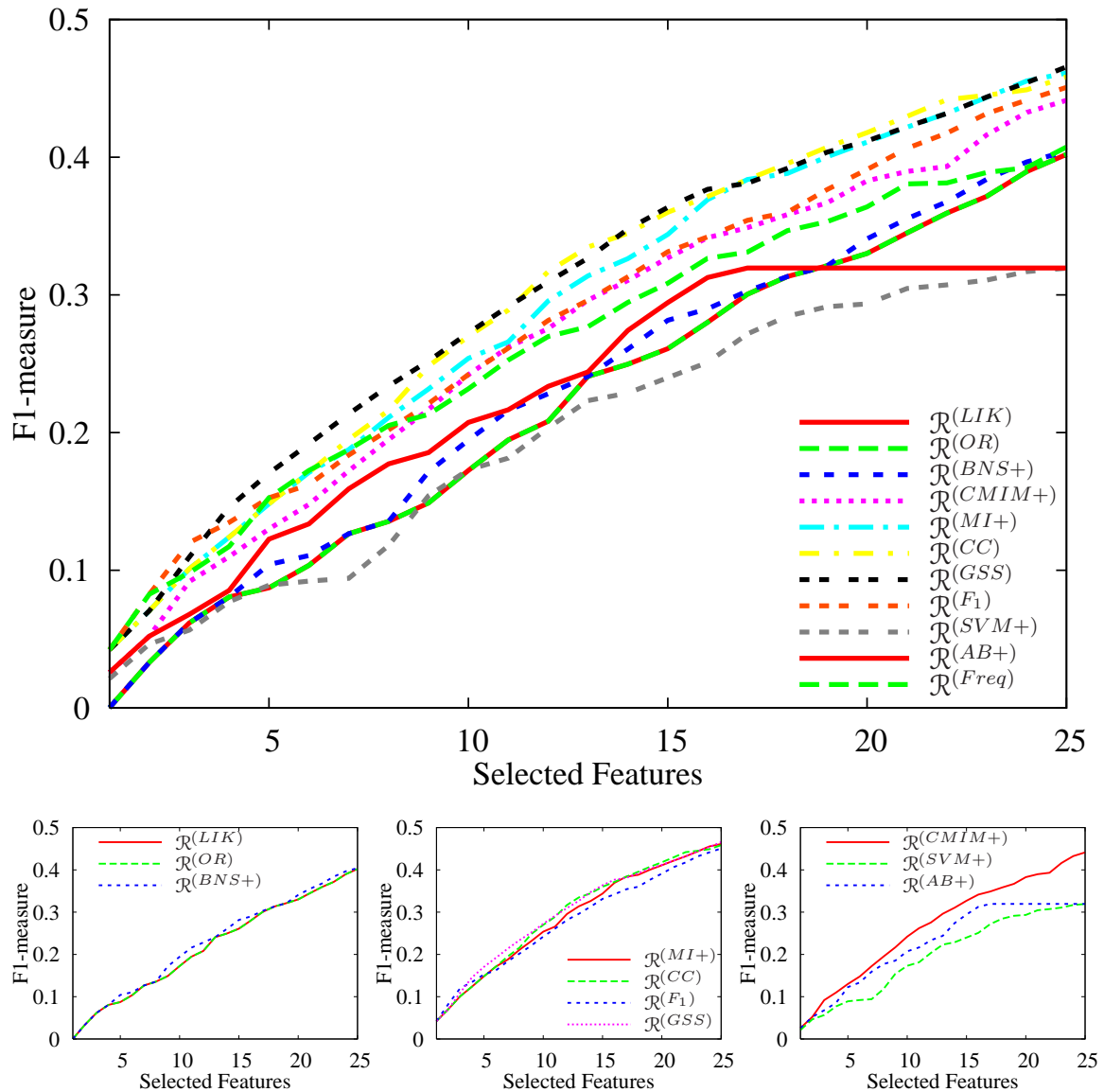


Figure 3.12:  $F_1$ -measures in the selected features for different ranking methods on the bicycle images using ENTR detector. See text for details.

features. Notice the lack of points close to the bottom right corner in the frequency diagram (a). Figure 3.15 (b and c) shows the  $\mathcal{R}^{(LIK)}$  and  $\mathcal{R}^{(MI+)}$  scores on the actual features. The similar locations of the peaks indicate that similar features are selected and therefore similar performance can be expected. We believe that in order to obtain “general” discriminative components for such a difficult class as people, more training examples are necessary.



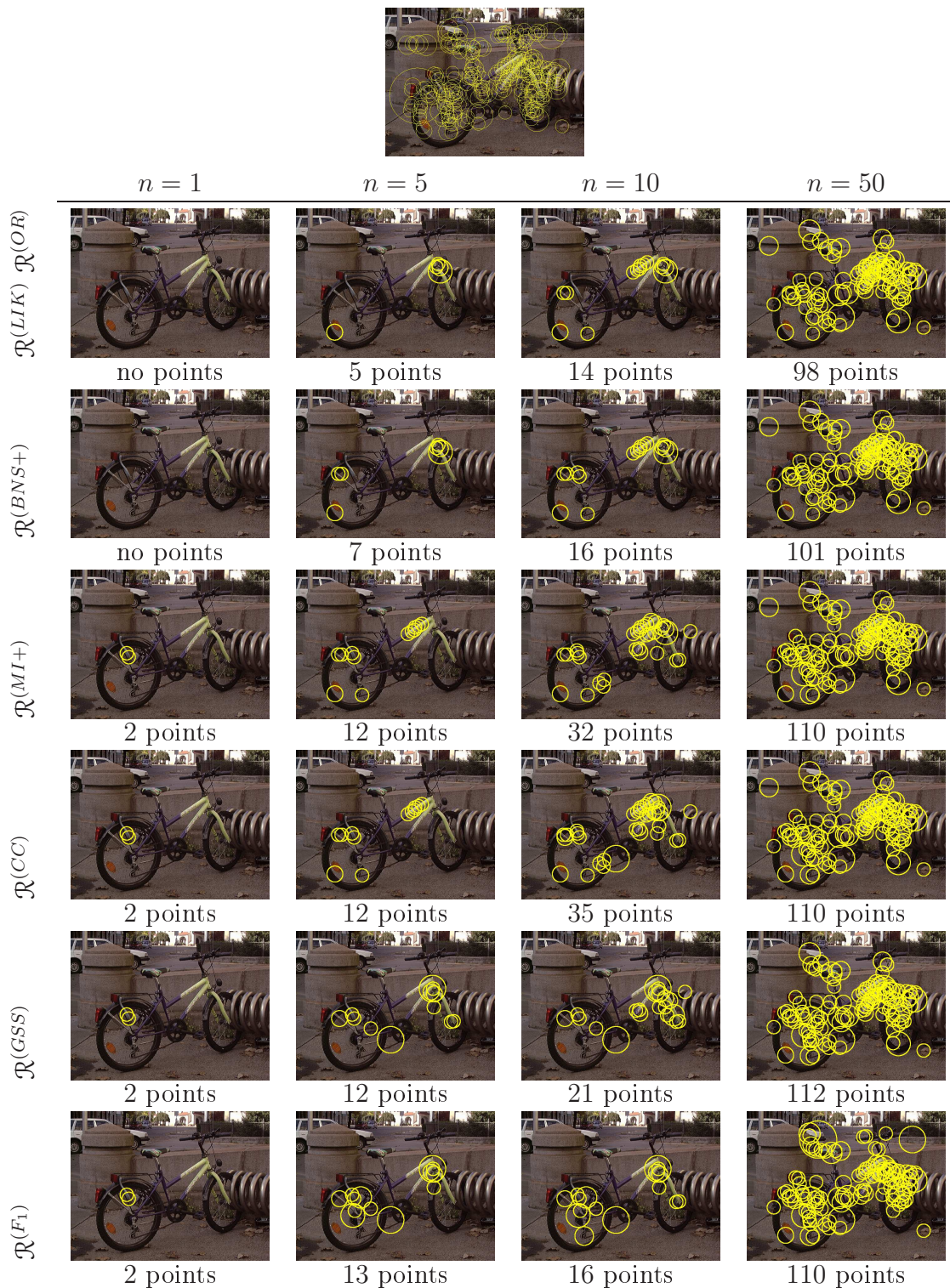


Figure 3.13: Feature selection (with non-conditional methods) for increasing  $n$  for the bicycle dataset. Regions are extracted with the ENTR detector.

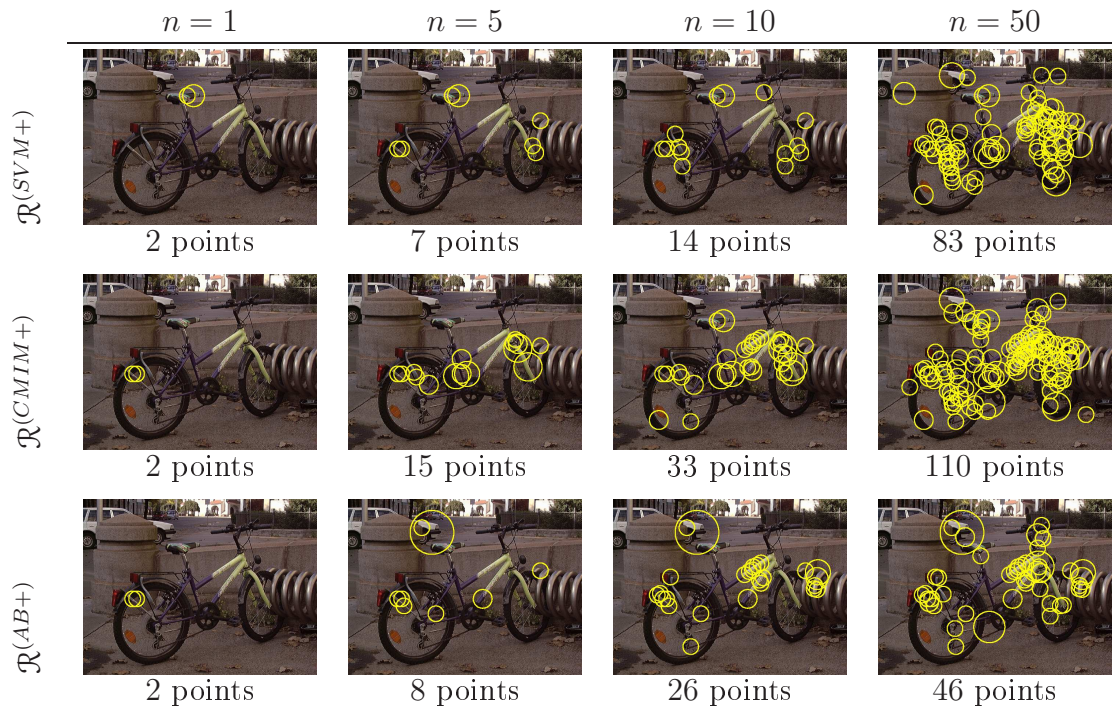


Figure 3.14: Feature selection with conditional methods for increasing  $n$  for the bicycle dataset. Regions are extracted with the ENTR detector.

### 3.4 Discussion

In this chapter, we have introduced appearance-based class-discriminative feature selection for object recognition.

Many different ranking techniques have been compared for selecting discriminative parts and dominant textures of object classes. Comparisons have shown that likelihood and odds ratios are well suited for object recognition and detection, while methods combining frequency with discriminative power, such as mutual information and chi-square are more appropriate for sparse representation and for focus of attention mechanisms (rapid localization based on a few classifiers). To further increase the sparsity of our feature set we have shown methods that reject redundant features. However, the benefits of these methods for real applications are yet to be evaluated (see Chapter 4). Table 3.2 summarizes the different properties of the discussed methods.

In this chapter have also shown how to create features (visual words) that are suitable for the selection task. Our constructed visual features are based on local descriptors, thus providing robustness to occlusion and cluttered backgrounds. In our experiments the local descriptors are partially labeled by marking their source images as positive or negative, so the demonstrated selection system is trained in a weakly-supervised fashion, while the learning of the parts (model estimation) is completely



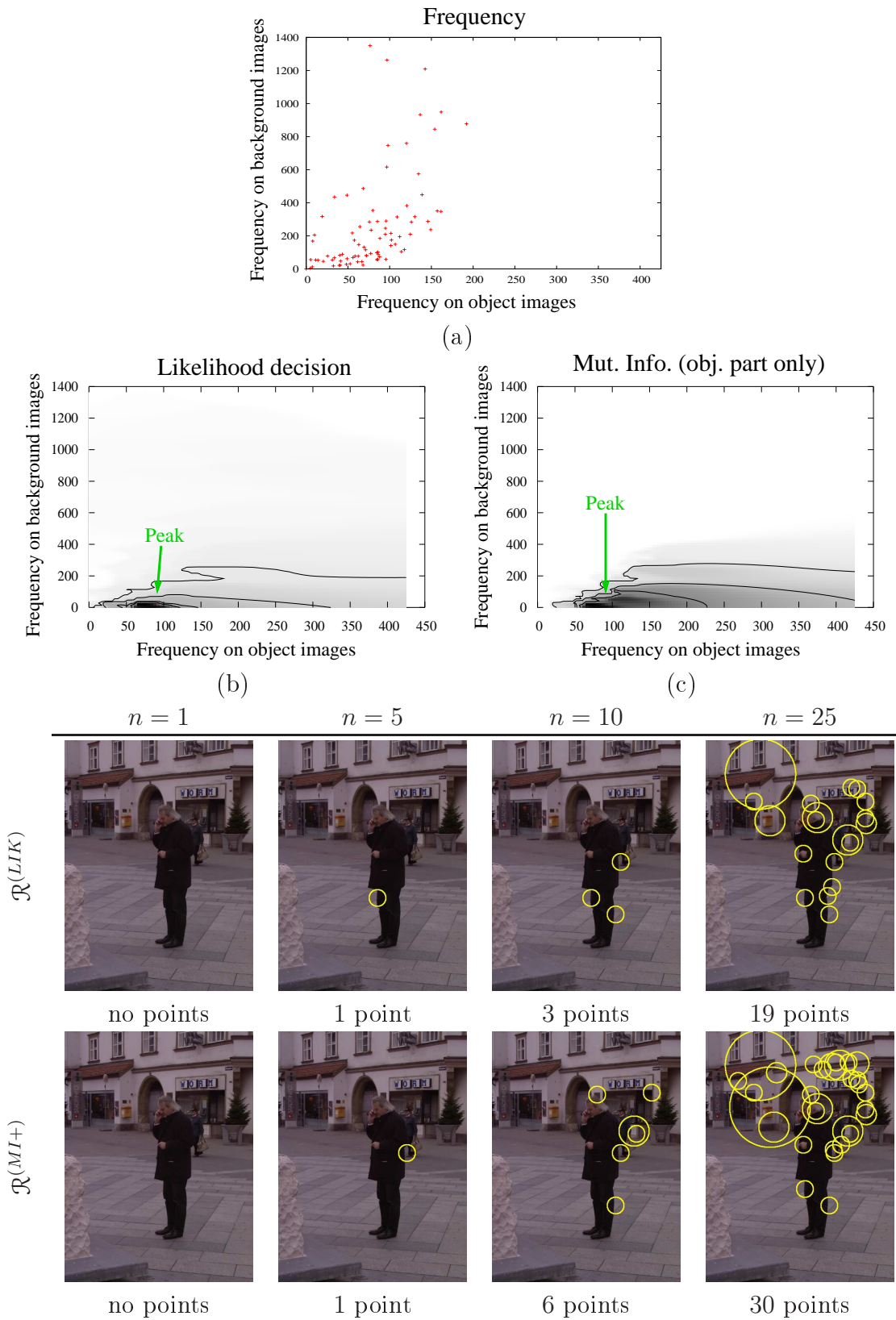


Figure 3.15: Feature selection on the people dataset. Regions are extracted with the ENTR detector. The top plot shows the distribution of all features; the next row the  $\mathcal{R}^{(LIK)}$  and  $\mathcal{R}^{(MI+)}$  selections indicate that the best ranked features are similar; and the last rows give an example image with increasing the number of selected features,  $n$ .

Ranking Method	Two Sided	Discr.	Freq.	Rej. Redu.
Frequency	✓	✗	✓	✗
Odds Ratio	✗	✓	✗	✗
Log Odds Ratio Squared	✓	✓	✗	✗
Chi Square	✓	✓	✓	✗
Correlation Coefficient	✗	✓	✓	✗
GSS	✗	✓	✓	✗
Squared GSS	✓	✓	✓	✗
$F_1$ -measure	✗	✓	✓	✗
Likelihood	✗	✓	✗	✗
Mutual Information	✓*	✓	✓	✗
Bi-Normal Separation	✓*	✓	✓ <sup>†</sup>	✗
SVM hyperplane coefficients	✗	✓	✗ <sup>‡</sup>	✓
SVM hp. coef. (absolute value)	✓	✓	✗ <sup>‡</sup>	✓
Conditional Mutual Information	✓*	✓	✓	✓
Adaboost	✓*	✓	✓	✓

Table 3.2: Summary of ranking methods and their main properities. Two sided measures selects both negative and positive features.

\*Originally a two sided measure, however with the requirement in (3.2) can be used to select only positive features.

<sup>†</sup>BNS uses frequency, but in our experiments it still selects rare features when they are overly discriminative.

<sup>‡</sup>SVM hyperplane coefficients does not explicitly use frequency, but many rare features can be rejected because of their redundancy. The rejection of rare features also depends on the generalization properties (cf. the  $c$  parameter) of linear SVMs.

unsupervised.

This chapter has already illustrated the importance of feature selection and has shown good results for descriptor classification. In the following chapter we integrate the introduced framework into appearance-based object classification and object class localization system.



# Classification and Localization of Object Classes

---

OBJECT CLASS recognition and localization are challenging problems in computer vision. The main difficulty is to design a method which can efficiently detect instances of a class under various image transformations without responding to clutter. Different instances of the category often vary in appearance or can be observed under different imaging conditions. Occlusions and highly textured background are also common factors in every day applications. In existing approaches these difficulties are addressed by the appropriate image representations and learning methods. In the last few years part-based representations have become popular, since they can deal with intra-class variation and occlusions. Some of these methods are limited to fixed size windows or require manually labeled parts (Mahamud and Hebert, 2003; Mohan *et al.*, 2001). As we show in earlier chapters, interest point detectors (Kadir *et al.*, 2004; Lindeberg and Garding, 1994; Lowe, 2004; Matas *et al.*, 2002; Mikolajczyk and Schmid, 2004b) provide an efficient way to extract informative features of different sizes and therefore, permit automatic selection of information-rich parts for local representations of images.

In this chapter, we combine local features introduced in Chapter 2 and feature selection methods discussed in Chapter 3 with state-of-the-art learning techniques from computer vision to develop framework for object class classification and localization. Our main goal is to demonstrate and to evaluate the methods introduced in the previous chapters. In this chapter we discuss two different tasks. The first one is object class classification, where the system aims to decide whether an instance of a class is present or absent in a test image. This task is often referred to image classification. The second task is object class localization, where feature selection improves an existing method estimating the exact location of class instances within test images. The two different systems share a common core: first we extract scale- and affine-invariant local features from images and construct a vocabulary of *visual words* to train a model. Then feature selection is used to order these visual words according to their discriminative power. The classification approach is “weakly supervised” in the sense that images are labeled

as positive and negative, but the objects in the positive images are not marked or segmented, and are present in arbitrary non-registered locations in cluttered scenes. The introduced system is invariant to viewpoint changes, without requiring alignment or pre-normalization of images. For the localization framework we restrict the invariance only for similarity transformations and use full supervision: object instances are marked by their rectangular bounding boxes on the positive training images. For both the classification and the localization tasks, each positive training image may contain multiple instances of the same object class in cluttered background.

### Related Work

Many state-of-the-art methods perform an exhaustive search on location and scale with a sliding window to determine the presence of an object class (Agarwal and Roth, 2002; Dalal and Triggs, 2005; Papageorgiou and Poggio, 2000; Schneiderman and Kanade, 2000; Viola *et al.*, 2003). These methods have three main disadvantages. First, they have to deal with a huge number of negative windows, and thus have to be developed for very low false positive rates. Furthermore, they usually require an additional step to reject multiple detections for the same object. And finally, searching the entire scale- and location-space with a strong classifier can be inefficient, sometimes impossible within a reasonable time. They not only require fast feature extractors, but also classifiers that can be evaluated very rapidly. The most popular classifiers are based on Support Vector Machines. Both Papageorgiou and Poggio (2000) and Dalal and Triggs (2005) use SVM within a sliding window framework. The former uses wavelets, and the latter uses histograms of oriented gradients as a representation.

Some recent methods represent the objects in a more flexible manner. Weber *et al.* (2000b) use localized image patches and explicitly compute their joint spatial probability distribution, yet does not explicitly deal with different scales. Fergus *et al.* (2003) extend their model by learning the explicit global structure of object classes based on scale-invariant image regions. While this method permits automatic part detection and object localization, the complexity of its joint probability estimations limits its applicability to a small number of parts. They only report results for image classification, and they are compared to ours in Section 4.1.3. Fei-Fei *et al.* (2003) introduce a Bayesian version of the previous model, which by incorporating priors permits the method to be trained with a limited number (1 – 5) of example images. Felzenszwalb and Huttenlocher (2000) manually build the spatial relations between parts which are stored in a tree-based structure rather than representing their full joint probability (Weber *et al.*, 2000b; Fergus *et al.*, 2003). Their efficient search for global matches in the recognition phase is recently used by Crandall *et al.* (2005) defining a simple probabilistic model with a similar performance to Fergus *et al.* (2003). Bouchard and Triggs (2005) introduce a two-layered star-based hierarchical model to allow rapid training and testing as well as soft intra-class variation of parts and sub-parts of objects. Their model can profit from the large number of detected features, and

is therefore particularly useful for objects captured at high resolutions. [Agarwal \*et al.\* \(2004\)](#) learn a vocabulary of parts, determine spatial relations for these parts, and use them to train a Sparse Network of Winnows (SNoW) Learning Architecture. [Leibe and Schiele \(2004\)](#), learn a vocabulary of local appearance and relative spatial positions of individual parts. They use a voting scheme to combine these parts and probabilistically segment unseen images. In [Section 4.2.1](#) we improve their voting scheme. We show that by integrating a discriminative feature selection, the predicted location of the voting significantly improves, owing to the elimination of votes of non-discriminative parts. Results are comparable to their full method ([Leibe and Schiele, 2004](#)) which includes verification by segmentation. Our approach does not contain any additional verification step and therefore, does not require the segmentation map of the training images.

A few recent methods using local features are available for classification tasks, i.e., deciding about the presence of an object class instance in a test image. Their main advantage is that they can profit from, and often deliver excellent results using the object features together with contextual information. Two of these approaches, [Opelt \*et al.\* \(2004\)](#) and [Willamowski \*et al.\* \(2004\)](#), have been mentioned in the previous chapters. The bag of keypoints method ([Willamowski \*et al.\*, 2004](#)) was used as a baseline approach for evaluation in [Section 2.4](#). [Winn \*et al.\* \(2005\)](#) extend the previous method by refining the visual vocabulary. They automatically determine the size of the vocabulary by merging elements of an initially large dictionary. This permits to produce a more compact, yet still discriminative representation. The bag-of-keypoint representation is also used by [Sivic \*et al.\* \(2005\)](#). They represent object categories by topics determined with probabilistic Latent Semantic Analysis (pLSA) and show that simple categories such as faces, motorbikes, airplanes, and cars can be separated automatically. [Opelt \*et al.\* \(2004\)](#) use AdaBoost to select discriminative features and build a *strong classifier* for image classification. We compare our results with [Willamowski \*et al.\* \(2004\)](#) as well as [Opelt \*et al.\* \(2004\)](#) in [Section 4.1.3](#).

## Overview

In [Section 4.1](#) we present an approach for object class classification. We compare different interest point detectors and evaluate the selection methods from [Chapter 3](#). [Section 4.2](#) introduces an approach for object localization. We demonstrate how to improve the performance of a state-of-the-art method by feature ranking and selection. We show results on three different, recently proposed and widely used datasets.

## 4.1 Object Class Classification with Discriminative Features

Image classification—often used as evaluation criterion in the literature—decides if an object is present or absent in an image. In this section, we build a simple classifier

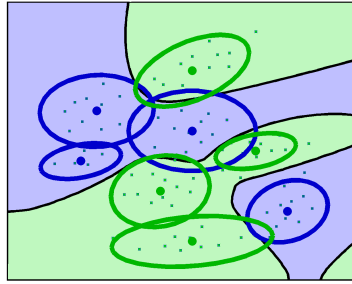


Figure 4.1: The final classifier, a Gaussian mixture model with  $K = 8$  components, four of which are selected. See Figure 3.8 for the individual classifiers. The separation boundary indicates if a test feature is classified as positive (object) or as background.

based on different feature rankings. This allows us (1) to show the efficiency of discriminative feature selection and (2) to evaluate the discriminative quality for each selection method. Experiments in this section are performed on two-class problems, i.e., object vs. background. The extension to multi-class is straightforward; a set of two class classifiers can be constructed, where each one is trained for a given object class.

#### 4.1.1 Classifier for Objects Presence

In Chapter 3 we have built a classifier for each feature (equivalent to a Gaussian component). We have seen that the ranking order of features reflects their discriminative power for a given category. By marking the  $n$  components with the highest rank as positive (cf. Chapter 3), a *final classifier* (see Figure 4.1) can be constructed. A descriptor is classified as positive if its closest component is marked positive. Note that this classifier may act as an initial step for localizing an object (see Figure 4.2). However, to make a decision on the existence of an object, an additional condition is required. In the following we classify an image as positive, if there are at least  $p$  positive detections, i.e., at least  $p$  regions assigned to the  $n$  selected components. This number  $p$  is automatically determined from the training set and  $n$  is the only parameter of our method. The parameter  $p$  depends on the number of selected components  $n$ , the feature type, and the appearance of the object class. If an object class contains a few unique discriminative components, i.e., can be described by a few *visual words*,  $p$  is low. Examples are the faces and the leaves categories. On the other hand, in the case of texture-like object classes, such as wildcats, the most discriminative components are *textons*, which appear multiple times on the object, and therefore  $p$  is high.

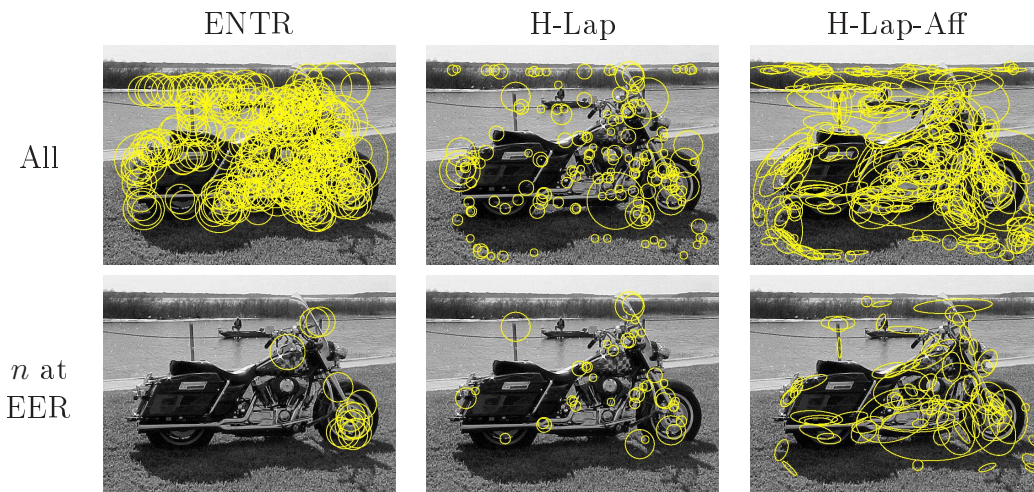


Figure 4.2: Feature selection with  $\mathcal{R}^{(LIK)}$  for the detectors ENTR, H-Lap and H-Lap-Aff. The top row shows the detected interest regions. The bottom row displays the regions corresponding to the  $n$  highest ranked components. The parameter  $n$  is chosen at the equal error rate point of the ROC curve. This example demonstrates that feature selection can act as an initial step for recognition and localization by identifying discriminative object parts.

### 4.1.2 Experimental Set-Up

For our recognition experiments we have used seven categories, see Figure 4.3. The categories airplanes, faces, motorbikes, and leaves are from the Caltech dataset. Training and test images are the same as in (Fergus *et al.*, 2003; Weber *et al.*, 2000a), but we have added half of the background images to our training set. The Caltech dataset may be downloaded from <http://www.robots.ox.ac.uk/~vgg/data.html>, and the wildcats are from the Corel Image Library. The categories bicycles and people are from the Graz1 dataset and available at [http://www.emt.tugraz.at/~pinz/data/GRAZ\\_01](http://www.emt.tugraz.at/~pinz/data/GRAZ_01). We use exactly the same training and test images as (Opelt *et al.*, 2004). Note that this dataset is more challenging than the Caltech dataset, as it contains significant changes in viewpoint and scale as well as large amounts of background clutter. Furthermore, the intra-class variation of people is high due to the changes in clothing and pose.

Note that there is a bias in the Caltech dataset, as there are significantly more interest points for images of motorbikes, airplanes, and faces as for their corresponding Caltech background. This potentially influences the classification results. Appendix A examines the influence of the number of interest points on image classification and shows that our method does not rely more than others on this bias.

Let us recall that the training is weakly-supervised, i.e., training images are annotated as positive or negative, but the objects in the positive images are not marked.



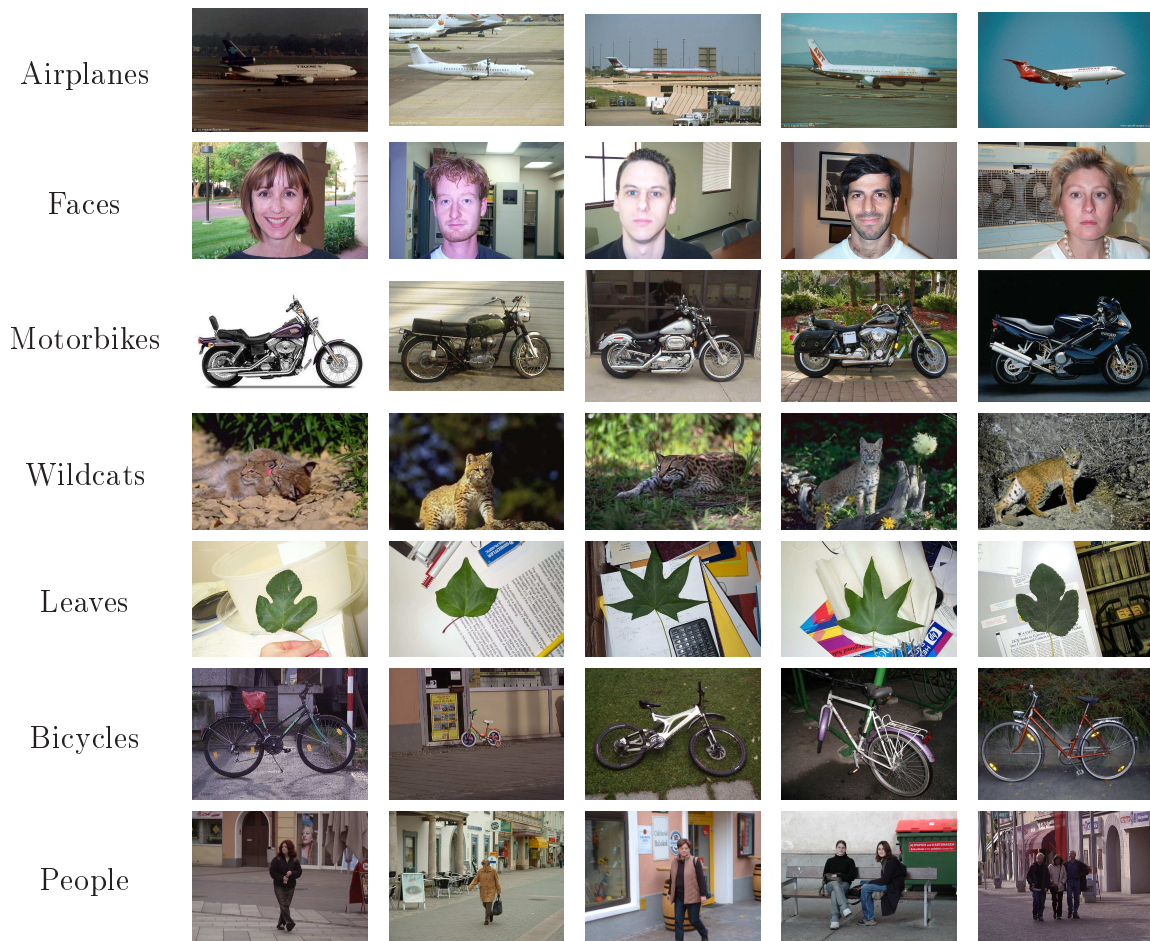


Figure 4.3: Images examples of the different categories used in our experiments.

We have divided the training set into two sets: the clustering and the ranking set. The GMM is estimated with the clustering set, and feature selection is performed with the ranking set. The minimum number of positive detections  $p$  is also determined on the ranking set. Half of the positive images are randomly assigned to the clustering set, the other half to the ranking set. All negative images are assigned to the ranking set. In general, we do not assign negative images to the clustering set, as the background clutter in the positive images allows to form negative clusters. The only exception is the bicycles dataset where all negative images contain people and the positive images do not. In this case we have added half of the negative training images to our clustering set. Note that the results are very similar if the entire training set is used for both clustering and ranking.

Receiver Operating Characteristic (ROC) curves measure the performance as the rate of correct detections with respect to the incorrect ones. To compare ROC curves we report their equal error rates, i.e., the points on the curve for which the rate of

true positives and true negatives are equal:  $p(\text{True Positive}) = 1 - p(\text{False Positive})$ , where

$$p(\text{True Positive}) = \frac{\text{Correctly classified positive images}}{\text{Total number of positive images}},$$

and

$$p(\text{False Positive}) = \frac{\text{Incorrectly classified negative images}}{\text{Total number of negative images}}.$$

### 4.1.3 Experiments: Image classification

In the following we first evaluate the performance of the individual detectors and then compare the different selection criteria introduced in Chapter 3. Finally, we compare our approach to existing results in the literature. In this section, for image classification we use the method described in Section 4.1.1, and refer to it as our *simple classifier*.

#### Comparison of detectors

The results for different detectors are summarized in Table 4.1. We compared them using both our simple classifier and the bag of keypoints method. On average our new detector, H-MSLSD performs the best, closely followed by ENTR, L-MSLSD, H-Har, and H-Lap. Apart from the leaves dataset—which we address later—these two Harris-based detectors, not surprisingly, show similar behavior. On the other hand, H-Lap performs better than ENTR for three categories, while ENTR is better than the Harris based detectors for four categories. This confirms that ENTR and H-Lap are complementary. The performance of H-Lap-Aff is similar to H-Lap, yet in most of the cases slightly below. This can be explained by the relatively small viewpoint changes in our datasets, and by the instability of the affine adaption process.

When further analyzing the results, the largest difference between ENTR and H-Lap detectors can be observed for leaves. The performance of H-Lap is exceptionally low, as only a very few H-Lap points are detected on the leaves, and most of those detections lie on the border of the objects, i.e., the characteristic regions contain a significant portion of background, see Figure 4.4 for an example. Figure 4.5 plots the equal error rate with respect to  $p$  for the two detectors showing the difference between H-Lap and ENTR.

For the bicycles the ENTR detector performs better than H-Lap, which can be explained by ENTR’s good performance for the discriminative tire regions. Figure 4.6 shows that ENTR detects a large number of regions around the tire. For bicycles the results for H-Lap-Aff are significantly worse than H-Lap, because the affine estimation adjusts the ellipse on the background between the spokes or on rich texture right next to the tire and other tubular parts.

DoG and LoG detectors, apart from a few exceptions, have similar results. The two blob-like detectors outperform on average H-Gen and MSER. Unfortunately the IBR

Classification with estimated required parts ( $p$ )								
Detector	Airplanes	Faces	Motorbikes	Wildcats	Leaves	Bicycles	People	Avg.
H-Lap	97.25	99.07	98.00	91	65.60	84	76	87.27
H-Lap-Aff	96.00	<b>100</b>	98.28	92	68.82	64	74	84.73
H-Har	96.50	99.54	97.25	92	93.55	76	78	90.41
H-Gen	94.00	91.71	96.00	79	65.59	74	60	80.04
ENTR	96.00	96.77	<b>98.50</b>	80	<b>98.92</b>	<b>90</b>	<b>80</b>	91.46
LoG	94.75	95.85	97.50	82	93.55	70	68	85.95
DoG	95.00	99.08	97.00	84	86.02	72	74	86.73
IBR	-	-	-	-	-	84	66	75.00
MSER	87.00	83.41	92.25	92	74.19	76	24	75.55
H-MSLSD	<b>98.25</b>	99.08	97.75	<b>93</b>	93.55	84	78	<b>91.94</b>
L-MSLSD	94.25	98.15	<b>98.50</b>	82	94.62	82	56	86.50

Bag of Keypoints								
Detector	Airplanes	Faces	Motorbikes	Wildcats	Leaves	Bicycles	People	Avg.
H-Lap	97.75	<b>100</b>	98.25	92	77.42	92	<b>86</b>	91.92
H-Lap-Aff	96.25	<b>100</b>	98.00	92	80.65	88	78	90.41
H-Har	97.75	<b>100</b>	97.25	94	93.55	86	78	92.36
H-Gen	97.00	95.39	97.75	82	81.72	88	72	87.69
ENTR	98.25	97.24	<b>99.00</b>	83	<b>97.84</b>	<b>94</b>	76	92.19
LoG	98.75	98.16	98.75	86	92.47	90	78	91.73
DoG	<b>99.50</b>	<b>100</b>	98.50	<b>96</b>	90.32	92	74	92.90
IBR	-	-	-	-	-	88	80	84.00
MSER	91.50	85.32	98.50	93	82.80	84	72	86.73
H-MSLSD	99.25	99.53	98.50	<b>96</b>	95.69	<b>94</b>	<b>86</b>	<b>95.57</b>
L-MSLSD	98.75	99.08	98.75	86	95.70	92	80	92.90

Table 4.1: Comparison of different detectors. Equal-error-rates for likelihood ranking. Results are shown for two classifiers: the simple decision the estimated  $p$  (feature selection), and the general bag of keypoints approach.

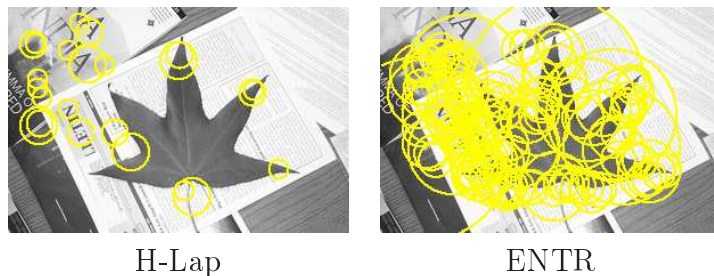


Figure 4.4: H-Lap and ENTR detections for a leaf image.

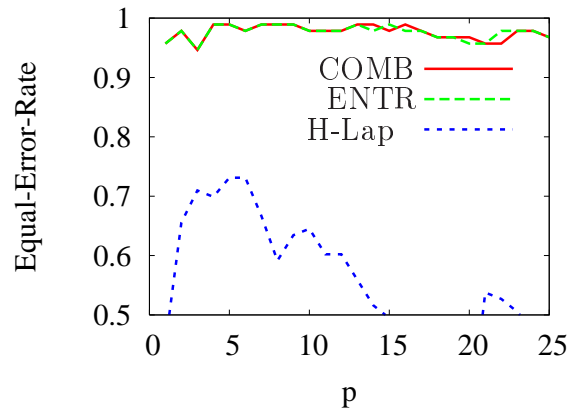


Figure 4.5: Classification accuracy (true positive rate) at the equal-error rate for the leaves.

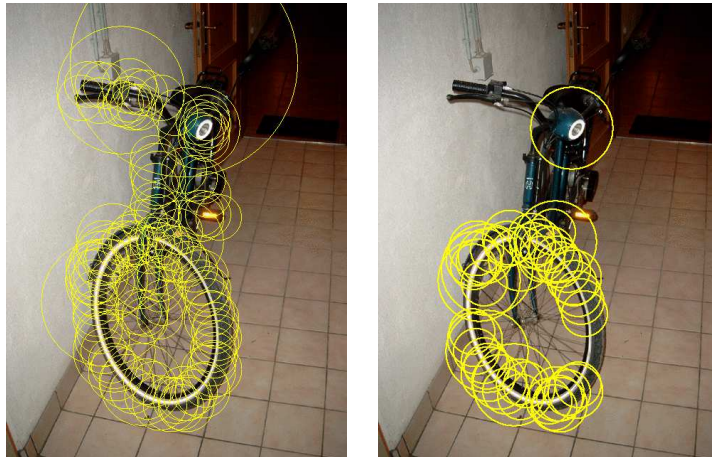


Figure 4.6: Selection results on the bicycle database. The ENTR detector output is shown on the left, and the selected discriminative features are shown on the right.

detector does not give any response on many images from the CalTech background set (due to their sizes), and therefore we have excluded it from those comparisons. On the Graz datasets IBR has not reached the top rates, but compares favorably with many others.

We have verified that results with estimated  $p$  are only slightly lower than the best achievable results, if  $p$  would have been estimated on the test set. This confirms that the estimation of the parameter  $p$  is robust.

### Comparison of different ranking methods

Table 4.2 compares feature rankings with different methods introduced in Chapter 3. The EER results are given for five best detectors (cf. Table 4.1). For the bicycles these

Ranking	Bicycles					
	ENTR	H-Lap	H-MSLSD	IBR	L-MSLSD	RND
$\mathcal{R}^{(LIK)}$	90	<b>84</b>	<b>84</b>	84	82	<b>80</b>
$\mathcal{R}^{(OR)}$	90	<b>84</b>	<b>84</b>	84	82	<b>80</b>
$\mathcal{R}^{(BNS+)}$	88	82	<b>84</b>	82	<b>84</b>	70
$\mathcal{R}^{(CMIM+)}$	<b>92</b>	80	<b>84</b>	<b>86</b>	78	72
$\mathcal{R}^{(MI+)}$	80	80	70	84	78	46
$\mathcal{R}^{(CC)}$	80	80	72	<b>86</b>	60	64
$\mathcal{R}^{(GSS)}$	80	76	74	82	76	70
$\mathcal{R}^{(F_1)}$	80	70	60	72	70	70
$\mathcal{R}^{(SVM+)}$	<b>92</b>	72	78	74	74	72
$\mathcal{R}^{(AB+)}$	<b>92</b>	78	76	82	<b>84</b>	76
$\mathcal{R}^{(Freq)}$	80	62	66	70	62	72

Ranking	People					
	ENTR	H-Har	H-MSLSD	H-Lap	LoG	RND
$\mathcal{R}^{(LIK)}$	80	<b>78</b>	78	76	<b>68</b>	84
$\mathcal{R}^{(OR)}$	80	<b>78</b>	78	76	<b>68</b>	84
$\mathcal{R}^{(BNS+)}$	74	<b>78</b>	<b>82</b>	74	66	84
$\mathcal{R}^{(CMIM+)}$	78	72	74	<b>80</b>	66	<b>86</b>
$\mathcal{R}^{(MI+)}$	<b>82</b>	76	<b>82</b>	76	64	84
$\mathcal{R}^{(CC)}$	76	76	<b>82</b>	78	64	84
$\mathcal{R}^{(GSS)}$	74	<b>78</b>	80	72	64	84
$\mathcal{R}^{(F_1)}$	54	52	48	54	42	62
$\mathcal{R}^{(SVM+)}$	64	68	68	62	64	<b>86</b>
$\mathcal{R}^{(AB+)}$	76	74	74	76	56	80
$\mathcal{R}^{(Freq)}$	22	26	38	28	36	56

Table 4.2: Comparison of different ranking methods on the Graz1 dataset. Reports are recognition rates at EER for classifiers with estimated  $p$ .

are ENTR, H-Lap, IBR and the new detectors from Chapter 2, and for the people dataset ENTR, H-Har, H-MSLSD, H-Lap and LoG. Ranking methods are listed (from top to bottom) based on their object feature retrieval performance for the ENTR detector for the bicycles database (cf. Figure 3.9). In overall,  $\mathcal{R}^{(LIK)}$  which is the best for retrieval, performs also well for classification. Large improvements compared to the experiments of Section 3.3.2 can be observed for  $\mathcal{R}^{(SVM+)}$ ,  $\mathcal{R}^{(AB)}$ , and  $\mathcal{R}^{(CMIM+)}$ . These selection methods compare favorably to  $\mathcal{R}^{(LIK)}$ . This also clarifies and confirms that these methods reject the discriminative features which are considered redundant for the discrimination task. We have two additional remarks concerning these three

methods. First, these methods have not been compared before, not even in the context of text classification, and our test shows remarkably good performance for  $\mathcal{R}^{(CMIM+)}$ .  $\mathcal{R}^{(CMIM+)}$  always outperforms or gives the same results as the other two. Second, these methods usually select less features for comparable performance than  $\mathcal{R}^{(LIK)}$ . However, this seems to be dependent on the number of features and the dataset. Several times the number of selected components ( $n$  with fixed  $p$ ) are similar to  $\mathcal{R}^{(LIK)}$ , possibly indicating no irrelevant discriminative features.

In the last column of the tables we compare the selection methods on randomly chosen points leading to similar conclusions. Random points are an indiscriminatively selected subset of 100 regions per image from the entire collections of regions at all scales and locations. As a surprise, random points on the people database performed better than ENTR, because interest point detectors often miss important features on people. For better results on this dataset probably the detector thresholds need to be adjusted. As a remark for the randomly selected patches, they may provide sufficient coverage for appearance based bag of features like representation and recognition, but to further incorporate them to use spatial relations, such as in Section 4.2, is much more challenging if not impossible.

The discussed feature selection methods with linear SVM classifiers on these seven databases have never improved our results, i.e., without exception we always achieved the best performance when all features are used in the classifier. This is due to the implicit feature selection of linear SVM ( $\mathcal{R}^{(SVM)}$ ).

### Combination of detectors and comparison with existing methods.

In this section we perform image classification experiments with the combination of two complementary detectors, H-Lap and ENTR. Table 4.3 shows the performance of the individual detectors as well as their combination on the seven object databases. As expected, a combination of detectors gives overall better performance than the individual ones. For motorbikes, airplanes, faces, and people it improves the individual results and for leaves it selects features from the better detector leading to the same results. First, we can observe that ENTR + H-Lap gives better results than each of the individual detectors, if H-Lap and ENTR perform about equally well and both have “good” discriminative components, see Figure 4.7. The combination of detectors also shows reduced sensitivity to the choice of  $p$ , and provides a useful protection against detectors that perform poorly on certain databases. Figure 4.5 shows that the COMB curve almost strictly follows the ENTR one, and in Table 4.3 COMB gives exactly the same results as ENTR alone. However, combining detectors does not always lead to improved results. In some cases poor quality of detection and additional noise may result in an overall performance in between the individual ones. An example is the wildcats category, for which the combination performs worse than H-Lap, but better than ENTR.

Table 4.3: Equal-Error-Rates for H-Lap + ENTR (COMB) and likelihood ranking.

Database	$\mathcal{R}^{(LIK)}$ ranking				Others
	Individual		COMB		
	H-Lap	ENTR	$p$	%	%
Databases with CalTech background					
Airplanes 	97.25	96.00	28	<b>98.5</b>	94.0 (Fergus), 88.9 (Opelt), 96.25 (Willamowski)
Faces 	99.07	96.77	29	<b>99.53</b>	96.8 (Fergus), 93.5 (Opelt), 100 (Willamowski)
Motorbikes 	98.00	98.50	24	<b>99.5</b>	96.0 (Fergus), 92.2 (Opelt), 98 (Willamowski)
Wildcats 	<b>91.0</b>	80.0	13	87.0	90.0 (Fergus), 92.0 (Willamowski)
Leaves 	65.60	<b>98.92</b>	8	<b>98.92</b>	84 (Weber), 80.65 (Willamowski)
TU-Graz1 Databases					
Bicycles 	84	<b>90.0</b>	14	88.0	86.5 (Opelt), 88.0 (Willamowski)
People 	76	80.0	13	<b>88.0</b>	80.8 (Opelt), 78.0 (Willamowski)

To compare our approach with existing methods, Table 4.3 also presents the results reported by their authors (Fergus *et al.*, 2003; Opelt *et al.*, 2004; Weber *et al.*, 2000a; Willamowski *et al.*, 2004). Only in the case of (Willamowski *et al.*, 2004) we have reimplemented the method to report comparable results on the same datasets. We can see that overall our method performs the best. However, we have run the bag-of-keypoints + linear SVM method (Willamowski *et al.*, 2004) on exactly features and vocabulary as our classifier, see Table 4.1 (bottom). This shows that SVM can outperform our simple classifier. However, the difference is in general not very large, i.e., around 5 – 6%. It is also important to emphasize that our classifier only selects object features, while BoK uses a two-sided selection mechanism. We conclude that many times, when only the decision whether an object is present or absent is important, an SVM classifier is the best solution, and there is no need to precede the learning



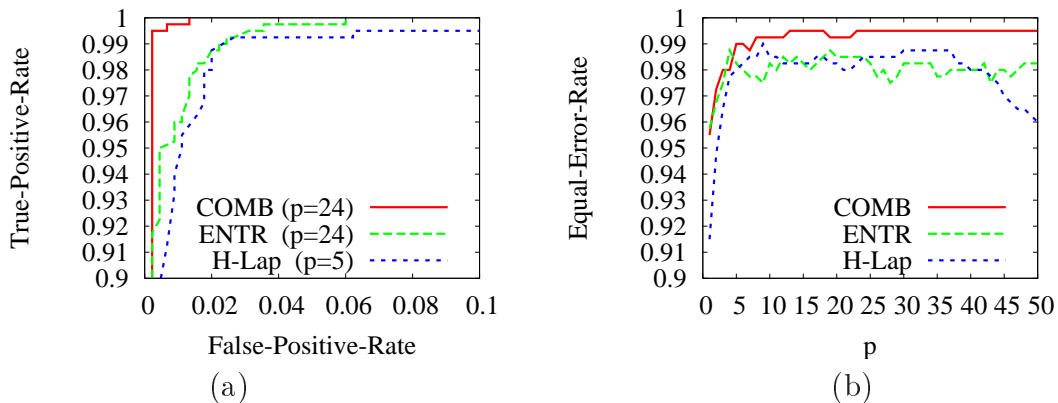


Figure 4.7: On the left, the ROC curves of different detectors for the motorbikes and “estimated  $p$ ”. On the right, the equal-error-rate curves for varying  $p$ .

by an additional feature selection. However, when we plan to extend our system, an explicit selection may still help to *understand* the features and improve performance. The next section shows an example.

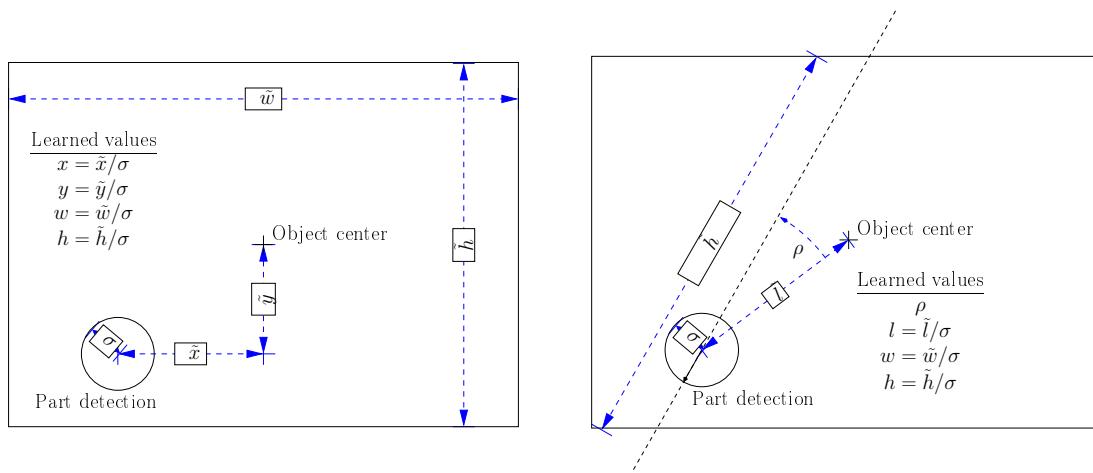
## 4.2 Object Localization with Discriminative Features

In this section we address the problem of object localization, which aims to determine the presence and exact locations of objects. Even though feature rankings can often correctly localize pieces of objects, there is no straightforward way to find the boundaries or bounding rectangles around the objects (cf. Figure 4.2 and 4.6). If we have more supervision by marking the locations on training images, spatial constraints can be learnt corresponding to the structure of the object class. In this section we show how to integrate feature selection into an existing system proposed by [Leibe and Schiele \(2004\)](#). Section 4.2.1 briefly describes the approach, the integration, as well as an extension for rotation invariant learning and localization. Section 4.2.2 discusses several parameters, the effect of selection, and experimentally evaluates those on a popular benchmarking dataset from [Agarwal and Roth \(2002\)](#), and on the bicycle dataset used earlier in Sections 4.1.2, 2.4, and 3.3.2. Our results on the PASCAL Visual Object Class Challenge (VOC2005) are summarized in Section 4.2.3. Finally Section 4.2.3 validates our method on butterflies taken under various viewpoints.

### 4.2.1 The Localization Approach

In this section we describe our approach for localizing object classes. The method can be divided into two parts: training and testing, preceded by the feature extraction step, which is detailed in Section 3.3.1. In the following we discuss the training and the localization steps separately.





(a) Scale invariant

(b) Scale and rotation invariant

Figure 4.8: At learning stage, for each detected part 4 properties are calculated according to the bounding box. (a) shows the properties in case of scale invariance learning, while (b) assumes both scale and rotation invariance. See text form more detail.

## Training

Our training consists of three steps. First we learn a vocabulary from the scale-invariant features (Section 3.3.1) and similarly to the image classification task we assign a rank to each cluster based on its discriminative power on the training data. Our criterion that we use in this section is  $\widehat{\mathcal{R}}^{(LIK)}$ . From Section 3.2, recall that the advantage of using this score over the classification likelihood ( $\mathcal{R}^{(LIK)}$ ) is that we can easily integrate it into probabilistic systems because its values lie within the range 0 to 1. After the ranking we learn a spatial distribution of the object positions and scales for each feature (cluster). For each training image, we assign all descriptors inside an object bounding rectangle to its feature (by MAP), and record the center  $(x, y)$  and the scale (width  $w$  and height  $h$ ) of the rectangle with respect to the related feature; see Figure 4.8(a). This step is equivalent to (Leibe and Schiele, 2004) with the difference that we collect the width and height separately, and that we do not require nor store any information of the figure-ground segmentation of the object.

A straightforward way to impose **rotation invariance** during training would be to learn the spatial direction of the center relatively to a given direction for the object (e.g. the direction of the head in the case of people). To be able to handle rotated objects on test images, the distribution of the main direction has to be learned additionally for each feature by taking into account an estimated main gradient direction for each patch. This would not only increase the dimension of our parameter space to 5 ( $x$ ,  $y$ , width, height and object orientation) but also require additional labeling

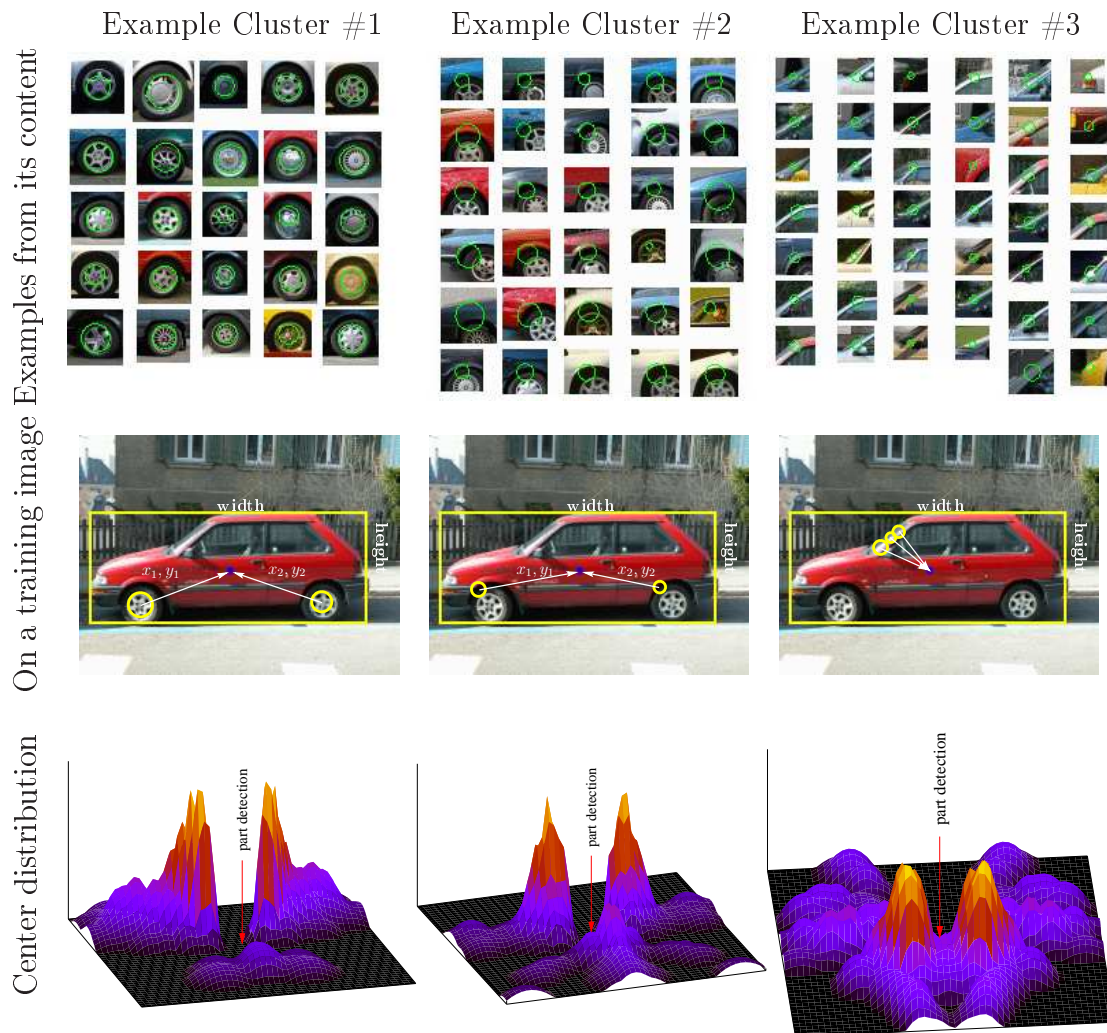


Figure 4.9: Three examples of clusters using *cars* as object class for training. First line shows example patches of the chosen cluster. The presence of these clusters on one training image is shown in the second line. The third line plots the learned distribution of the object centers relative to the presence of the clusters (marked with an arrow in the middle of the graph). For the estimation we used many car images and for visualization the density function was projected (to two dimensions lower) to the location space by taking the maximum values over dimensions of object width and height.

from the user. Unfortunately, in most of the rotation invariant applications this main object direction is not available. As an example Figure 4.13 shows some representative training data of butterflies with simple bounding-boxes. Notice, that we do not have any information about the orientation of the butterflies within the bounding boxes. We therefore propose another solution. We could call the following approach quasi-

rotation invariant, but for the sake of simplicity, we refer to it as rotation invariant in the rest of this chapter. Figure 4.8(b) illustrates the 4 dimensions (note same complexity) of our parameter space with rotation invariance. The relative center position is still normalized by the scale ( $\sigma$ ) of the descriptor, and now we also transform it according to the dominant gradient estimated prior to the descriptor computation (marked with a black solid arrow within the part detection). In this case the relative center position is described by  $\rho$  (relative direction) and  $l$  (relative distance). The relative width and height are additionally projected on the line defined by the estimated gradient. (We omitted the projection of the width from the figure, since it is too large, but it can be done similarly to the height.)

As a summary, the output of the training phase is a list of features with the following properties:

- the mean and variance representing the appearance distribution of the feature,
- a probabilistic score for its discriminative power,
- and a spatial distribution of the object positions and scales.

Figure 4.9 shows examples from the output of our training process. Three example features were chosen for the car class. The first line shows sample patches for the features. On the second line we marked some feature-members on a chosen training image. In practice, as the example shows, it often happens that we have multiple detections of a given feature on an image. Arrows pointed to the center of object, indicate that we used these relations to learn a density function displayed in the last row. For the density estimate we use several car images with multiple detections of those features.

### Testing by Probabilistic Hough Voting

The localization procedure on a test image is similar to the initial hypothesis generation of Leibe and Schiele (2004). The difference here is that we incorporate the discriminative score into the voting scheme. First, we allow only the most discriminative clusters to participate in the decision of predicted object locations, and second, we integrate our probabilistic score in the voting scheme. These steps allow better confidence estimations for the different hypotheses. Our algorithm is the following. The extracted scale-invariant descriptors of the test image are assigned to the closest visual world (codebook entry) by appearance (MAP). Then, the chosen feature places its votes to possible object locations and scales (4D space). In practice we simplified the voting scheme from (Leibe and Schiele, 2004) by only allowing one feature per descriptor to vote, and extended their formulation by weighting each vote with the discriminative score obtained from  $\widehat{\mathcal{R}}^{(LIK)}$ . Furthermore, to eliminate votes of non-discriminative features, we limit the voting only for the ones that received the top

$n$  highest scores. The predicted object locations and scales are found as maxima in the 4D voting space using the Mean-Shift (Comaniciu and Meer, 1999) algorithm with a scale-adaptive balloon density estimator (Comaniciu *et al.*, 2001). The confidence level for each detection is determined by the peak value of the kernel density estimate.

## 4.2.2 Evaluation of Different Parameters

### Experimental Setup

In this section we evaluate the influence of different parameters of our system using a popular and publicly available *car* database, as well as the *bicycle* set from Graz1 that we have used in the previous chapters. For the cars, we train our system on scale-invariant features extracted from 50 images with hand-segmented cars (bounding boxes only). This training set has been introduced by Leibe and Schiele (2004). We run the localization process on the *UIUC test II* (Agarwal and Roth, 2002) dataset which consists of 108 images containing 139 cars of different sizes. Test images are of different resolutions often with highly textured background and include instances of partially occluded cars and cars with low contrast compared to their background. Notice, that our training and test sets are completely independent datasets, which allows us to even better evaluate the generalization capabilities for cars.

As for the bicycles, we use the same setup as earlier in Sections 4.1.2, 2.4, and 3.3.2. For the training and the evaluation we naturally use bounding-boxes instead of the pixel-wise segmentation from Section 3.3.2. The Graz1 bicycles dataset was originally collected for image classification by the authors, and therefore, several images are not very suitable to evaluate object localization (e.g. large number of multiple bicycle instances overlap in a parking lot). Even though, we have marked these images as good as possible, and have kept the same training and testing set as before, we expect reasonable, yet a bit lower performance on this set.

For a test image, the output of our method is a list of possible locations of the object class together with a confidence level, obtained as the value of the kernel density estimate. A location is given by a bounding box  $B = (c_i, c_j, w, h)$  with the position of the center, and the width and height of the object. To be considered a correct detection, the area of the overlap  $\iota$  between the predicted ( $B_p$ ) and the ground truth ( $B_g$ ) locations must exceed 50% specified as:

$$\iota = \frac{\text{area}(B_p \cap B_g)}{\text{area}(B_p \cup B_g)}$$

Furthermore, we only accept one correct detection per objects and count each additional predicted bounding boxes as false detections on the same object.

The false and correct detections are counted for each confidence level to draw the recall-precision curve, where

$$\text{Recall} = \frac{\# \text{ correct detections}}{\# \text{ objects}}; \quad \text{Precision} = \frac{\# \text{ correct detections}}{\# \text{ detections}}$$

There are several ways to compare two recall-precision curves. In this chapter we used the same as [Everingham et al. \(2005\)](#), the *average precision* (AP). It is used by TReC and is defined as the arithmetic mean of 11 *interpolated* precision  $\tilde{p}(r)$  values determined on thresholds of recall  $r \in \{0, 0.1, \dots, 0.9, 1\}$ . The *interpolated* precision  $\tilde{p}(r)$  is defined as the *maximum* precision for which the corresponding recall is greater than or equal to the threshold  $r$ . We used this measure in order to be comparable with the results of the PASCAL challenge in [Section 4.2.3](#).

### Performance of Different Feature Detectors

The following experiments use three of the most popular detectors and the new ones introduced in [Chapter 2](#). H-Lap has excellent repeatability in location (cf. [Section 2.3](#)) and its extracted regions are very rich in structures. On the other hand, the detected corner-like structures often lie on the boundary of the objects, and thus, the extracted features are less reliable for recognition of objects particularly with small sizes. Blobs extracted by LoG and DoG are well localized structures, but due to their homogeneity, the information content can be poor in the center of the region. To enrich this information, a common practice is to enlarge the neighborhood by a factor of 2 or 3, as we also did in our experiments for H-Lap, LoG, and DoG detectors. Due to the different nature of these detectors it is interesting to compare them in our object class localization approach. [Figure 4.10](#) shows the results of our system trained on different types of features using the setup described in above. For the cars LoG performs the best, followed by our new detectors. H-Lap and DoG come last. This can be explained by, first, on average a larger percentage of detected points ( $> 35\%$ ) lie on the cars, while in the case of the other two detectors this ratio is slightly lower ( $30\%$ ). Furthermore, LoG and L-MSLSD also detect larger number of points which could lead to better defined peaks in the voting space. Apart from the DoG and L-MSLSD detectors' improved performance, the detectors perform similarly on the bicycles. Due to the difficulty of this dataset the best result (L-MSLSD) is lower than the one on the car set. For both datasets, we also believe that the poor performance of some detectors can also be caused by the imprecise estimation of scales which is often unstable on e.g. corner-like structures like Harris points. In our scale-invariant approach the learned object properties (location of the center and the scale) are relative to the characteristic scale of detected points. As a consequence, the individual scales are essential parameters and the method can substantially suffer from their noisy or imprecise estimation. This is the reason why H-MSLSD outperforms H-Lap. The leading performances of LoG and L-MSLSD are partially due to the good scale estimation on blob-like structures.

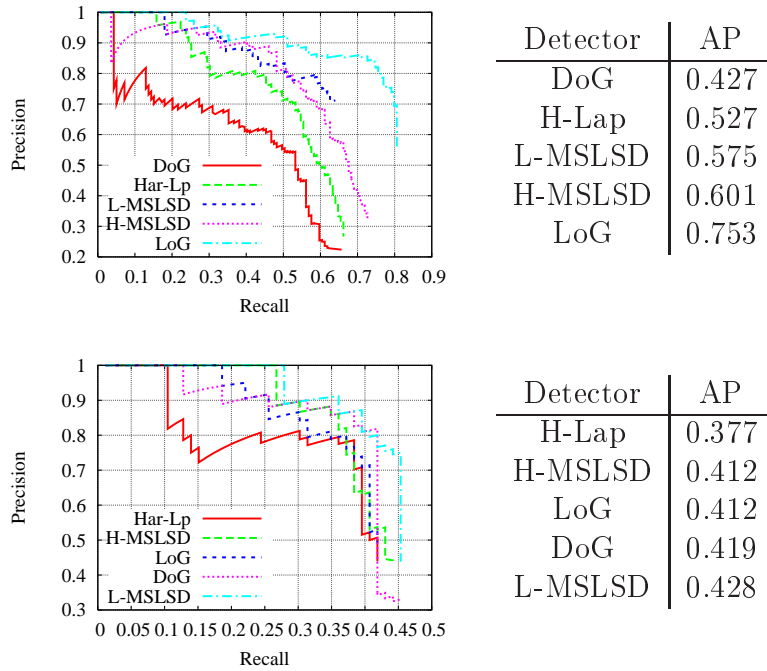


Figure 4.10: Recall-Precision curve of our localization approach on the UIUC test II (top) and the Graz1 bicycles (bottom) sets for different interest point detectors. The tables on the right show the average precision computed on the curves.

### Importance of Feature Selection

To measure the improvement and discuss the advantages of the integrated feature selection, we perform three different experiments with each of the interest point detectors.

- (A) We use all the 100 features in our vocabulary and for each, we learn the spatial distribution of the object positions and scales. We do not perform any feature selection. (The baseline method.)
- (B) Same as in (A), but we only use the top 25 features determined by  $\hat{\mathcal{R}}^{(LIK)}$  to vote. We do not integrate  $\hat{\mathcal{R}}^{(LIK)}$  scores in the voting.
- (C) Same as in (A), using all the 100 features, but we also compute the discriminative scores  $\hat{\mathcal{R}}^{(LIK)}$  for each feature, and use it to weight its votes in the mean-shift space.
- (D) Similar to (C), but we only use only the top 25 features to vote. (Combination of (B) and (C).)

	H-Lap	DoG	LoG	H-MSLSD	L-MSLSD
UIUC Cars II					
no f.sel., no weights (A)	0.414	0.162	0.389	0.452	0.402
best 25, no weights (B)	0.503	0.368	0.689	0.516	0.447
no f.sel., weights by $\hat{\mathcal{R}}^{(LIK)}$ (C)	0.441	0.383	0.512	0.498	0.406
best 25, weights by $\hat{\mathcal{R}}^{(LIK)}$ (D)	0.527	0.427	0.753	0.601	0.575
Graz1 Bicycles					
no f.sel., no weights (A)	0.417	0.423	0.432	0.432	0.434
best 25, no weights (B)	0.330	0.419	0.409	0.410	0.423
no f.sel., weights by $\hat{\mathcal{R}}^{(LIK)}$ (C)	0.405	0.419	0.432	0.430	0.431
best 25, weights by $\hat{\mathcal{R}}^{(LIK)}$ (D)	0.377	0.419	0.428	0.412	0.428

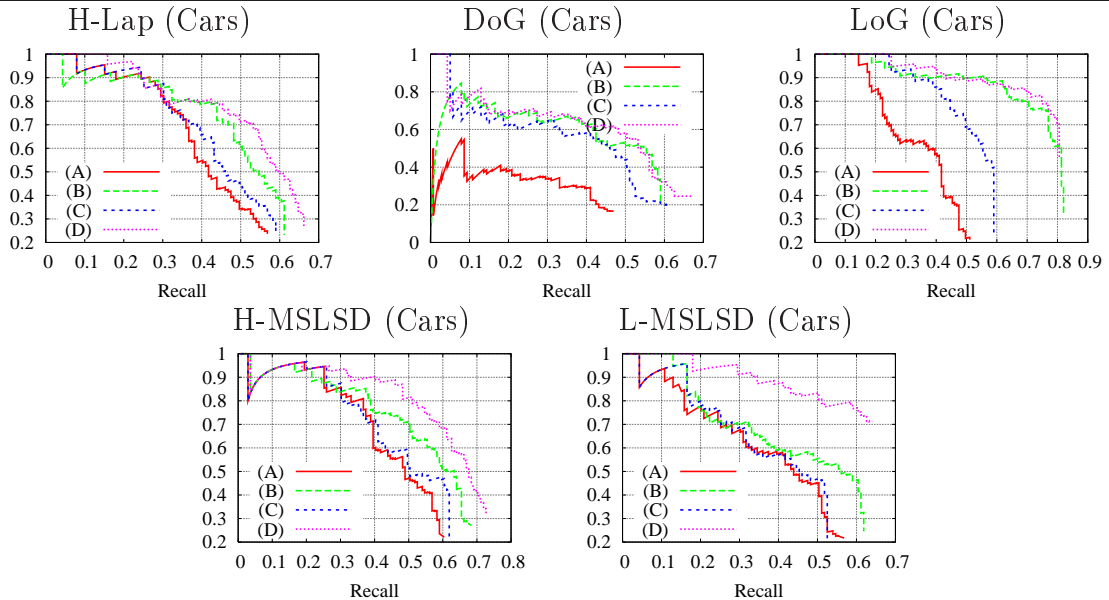


Figure 4.11: The effect of ranking and selection for different interest point detectors. For the cars each recall-precision curve shows the results of the object localization without feature selection (A), using only the top 25 clusters to vote (B), using the discriminative score for voting with all the features (C), and additionally to the weights select the top 25 clusters to vote (D). The table above shows the average performances for the cars and the bicycles datasets.

The results are summarized in Figure 4.11. For the cars, version (D), the feature selection together with the weighting, shows significant improvement for each detector. The sample detection in Figure 4.12 helps understanding why feature selection is so important for the voting phase. In general the best results can be achieved with the detector that delivers the *most* points on the objects. In our case the LoG detector selects the most points and delivers the best results. On the other hand, we



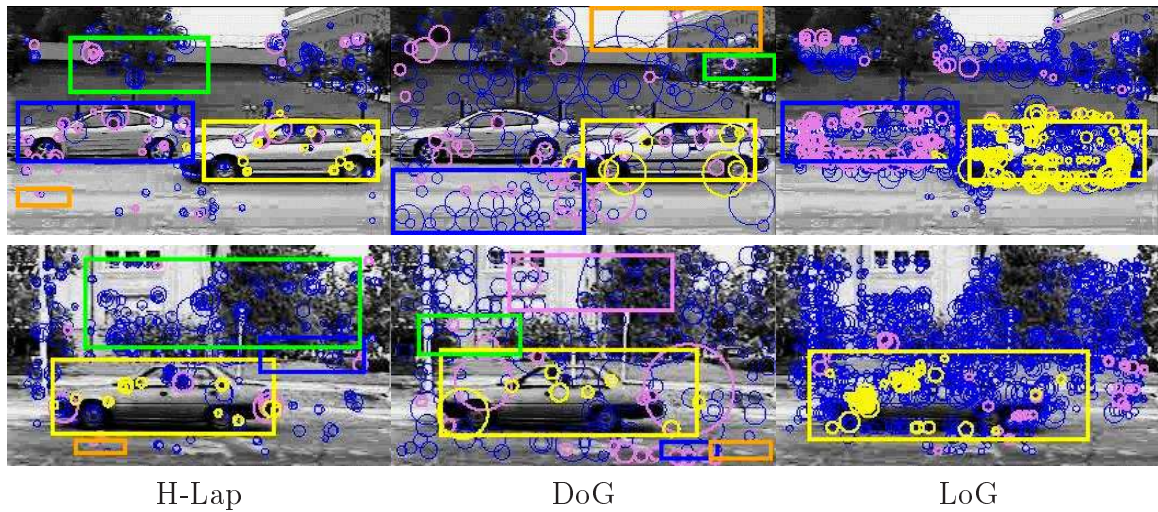


Figure 4.12: Example detection on two different test images (first and second row). Blue detections were eliminated by the top 25 discriminative cluster selection, their votes were not included in the mean-shift space. Yellow circles indicate points that actually participated in the selection of the most probable object location (yellow rectangle). Violet points voted for some other centers. Non-yellow bounding boxes indicate further possible solutions with lower confidence.

prefer few background points, because they add noise to the voting space. The best possible way to benefit from the increased detections on the objects is to reject the non-discriminative, and non-object features. As the examples show the pre-selection of the 25 most discriminative clusters reject a huge amount of points (indicated by blue circles). This clearly shows the advantage of discriminative feature selection. On the bicycles dataset, the feature selection does not show the same improvement. This is due to the nature of the database: several bicycles that are localized correctly are sideviews covering a huge part of the images, while others can never be found due to their previously unseen viewpoint or to their occlusion by other bikes. On this set feature selection and weighting cannot improve the results, moreover it sometimes caused weaker object estimates due to the fewer number of points.

### Comparison of Different Feature Selection Methods

As we did earlier for object feature retrieval (Section 3.3.2) and for image classification (Table 4.2) we compare the different selection techniques, here, for object localization. Unlike  $\mathcal{R}^{(LIK)}$ , many of the introduced method do not offer a straightforward conversion to probabilistic scores. Therefore, the following set of experiments do not take advantage of feature weighting as before; we only select the top 25 features and equally weight their contributions during the voting phase. Note that this is equivalent to experiments (B) from the previous section.



Ranking	Cars				
	H-Lap	DoG	LoG	H-MSLSD	L-MSLSD
$\mathcal{R}^{(LIK)}$	0.503	0.368	0.689	0.516	0.447
$\mathcal{R}^{(OR)}$	0.503	0.368	0.689	0.516	0.447
$\mathcal{R}^{(BNS+)}$	0.481	0.370	0.710	0.486	0.395
$\mathcal{R}^{(CMIM+)}$	0.432	0.390	0.765	0.529	0.388
$\mathcal{R}^{(MI+)}$	<b>0.504</b>	0.421	0.694	<b>0.539</b>	0.410
$\mathcal{R}^{(CC)}$	0.502	0.437	0.735	0.510	0.408
$\mathcal{R}^{(GSS)}$	0.499	0.420	0.726	0.528	0.403
$\mathcal{R}^{(F_1)}$	0.474	0.388	0.628	0.534	<b>0.482</b>
$\mathcal{R}^{(SVM+)}$	0.440	0.353	<b>0.773</b>	0.506	0.348
$\mathcal{R}^{(AB+)}$	0.397	<b>0.446</b>	0.691	0.531	0.275
$\mathcal{R}^{(Freq)}$	0.063	0.133	0.441	0.357	0.352

Ranking	Bicycles				
	H-Lap	DoG	LoG	H-MSLSD	L-MSLSD
$\mathcal{R}^{(LIK)}$	0.330	0.419	0.409	0.410	0.423
$\mathcal{R}^{(OR)}$	0.330	0.419	0.409	0.410	0.423
$\mathcal{R}^{(BNS+)}$	0.417	0.412	0.417	<b>0.428</b>	0.426
$\mathcal{R}^{(CMIM+)}$	0.415	0.417	0.418	<b>0.428</b>	<b>0.445</b>
$\mathcal{R}^{(MI+)}$	0.409	0.411	0.414	0.415	0.360
$\mathcal{R}^{(CC)}$	0.418	<b>0.430</b>	0.413	0.415	0.442
$\mathcal{R}^{(GSS)}$	0.412	0.412	<b>0.439</b>	0.415	0.437
$\mathcal{R}^{(F_1)}$	0.415	0.402	0.327	0.368	0.401
$\mathcal{R}^{(SVM+)}$	0.422	0.423	0.315	0.427	0.419
$\mathcal{R}^{(AB+)}$	<b>0.429</b>	0.369	0.302	0.401	0.409
$\mathcal{R}^{(Freq)}$	0.401	0.396	0.420	0.386	0.413

Table 4.4: Comparison of different ranking methods on the cars and the bicycles dataset. Reports are the Average Precision rates using the best (highest ranked) 25 components.

Table 4.4 details the average performance of each selection method for the same detectors as before. Even though, the best performances are several times achieved by  $\mathcal{R}^{(SVM+)}$  or  $\mathcal{R}^{(AB)}$ , on average  $\mathcal{R}^{(CC)}$ ,  $\mathcal{R}^{(GSS)}$ , and  $\mathcal{R}^{(MI+)}$  perform best (in this order). Notice that these three methods are non-conditional, i.e., select features independently of what has been selected before, and mix frequency with discriminative power. The benefit of discriminative features is that they help to remove noise from the voting space, while frequent parts, as they appear more often, have better estimation during the training, and therefore they lead to better object class models. Conditional

Class	Train		Test 1		Test 2	
motorbikes	214	217	216	220	202	227
bicycles	114	123	114	123	279	399
people	84	152	84	149	526	1038
cars	272	320	275	341	275	381

Table 4.5: The number of training and test images/objects in the PASCAL VOC2005 Challenge database.

selection methods may provide sufficient number of parts to well localize the objects (e.g. the LoG detector for cars), but on the other hand the rejected, and individually discriminative features could be missing, as they could still contribute to more precise location estimates (e.g. Harris based detectors for cars). Even though  $\mathcal{R}^{(CMIM)}$  is a conditional method, for the bicycle dataset it performed the best, and for the car dataset it performed better than  $\mathcal{R}^{(SVM+)}$  and  $\mathcal{R}^{(AB)}$ .  $\mathcal{R}^{(CMIM)}$  similarly to  $\mathcal{R}^{(CC)}$ ,  $\mathcal{R}^{(GSS)}$ , and  $\mathcal{R}^{(MI+)}$ , explicitly takes the feature frequency into account.

From all these we conclude that for object localization based on part distribution estimates, discriminative feature selection may improve the results, and techniques based on individual filtering that take into account both discriminative power and frequency are the most suitable.

### 4.2.3 Additional Results: PASCAL Challenge, Butterfiles

In this section we present results for the PASCAL Visual Object Classes Challenge (VOC2005)<sup>1</sup> dataset and the butterfly dataset. Results on the PASCAL Challenge allow to compare our method to the state-of-the-art. The butterfly dataset validates our approach to rotation invariance.

In the PASCAL Challenge dataset there are four different object categories: motorbikes, bicycles, people and cars. Table 4.5 shows the the number of images and objects per category. We train our detector for each class with the given training set; we used 1200 clusters and descriptors extracted by LoG (the detector performed best on cars in our previous localization experiments). For localization we run the 4 object class detectors with 100 selected clusters, with weights computed by  $\hat{\mathcal{R}}^{(LIK)}$ , separately on the test1 and the test2 sets. Note, that each detector is tested on *all* test images i.e., 689 images for test1 and 1282 for test2. The test1 set is taken from the same distribution of images as the training data, i.e., same type of scene and conditions, while the test2 set provides a “more difficult” set of specifically collected images for the challenge.

<sup>1</sup>Dataset, description and report of other methods on the same set are available at <http://www.pascal-network.org/challenges/VOC/>.



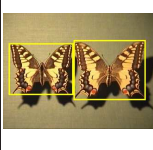


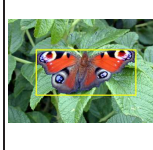
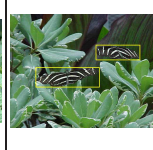



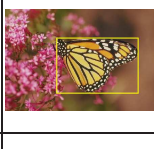
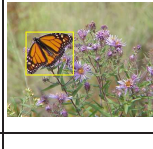
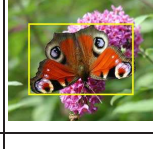
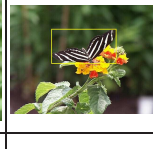
	Admiral	Black Swallow-tail	Machaon	Monarch closed	Monarch open	Peacock	Zebra
Sample images							
							
#img	26/85	26/16	26/57	26/58	26/48	26/108	26/65

Figure 4.13: The butterfly database (Lazebnik *et al.*, 2004). Example images from each of the seven categories. Bouding boxes indicate the hand-segmented groundtruth. Last line shows the number of images in the training/test sets.

The *butterflies* (Lazebnik *et al.*, 2004) database consist of seven different categories of butterflies. For the training and the localization we use our extension to rotation invariance (Section 4.2.1), because both on the training and on the testing images the butterflies are in different pose and direction. The seven categories with examples can be found in Figure 4.13. The *butterflies* database has the following challenging properties:

- rotation invariance (in reality there are also 3-D viewpoint changes),
- similarities between different classes of butterflies,
- less rigidity, due to their wings,
- *monarch open* and *monarch closed* are two different categories with the same type of butterflies.

Please note, that we have only used grey-scaled images with SIFT descriptors in the following experiments, to keep our local representation consistent throughout the thesis, we believe that adding color information may significantly improve the results. Similarly to the PASCAL Challenge, we used the LoG detector, and weight our votes by  $\hat{\mathcal{R}}^{(LIK)}$  scores. For the butterflies we select 50 features out of 300. We split the multi-class problem into seven two-class localizers, and we allow multiple labels on the test images. Even though we do not have images in our set in which two or more different types of butterflies occur, our system is built to be able to localize or all of them. In each experiment we use the same test images. Detected instances of other butterflies and multiple detections of the same butterfly are counted as false positives.

TEST 1					
	Ours	Darmstadt	Fr.Telecom	Inria-Dalal	Edinburgh
motorbikes	0.824	0.886	0.729	0.490	0.470
bicycles	0.355	-	-	-	0.119
people	0.103	-	-	0.013	0.002
cars	0.456	0.489	0.353	0.613	0.000

TEST 2					
	Ours	Darmstadt	Fr.Telecom	Inria-Dalal	Edinburgh
motorbikes	0.245	0.341	0.289	0.124	0.116
bicycles	0.209	-	-	-	0.113
people	0.021	-	-	0.021	0.000
cars	0.110	0.181	0.106	0.304	0.028

Table 4.6: Average precision rates on the four different categories of the PASCAL VOC2005 challenge dataset. Our performance is compared to the four competing institutions’ best results. Empty cells indicate that the competitor did not run their method(s) on given test set.

## Results

Table 4.6 compares our performance with the best results of the challenge. Most of the competitors submitted several results of different methods. Here, we always take their best results for comparison, for more detail which method they used we refer to the book chapter dedicated to the challenge (Everingham *et al.*, 2006). Figure 4.14 shows our recall-precision curves on the different categories. Example detections are shown in Figure 4.15.

In categories of bicycles and people (test1) our method outperformed all existing results while in the other cases it showed comparable performance. The method of Darmstadt is based on (Leibe and Schiele, 2004) and is only slightly better than ours’. Their algorithm includes two additional verifications steps. One, based on the figure-ground segmentation requiring additional segmentation masks, while the other one is an SVM-based step to reject false detections. It is remarkable that the simple voting algorithm used together with our feature selection can compete with theirs. An additional advantage of the proposed solution is the gain in execution time due to the elimination of unnecessary votes by feature selection.

Figure 4.16 shows the average precision rates on the different butterfly categories, using the same evaluation criteria. Our rotation invariant localization correctly retrieve (recall) around 70 – 80% of the butterflies. As we may have expected, the main difficulty is to separate between different categories. This is the main reason of the poor precision for the *Admirals* and the *Black Swallowtails*. Our *Machaon* detector

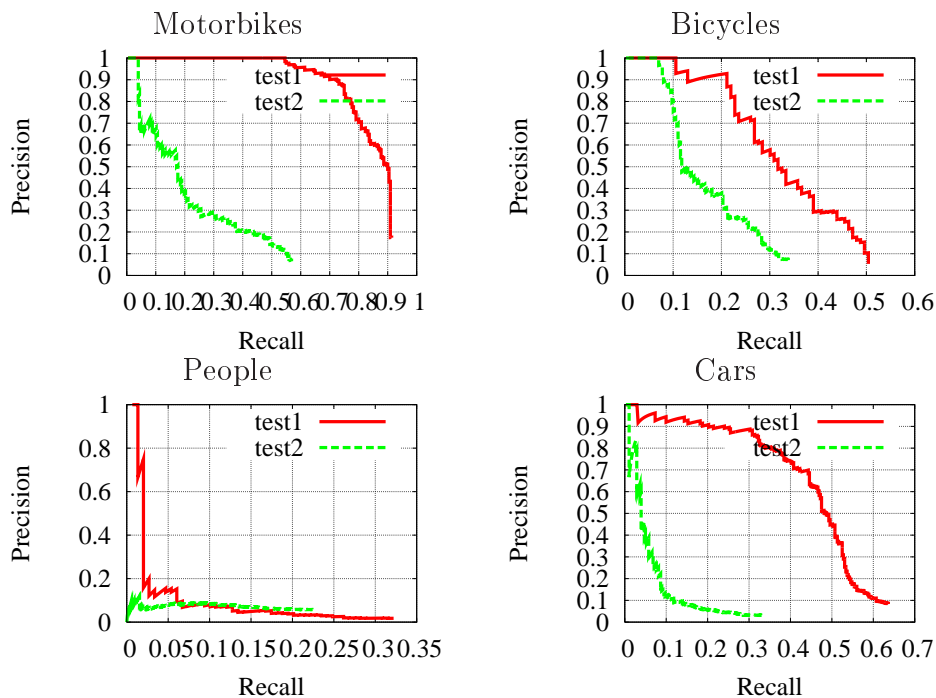


Figure 4.14: Recall-Precision curves on the PASCAL VOC2005 Challenge dataset. Each plot shows a different category. For each category two curves are showed for the two different test sets.

also falsely detects many other butterflies, but due to the *Machaons*' particular and discriminative patterns in their middle stripe our system assigns a much higher confidence for correct instances. *Monarch closed* and *open* are two different categories, and unfortunately a substantial performance drop is due to mixing them up. Even though *Peacocks* have a particular pattern at the end of their wings using the LoG detector the rest of the butterflies (middle wings and body) remain almost featureless. While those few number of parts—not so discriminative without color—provide some correct detections, the lack of features leads to very few, weak (in confidence) localizations. The *Zebra* butterflies have a lots of detection along their stripes leading to outstanding precision on all recall rates.

### 4.3 Implementation Details

In this section we give more insight and detail about the implementation and the parameters used over our experiments. In Section 2.5 we have already discussed the details of the interest point detectors used in Chapter 2. These are the same here as well. For the other detectors, ENTR, DoG, IBR, and MSER, we use the publicly available binaries from the authors with their default parameters.

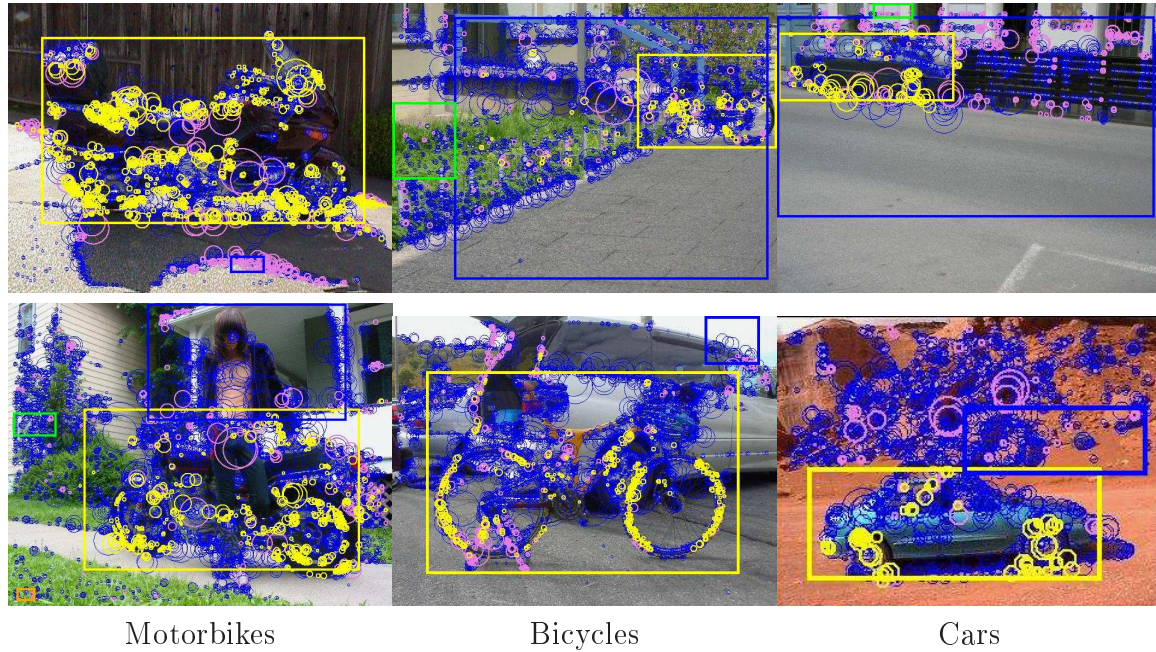


Figure 4.15: Example detections for the PASCAL Challenge (2005) dataset. First row shows images from the test1 while the second row from the test2 set. Blue points are eliminated due to feature selection, and yellow points are voted for the best solution (yellow rectangle). Non-yellow rectangles indicate false detections with lower confidence.

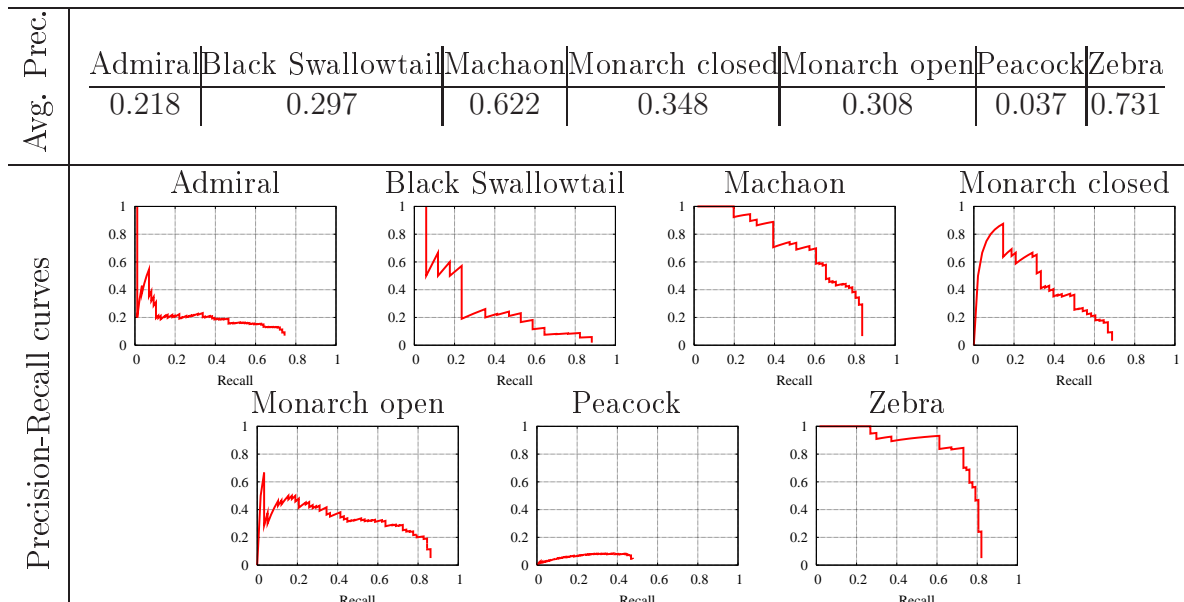


Figure 4.16: Results on the butterfly database.

### Object Class Classification Experiments

After extracting interest points and computing the local SIFT descriptors, we proceed with EM clustering to estimate a GMM for each class. Notice, that all experiments are two-class problems, an object category vs. background. We create a vocabulary for each class independently. EM algorithms are initialized with kmeans, where the number of clusters, and therefore the number of modes of the GMM is 400 for motorbikes and airplanes, 200 for bicycles and faces, 50 for wild cats, and 25 for leaves. These numbers are chosen manually, according to the size of the database. The EM-loop quickly converge within the first 5 – 20 iterations. The clustering is done on one half of the training set. The other half is used to compute feature ranks for each center (Gaussian mode). While for the ranking we use soft feature assignment, i.e., the actual probabilities from the GMM, at testing time we hard assign each descriptor to the most probable cluster. The features (clusters) are ordered based on their ranking, and the top  $n$  are marked as object features.  $n$  is a parameter of the system and the variable in our ROC curves. When all descriptors of a test image are assigned to their clusters we count how many of them fall into an object cluster. If this number is greater than  $p$ , we classify the image as positive. The  $p$  parameter is learned during training by optimizing the classifier to the highest EER on the ranking set.

For the bag-of-keypoint experiments we use similar setting as described Section 2.5. The number of clusters is the same as for the previous classifier.

### Object Class Localization Experiments

In general, the first steps of these experiments are identical to previous ones. We extract interest points, compute SIFT descriptors and cluster the descriptors to create the visual vocabulary. The number of clusters is 100 for cars, 1200 for the PASCAL challenge, and 300 for the butterflies. For the PASCAL challenge and butterflies databases we use one-one common vocabulary for all categories. Each feature (cluster) is ranked by the likelihood ratio criterion. Unless it is stated otherwise, the best  $n$  features are selected, and the distribution of the object positions relative to these features are learned. We use non-parametric distributions, and basically store all occurrences of object centers and scales (width + height) for each feature. Note that both location and object scale values are normalized by the scale of the descriptor. At detection time, each local descriptor is hard-assigned to its closest cluster, which places its votes to the 4D voting space. All votes are weighted by the discriminative power of its cluster, i.e., the sum of the placed votes per descriptor equals to the  $\hat{\mathcal{R}}^{(LIK)}$  score of its assigned cluster. The final predicted object locations and scales are found as maxima in the 4D voting space using the scale-adaptive Mean-Shift (Comaniciu and Meer, 1999; Comaniciu *et al.*, 2001) algorithm. The confidence level for each detection is determined by the peak value of the kernel density estimate.



## 4.4 Discussion

In this chapter, we have shown two different tasks which use discriminative feature selection. Our image classification is based purely on the number of selected features for a given object class. For this task, we have found  $\mathcal{R}^{(LIK)}$  (and  $\mathcal{R}^{(OR)}$ ) the best suited selection methods as they retrieve the most *trusted* features. Note that this agrees with our earlier results in Chapter 3 where these methods provide the best object coverage. However, the  $\mathcal{R}^{(LIK)}$  and  $\mathcal{R}^{(OR)}$  are followed by  $\mathcal{R}^{(SVM+)}$ ,  $\mathcal{R}^{(AB)}$ , and  $\mathcal{R}^{(CMIM)}$ , providing low object coverage (cf. Section 3.3.2), yet good discrimination. These experiments confirm our earlier observation, that these three methods provide fewer features on the objects, because they reject redundant discriminative components.

In the object localization experiments each feature has a (non-parametric) probability estimate for the relative object position. Frequent features have more stable estimates, due to more training examples, and therefore on average the leading feature selection methods are  $\mathcal{R}^{(CC)}$ ,  $\mathcal{R}^{(GSS)}$ , and  $\mathcal{R}^{(MI+)}$ . The other selection methods may also provide good performance—as they do in our experiments—when the selected features (accidentally) have enough statistics. This explains that  $\mathcal{R}^{(SVM)}$  and  $\mathcal{R}^{(AB)}$  several times are the leading methods, but on average they are below the trio that takes the frequency explicitly into account. For the same reason among the *conditional* methods  $\mathcal{R}^{(CMIM)}$  performs the best.

Image classification experiments has shown that by using only feature selection our system provide competitive results on popular datasets. As for the localization we have shown how to improve both speed and accuracy of the voting phase of [Leibe and Schiele \(2004\)](#) by the integration of discriminative feature selection. We have also generalized their method to general similarity transformations by adding rotation invariance without requiring any pre-normalization (pre-rotation) of the training images.





---

## Conclusion and Future Work

---

RECOGNITION of object categories is a challenging task. Learning algorithms have to generalize over all specific instances of an object class, and at the same time they have to learn enough distinctive information to separate the objects from the background. A system which solves such a challenging goal has to rely on a high quality image representation as well as appropriate learning methods. This thesis has addressed these key features by proposing a novel scale-invariant keypoint detection method, and by investigating class-discriminative feature selection.

We have introduced a new technique, called the Maximally Stable Local Description, to provide more stable local descriptors, and consequently a better appearance based representation for images. We have applied MSLD for scale selection on keypoints extracted on multiple scales by the Harris and the Laplacian operators. The algorithm uses *description stability* as criterion for scale selection: the characteristic scale for each location is chosen such that the corresponding representation (in our experiments SIFT) changes the least with respect to scale. The informative content and *repeatability* of the detections are guaranteed by the keypoint detectors, while *description stability* is preserved by MSLD scale selection. This balanced solution has demonstrated competitive results, many times outperforming the Laplacian selection. We have particularly found MSLD scale selection beneficial in the following situations.

While the new detectors may have weaker repeatability rates in standard image matching environments, they can provide additional robustness, invariance, and therefore improved performance in challenging conditions. For example, due to the inherent property of SIFT, i.e., being invariant to affine light changes, we have demonstrated improved performance for image matching under different lighting conditions.

The stability of the bag-of-keypoints representation rely on the stability of the local descriptors. Since our method enforces this stability it has consistently demonstrated better results on textures and materials, as well as several times improvements on object categories.

Recent works on object recognition mostly reported a decrease in performance when imposing additional levels of invariance, such as rotation. The standard explanation is

that more invariance makes descriptors more similar, and therefore they lose distinctiveness. On texture databases we have shown in several experiments that their poor performance is mainly due to the instability in the parameter estimation, and that our new criterion, which maximizes stability, can overcome those challenges. Adding rotation invariance has consistently improved our results using our new detectors.

Experiments on object localization matches the features by appearance to a code-book entry, and it explicitly uses the scale estimation (the region shape) to normalize all distances to learn the spatial configuration of the object class. Therefore, both repeatability (location + scale) as well as description stability is crucial for this task. Our detectors have shown competitive results: while the 3D Laplacian detector (LoG) performs better than our scale estimation on 2D Laplacian points, MSLSD (Maximally Stable Local SIFT Description) on Harris corners is consistently better than Harris-Laplace. The reason is two-fold. First, probably for both cases the appearance matches are improved with MSLSD, and second, for Harris points, the scale estimates are less stable, i.e., less repeatable, using Laplacian.

In this thesis, we have also adopted several feature filtering techniques from the text literature. We have shown how to use them for class-discriminative feature selection and ranking. Several properties have been analyzed and explained. One major difference is whether the selection of positive and negative features are treated equally or not. Consequently, we have shown how to convert one-sided measures to two-sided, and vice versa. Some selection methods (e.g., mutual information, chi-square) explicitly take into account feature frequency, while others (e.g., likelihood, odds ratio) are based only on discriminative power. We have also listed three different methods, SVM coefficient, AdaBoost, and conditional mutual information maximization which reject redundant features. On practical terms, where visual features are quantized distribution of sparsely extracted local descriptors, we have observed that all these methods select class-discriminative locally consistent object-parts (e.g., tires of cars, eyes of faces, etc.) and dominant textures (e.g., pattern of wild cats). We have evaluated the selection methods in three different scenarios, and have come up with the following recommendations (see Figure 5.1).

The purely appearance based object coverage problem tries to retrieve as many features on objects as possible while minimizing the number of background features. This is a typical scenario when only the discriminative power of the features are important, and even special features, i.e., features that corresponds to some special usually rare object structures, are very valuable besides the frequent ones. Our experimental results confirming the use of classification likelihood and odds ratios for such tasks.

Our image classification scenario has used purely an appearance based representation to decide about the presence of an object instance in an image. In our simple classifier—where we have required a predefined number of object features for the presence of an object—likelihood and odds ratio have shown the best performance. The runners-up are SVM coefficients, AdaBoost, and conditional mutual information, which is the group of methods that reject redundant features. Our experiments have

Scenario	Aim	Relevant properties	Recommendations
object coverage	retrieve as many object features as possible (Chapter 3) :	discriminative power; include redundant features	likelihood ratio, odds ratio
image classification	presence/absence test of object instances; appearance-based (Chapter 4)	discriminative power; redundant features less important	likelihood ratio, odds ratio, SVM coef., Adaboost, conditional mutual information maximization
object class localization	determine the position & scale of object instances in a scene; appearance + spatial relations (Chapter 4)	discriminative power; frequency to support statistics of spatial distributions	chi-square, correlation coef., GSS, mutual information; conditional mutual information maximization (if sparser representation required)

Figure 5.1: Recommendations for feature selection in three different scenarios. For each scenario we list the main properties of the features (third column) and the selection methods that performed the best, and therefore are recommended (last column).

also shown that a more sophisticated classifier, a linear SVM which includes implicit selection based on SVM coefficients, outperforms our simple classifier on average, but at the same time has also confirmed the success of the selection method by SVM coefficients.

The demonstrated object class localization method estimates the spatial distribution of different features, and therefore sufficient statistics also play an important role. Even though we have used non-parametric estimates, our experimental results have indicated that good features must be supported by sufficient training examples. Consequently, the most appropriate selection methods are chi-square (and its derivatives, correlation coefficient and GSS) and mutual information. If a sparser representation is preferred, with the price of loss in accuracy, redundant features can be most efficiently rejected by conditional mutual information maximization, since it also ensures the sufficient frequency of the selected features. In general, we have shown that class-discriminative feature selection plays an important role in object localization. The combination of existing methods have led to a simple framework which outperforms or obtain comparable results to state-of-the-art methods. Our integration of feature selection in the voting framework provides the following advantages:

- While keeping the background features nearly constant, the number of object descriptors can be increased, e.g., by adding more interest point detectors (cues) or lowering the thresholds of existing ones. More object features usually improves the performance of localization.

- The spatial (foreground) model is learned on object features, i.e., is not built on features that may often appear on the background, providing a better model for the object class.
- The speed of the final detection on new images is significantly improved due to the removed non-discriminative object and background features.

## Future Work

The presented work has many possible extensions. In the following we summarize our ideas for such future work.

### Extensions for MSLD

In the following we describe three possible extensions for our MSLD criterion. The first is to initialize the scale selection by different types of regions, the second is to embed different types of descriptors, and finally an extension to develop scale-invariant dense representations.

Scale selection via Maximally Stable Local Description can be applied on different types of regions, such as Hessian points or extremal regions, etc. In most cases this is straightforward, e.g., for Hessian points it can be done in the same way as we have shown for the Harris corners; for extremal regions the MSER detector shape stability (region area or boundary length) can be replaced by descriptor stability. Various initial conditions, e.g., interest point detectors, constrain the search space differently. It would be interesting to analyze what types of final regions are selected by the MSLD scale selection in these different environments.

While the SIFT descriptor is one of the most successful general descriptors, using other image features can be beneficial: scale selection, and consequently interest point detection, can be improved by adding different levels of invariance. SIFT provides invariance for changes in light conditions; other descriptors may provide other invariance, e.g., to certain color changes, or to specific types of noise. Embedding these descriptors in the detector, MSLD can provide more robust detections in those specific conditions. Moreover, invariance is obtained locally on several parts of the image, and therefore more powerful than a global preprocessing of the image.

Scale selection on each pixel provides a scale-invariant dense representation. Using MSLD for scale estimation may provide a particularly stable local description, and therefore could be very useful for representing textures and patterns.

### Future Prospects of Discriminative Feature Selection

In the following we describe unsolved problems and possible extensions for discriminative feature selection. First we mention the problem of choosing the number of

---

features. We then discuss an extension to a dense feature space. After that, we also point out the benefits of integrating codebook creation and feature selection. Discussion on the types of selected features motivates us to investigate more in textured objects. Finally, we present the possibility to generalize our appearance based applications towards object structures and other types of features.

Typically feature selection has one parameter (we have called it  $n$ ), the number of components. Many times the complexity of the learning model limits this parameter, and sometimes it can be set intuitively (e.g., likelihood ratios are meaningful values). It can be a main parameter of the task (such as in image classification, the ROC curves has been plotted respectively to  $n$ ), but many times it has to be appropriately chosen (e.g., for localization). If no intuition and no previous experience is available it might be set by cross-correlation, however, it would be interesting to investigate more sophisticated ways.

In this thesis we have focused on sparse image representations. Keypoint detectors reduce the complexity, i.e., the number of local features to deal with, by *smart* sampling of the space. Reduced memory consumption and systematically sampled data allow to deal with more training examples, and therefore lead to better statistics and performance. Feature selection as an immediate first step could efficiently sample the descriptor space as well. This requires to run feature selection on the quantized space of dense descriptors (descriptors extracted at every pixel and at every scale). Existing quantization methods, and the selection techniques discussed in this thesis allow this computation with linear complexity on the descriptor space. In practice, this is possible for dense multi-scale features. The resulting selection of descriptors would be a sparse set of mostly object features, where the sparsity is set by the number of selected features. Since the distribution of the discriminative points are very different from the existing keypoint detectors, such a new representation should be thoroughly tested and the learning methods should be adapted.

This thesis has demonstrated feature selection on (pre)extracted visual words. Coupling the two steps, codebook creation and feature selection, could be beneficial for the following reasons.

- From our experiments we have learned that various tasks have different needs. Many *special* discriminative features support the best object coverage, while frequent features are necessary for distribution estimates. We can select the required features with the appropriate techniques as shown in this thesis. However, the performance could be improved by constraining the feature creation (codebook construction) to satisfy these additional requirements. These requirements can be derived from the feature selection.
- Creating codebook entries that can later be used to better separate object and background features may lead to better foreground appearance model, and therefore, improved performance. Feature selection can guide codebook creation to obtain more discriminative features. In practice this leads to (semi-)supervised

clustering, because the available class labels are used by discriminative feature selection.

Our experiments have shown that feature selection methods select discriminative object parts as well as dominant texture features. However, we believe that recognition methods, like the one we have used for localization, can benefit from these two types of features differently. On rigid objects, the part-type features have more precise relative spatial distribution to the object center, while texture-type features should be grouped together for efficient spatial estimates. An interesting extension would be to learn to distinguish between these two types of features.

This thesis has applied class-discriminative feature selection to codebooks of appearance. This, of course, can be extended to select object structures, i.e., spatially constrained tuples of object parts. Creating features that encode small or large parts, loose or strong relationship of appearance based features may allow to determine object class specific rigid and less rigid geometrical structures.

---

---

# Appendix A

## Influence of the number of interest points

---

Our method for image classification relies on the parameter  $p$ , the threshold on the number of positively classified interest points. To evaluate the bias of our approach, we examine the influence of the number of points on image classification. Note that a bias exists for almost any classification method, i.e. a low information content or low image resolution of the negative images can influence their classification results. The following study therefore also evaluates the *difficulty* of the databases.

In the following we evaluate the performance of a classifier based on the number of interest points. An image is classified as positive, i.e. containing the object category, if the number of detections are higher than a certain threshold  $t$ . Changing this parameter  $t$  determines an ROC curve, on which we report the equal-error-rates in Table A.1. The experimental set-up is the same as in Section 4.1.2.

Table A.1 shows the results for HL and ENTR detectors as well as for the combination HL + ENTR. In each case the first column gives the average number of interest points on the foreground and background images. The second column shows the equal-error-rate. Results with HL are very good for airplanes, faces, motorbikes and wild cats. For these categories significantly more points are detected on the object images than on the background ones; on average for these databases the EER of our approach only slightly increases. Results are at chance-level for leaves, bicycles, and people; for them our approach increases the EER significantly. Results with the entropy detector lead to similar conclusions. Note the classification performance for people are below chance-level, as more points are detected on the background. The results for the combination are again very good for the Caltech dataset, and the results for the Graz database are at chance-level. This shows that the Caltech dataset is biased, whereas the Graz dataset is not. Background images of the Graz dataset are of the same size as object images and contain a significant amount of clutter, whereas Caltech background images have lower resolution and contain less clutter. In conclusion, when images containing the object class are consistently more informative than the background, the extracted number of interest points can help image classification to find



Table A.1: Equal-Error-Rate for image classification based on the number of interest points.

Database	HL		ENTR		HL+ENTR	
	Avg. # IP fg./bg.	EER %	Avg. # IP fg./bg.	EER %	Avg. # IP fg./bg.	EER %
CalTech Databases						
Airplanes	119/25	88.2	90/54	70.4	209/79	78.7
Faces	311/25	98.9	115/54	76.2	426/79	92.9
Motorbikes	199/25	95.8	207/54	89.8	406/79	95.3
Wild Cats <sup>1</sup>	125/25	90.9	164/54	80.7	290/79	86.7
Leaves	23/25	53.6	96/54	73.1	119/79	89.1
TU-Graz1 Databases						
Bicycles	243/219	52.0	254/138	84.0	498/357	66.0
People	219/241	56.0	137/201	30.0	357/441	44.0

the right category. As a consequence, “badly” constructed datasets can bias the results which can influence any image based classification method. Note that our method is independent of the bias, as it allow to obtain excellent results on the unbiased Graz database.

---

---

# List of Publications

## Gyuri Dorkó

---

### CONFERENCE AND WORKSHOP PAPERS

Caroline Pantofaru, **Gyuri Dorkó**, Cordelia Schmid, and Martial Hebert. *Combining regions and patches for object class localization*. In Beyond Patches, Workshop in conjunction with CVPR, New York, 2006.

**Gyuri Dorkó** and Cordelia Schmid. *Maximally stable local description for scale selection*. In Proceedings of the 9th European Conference on Computer Vision, Graz, Austria, 2006.

Diane Larlus, **Gyuri Dorkó**, and Frederic Jurie. *Création de vocabulaires visuels efficaces pour la catégorisation d'images*. In Reconnaissance des Formes et Intelligence Artificielle, 2006.

**Gyuri Dorkó** and Cordelia Schmid. *Selection of scale-invariant parts for object class recognition*. In International Conference on Computer Vision, volume 1, pages 634–640, 2003.

**Gyuri Dorkó** and Cordelia Schmid. *Feature selection for object class detection*. In Workshop on Learning, Snowbird, Utah, 2003.

Barbara Caputo and **Gyuri Dorkó**. *How to combine color and shape information for 3d object recognition: kernels do the trick*. In S. Becker, S. Thrun, and K. Obermayer, editors, Advances in Neural Information Processing Systems, 2002.

Barbara Caputo, **Gyuri Dorkó**, and Heinrich Niemann. *An ultrametric approach to object recognition*. In G. Greiner, H. Niemann, T. Ertl, B. Girod, and H.-P. Seidel, editors, Vision, Modeling, and Visualization, 2002.

Barbara Caputo, **Gyuri Dorkó**, and Heinrich Niemann. *Combining color and shape information for appearance-based recognition using kernel gibbs distributions*. In In-

ternational Conference on Pattern Recognition, Support Vector Machines Workshop, 2002.

**Gyuri Dorkó**, Dietrich Paulus, and Ulrike Ahlrichs. *Color segmentation for scene exploration*. In Workshop Farbbildverarbeitung, 2000.

#### JOURNAL PAPERS

**Gyuri Dorkó** and C. Schmid. *Object class recognition using discriminative local features*. Computer Vision and Image Understanding, Submitted, 2006.

Peter Carbonetto, **Gyuri Dorkó**, Cordelia Schmid, Hendrik Kück, and Nando de Freitas. *Learning to recognize objects with little supervision*. International Journal of Computer Vision. Submitted, 2005.

#### BOOK CHAPTERS

**Gyuri Dorkó** and C. Schmid. *The 2005 pascal visual object classes challenge - section inria-dorko*. In F. d'Alche Buc, I. Dagan, and J. Quinero, editors, Selected Proceedings of the first PASCAL Challenges Workshop. LNAI, Springer, 2006.

Frederic Jurie, **Gyuri Dorkó**, Diane Larlus, and Bill Triggs. *The 2005 pascal visual object classes challenge - section inria-jurie*. In F. d'Alche Buc, I. Dagan, and J. Quinero, editors, Selected Proceedings of the first PASCAL Challenges Workshop. LNAI, Springer, 2006.

Cordelia Schmid, **Gyuri Dorkó**, Svetlana Lazebnik, Krystian Mikolajczyk, and Jean Ponce. *Pattern recognition with local invariant features*. In C.H. Chen and P.S.P Wang, editors, Handbook of Pattern Recognition and Computer Vision. World Scientific, third edition, 2005.

#### OTHERS

**Gyuri Dorkó** and Cordelia Schmid. *Object class recognition using discriminative local features*. Rapport de recherche RR-5497, INRIA - Rhone-Alpes, February 2005.

**Gyuri Dorkó** and Zoltán Ladányi. *Signature verification on the basis of dynamic attributes*, 1999. Presented at the National Conference of Student's Scholarly Circles (in Hungarian language).

---

---

# Bibliography

---

- 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. [51](#), [81](#)
- S. Agarwal and D. Roth. Learning a sparse representation for object detection. In *Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark*, volume IV, pages 113–127, 2002. [80](#), [91](#), [95](#)
- J. Babaud, A. P. Witkin, M. Baudin, and R. O. Duda. Uniqueness of the gaussian kernel for scale-space filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):26–33, 1986. [26](#)
- A. Baumberg. Reliable feature matching across widely separated views. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, volume I, pages 774–781, 2000. [32](#)
- S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002. [29](#)
- C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995. [66](#)
- D. Blostein and N. Ahuja. A multi-scale region detector. *Computer Vision, Graphics and Image Processing*, 45(1):22–41, January 1989. [30](#)
- G. Bouchard and B. Triggs. Hierarchical part-based visual object categorization. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Diego, California, USA*, volume 1, pages 710–715, June 2005. [80](#)

- J. Brank, M. Grobelnik, N. Milič-Frayling, and D. D. Mladenič. Feature selection using linear support vector machines. Technical report msr-tr-2002-63, Microsoft Research, Redmond, WA, USA, 2002. 62
- P. Brodatz. *Textures: A Photographic Album for Artists and Designers*. Dover Publications, New York, 1966. 43
- M. F. Caropreso, S. M., and F. Sebastiani. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In A. G. Chin, editor, *Text Databases and Document Management: Theory and Practice*, pages 78–102. Idea Group Publishing, Hershey, US, 2001. 55
- X. Chen, L. Gu, S. Li, and H.-J. Zhang. Learning representative local features for face detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, USA*, volume I, pages 1126–1131, 2001. 51
- D. Comaniciu and P. Meer. Distribution free decomposition of multivariate data. *Pattern Analysis and Applications*, 2:22–30, 1999. 95, 106
- D. Comaniciu, V. Ramesh, and P. Meer. The variable bandwidth mean shift and data-driven scale selection. In *Proceedings of the 8th International Conference on Computer Vision, Vancouver, Canada*, volume 1, pages 438–445, July 2001. 95, 106
- J. Cottier. Extraction et appariements robustes des points d’intérêt de deux images non étalonnées, September 1994. 25
- D. Crandall, P. F. Felzenszwalb, and D. P. Huttenlocher. Spatial priors for part-based recognition using statistical models. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Diego, California, USA*, volume 1, June 2005. 80
- G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. In *Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004. 41
- N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Diego, California, USA*, pages 886–893, 2005. 80
- G. Dorkó and C. Schmid. Selection of scale-invariant parts for object class recognition. In *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, volume 1, pages 634–640, 2003. 56
- Y. Dufournaud, C. Schmid, and R. Horaud. Matching images with different resolutions. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, pages 612–618, June 2000. 26

- M. Everingham, L. van Gool, C. Williams, and A. Zisserman. Pascal visual object classes challenge results. <http://www.pascal-network.org/challenges/VOC/voc/>, 2005. 96
- M. Everingham, A. Zisserman, C. K. I. Williams, L. van Gool, M. Allan, C. M. Bishop, O. Chapelle, N. Dalal, T. Deselaers, G. Dorkó, S. Duffner, J. Eichhorn, J. D. R. Farquhar, M. Fritz, C. Garcia, T. Griffiths, F. Jurie, T. Keysers, M. Koskela, J. Laaksonen, D. Larlus, B. Leibe, H. Meng, H. Ney, B. Schiele, C. Schmid, E. Seemann, J. Shawe-Taylor, A. Storkey, S. Szedmak, B. Triggs, I. Ulusoy, V. Viitaniemi, and J. Zhang. The 2005 pascal visual object classes challenge. In F. d’Alche Buc, I. Dagan, and J. Quinonero, editors, *Selected Proceedings of the first PASCAL Challenges Workshop*. LNAI, Springer, 2006. 103
- Z.-G. Fan and B.-L. Lu. Fast recognition of multi-view faces with feature selection. In *Proceedings of the 10th International Conference on Computer Vision, Beijing, China*, volume 1, pages 76–81, 2005. 52, 65
- L. Fei-Fei, R. Fergus, and P. Perona. A bayesian approach to unsupervised one-shot learning of object categories. In *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, pages 1134–1141, Oct. 2003. 80
- P. Felzenszwalb and D. Huttenlocher. Efficient matching of pictorial structures. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, pages 66–75, 2000. 80
- R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA*, volume II, pages 264–271, 2003. 21, 41, 80, 83, 90
- F. Fleuret. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5:1531–1555, 2004. 52, 62, 68
- L. M. J. Florack, B. M. ter Haar Romeny, J. J. Koenderink, and M. A. Viergever. Scale and the differential structure of images. *Image and Vision Computing*, 10: 376–388, 1992. 26
- G. Forman. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research*, 3:1289–1305, 2003. ISSN 1533-7928. 50, 51, 55, 60, 64
- G. Forman. A pitfall and solution in multi-class feature selection for text classification. In *Proceedings of the 21st International Conference on Machine learning, Banff, Alberta, Canada*, page 38, New York, NY, USA, 2004. ACM Press. ISBN 1-58113-828-5. 65

- W. Freeman and E. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991. 29
- Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Proceedings of the 13th International Conference on Machine Learning, Bari, Italy*, pages 148–156, July 1996a. 61
- Y. Freund and R. E. Schapire. Game theory, on-line prediction and boosting. In *COLT '96: Proceedings of the 9th Annual Conference on Computational Learning Theory*, pages 325–332, New York, NY, USA, 1996b. ACM Press. ISBN 0-89791-811-8. 61
- E. Gabrilovich and S. Markovitch. Text categorization with many redundant features: Using aggressive feature selection to make svms competitive with c4.5. In *Proceedings of the 21st International Conference on Machine Learning, Banff, Alberta, Canada*, page 41, New York, NY, USA, 2004. ACM Press. ISBN 1-58113-828-5. 50
- L. Galavotti, F. Sebastiani, and M. Simi. Experiments on the use of feature selection and negative evidence in automated text categorization. In *ECDL '00: Proceedings of the 4th European Conference on Research and Advanced Technology for Digital Libraries*, pages 59–68, London, UK, 2000. Springer-Verlag. ISBN 3-540-41023-6. 57, 59
- A. Grossmann and J. Morlet. Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM Review*, 15:723–736, 1984. 29
- C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, 1988. 22, 25, 30
- E. Hayman, B. Caputo, M. Fritz, and J.-O. Eklundh. On the significance of real-world conditions for material classification. In *Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic*, volume IV, pages 253–266, May 2004. 43
- F. Heitger, L. Rosenthaler, R. von der Heydt, E. Peterhans, and O. Kuebler. Simulation of neural contour mechanism: from simple to end-stopped cells. *Vision Research*, 32(5):963–981, 1992. 25
- R. Horaud, T. Skordas, and F. Veillon. Finding geometric and relational structures in an image. In *Proceedings of the 1st European Conference on Computer Vision, Antibes, France*, pages 374–384, April 1990. 25
- T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In C. Nédellec and C. Rouveirol, editors, *Proceedings of the 10th European Conference on Machine Learning, Chemnitz, Germany*, pages 137–142. Springer Verlag, Heidelberg, Germany, 1998. 50

- T. Joachims. Making large-scale svm learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. The MIT Press, Cambridge, MA, USA, 1999. 46, 68
- G. H. John, R. Kohavi, and K. Pflieger. Irrelevant features and the subset selection problem. In *Proceedings of the 11st International Conference on Machine learning, New Brunswick, NJ, USA*, pages 121–129, 1994. 51
- F. Jurie and C. Schmid. Scale-invariant shape features for recognition of object categories. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Washington, DC, USA*, volume II, pages 90–96, 2004. 23
- F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. volume 1, pages 604–610, 2005. 52
- T. Kadir and M. Brady. Scale, saliency and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001. 66
- T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic*, volume I, pages 228–241, 2004. 22, 23, 67, 79
- Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Washington, DC, USA*, pages 66–75, 2004. 29
- J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–396, 1984. 26
- J. J. Koenderink and A. J. van Doorn. Representation of local geometry in the visual system. *Biological Cybernetics*, 55(6):367–375, 1987. ISSN 0340-1200. 29
- R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997. ISSN 0004-3702. 51
- S. Lazebnik, C. Schmid, and J. Ponce. Sparse texture representation using affine-invariant neighborhoods. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA*, volume 2, pages 319–324, 2003. 21, 23, 29, 32
- S. Lazebnik, C. Schmid, and J. Ponce. Semi-local affine parts for object recognition. In *Proceedings of the 15th British Machine Vision Conference, London, England*, 2004. 102
- S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, August 2005. 29



- B. Leibe and B. Schiele. Scale invariant object categorization using a scale-adaptive mean-shift search. In *Proceedings of the DAGM'04 Annual Pattern Recognition Symposium, Tuebingen, Germany*, volume 3175, pages 145–153. Springer LNCS, August 2004. 81, 91, 92, 94, 95, 103, 107
- T. Lindeberg. Scale-space for discrete signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):234–254, 1990. 26
- T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79–116, 1998. 24, 26, 30
- T. Lindeberg and J. Garding. Shape-adapted smoothing in estimation of 3D depth cues from affine distortions of local 2D brightness structure. In *Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden*, pages 389–400, 1994. 22, 32, 79
- T. Lindeberg and J. Garding. Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure. *Image and Vision Computing*, 15(6):415–434, 1997. 23
- D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 21, 22, 24, 27, 29, 30, 31, 79
- S. Mahamud and M. Hebert. The optimal distance measure for object detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA*, 2003. 51, 79
- J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proceedings of the 13th British Machine Vision Conference, Cardiff, England*, pages 384–393, 2002. 23, 24, 28, 39, 44, 79
- K. Mikolajczyk. *Detection of local features invariant to affine transformations*. PhD thesis, Institut National Polytechnique de Grenoble, 2002. 43, 45
- K. Mikolajczyk, B. Leibe, and B. Schiele. Local features for object class recognition. In *Proceedings of the 10th International Conference on Computer Vision, Beijing, China*, volume 2, pages 1792–1799, 2005a. 23
- K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *Proceedings of the 8th International Conference on Computer Vision, Vancouver, Canada*, volume 1, pages 525–531, 2001. 27, 33
- K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004a. Accepted. 21, 29

- K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004b. [22](#), [23](#), [24](#), [28](#), [32](#), [79](#)
- K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 2005b. Accepted. [22](#), [23](#), [34](#), [35](#), [36](#), [39](#), [40](#), [46](#)
- D. Mladenić, J. Brank, M. Grobelnik, and N. Milić-Frayling. Feature selection using linear classifier weights: Interaction with classification models. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, Sheffield, United Kingdom*, pages 234–241, New York, NY, USA, 2004. ACM Press. ISBN 1-58113-881-4. [50](#), [55](#), [62](#), [63](#)
- D. Mladenić and M. Grobelnik. Feature selection for unbalanced class distribution and naïve bayes. In *Proceedings of the 16th International Conference on Machine learning, Bled, Slovenia*, pages 258–267, June 1999. [55](#)
- A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4):349–361, April 2001. [79](#)
- P. Moreels and P. Perona. Evaluation of feature detectors and descriptors based on 3d objects. In *Proceedings of the 10th International Conference on Computer Vision, Beijing, China*, volume 1, pages 800–807, 2005. [23](#), [29](#)
- H. T. Ng, W. B. Goh, and K. L. Low. Feature selection, perceptron learning, and a usability case study for text categorization. In *Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval, Philadelphia, Pennsylvania, USA*, pages 67–73, New York, NY, USA, 1997. ACM Press. [59](#), [64](#)
- 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, Prague, Czech Republic*, volume II, pages 71–84, 2004. [21](#), [41](#), [51](#), [67](#), [81](#), [83](#), [90](#)
- C. Papageorgiou and T. Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33, 2000. [80](#)
- A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw Hill, 1991. [59](#)
- T. Rikert, M. Jones, and P. Viola. A cluster-based statistical model for object detection. In *Proceedings of the 7th International Conference on Computer Vision, Kerkyra, Greece*, pages 1046–1053, 1999. [51](#)

- M. E. Ruiz and P. Srinivasan. Hierarchical neural networks for text categorization. In *Proceedings of the 22th annual international ACM SIGIR conference on Research and development in information retrieval, Berkeley, California, USA*, pages 281–282, New York, NY, USA, 1999. ACM Press. ISBN 1-58113-096-1. [55](#)
- F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets. In *Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark*, volume I, pages 414–431, 2002. [21](#), [29](#)
- C. Schmid. Constructing models for content-based image retrieval. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, USA*, 2001. [51](#), [56](#), [57](#)
- C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–534, May 1997. [15](#)
- C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2):151–172, 2000. [25](#)
- H. Schneiderman and T. Kanade. A statistical method for 3D object detection applied to faces and cars. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, volume I, pages 746–751, 2000. [80](#)
- F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002. ISSN 0360-0300. [51](#), [57](#)
- C. E. Shannon. A mathematical theory of communication. *Bell Sys. Tech. J.*, 27:379–423, 623–656, 1948. [59](#)
- K. Shih, Y. han Chang, J. Rennie, and D. Karger. Not too hot, not too cold: The bundled-svm is just right! In *ICML-2002 Workshop on Text Learning*, 2002. [63](#)
- V. Sindhwani, P. Bhattacharya, and S. Rakshit. Information-theoretic feature crediting in multiclass support vector machines. In *Proceedings of the 1th annual SIAM International Conference on Data Mining, Chicago, USA*, April 2001. [62](#)
- J. Sivic, B. Russell, A. A. Efros, A. Zisserman, and B. Freeman. Discovering objects and their location in images. In *Proceedings of the 10th International Conference on Computer Vision, Beijing, China*, October 2005. [81](#)
- J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, 2003. [51](#)

- M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991. 29
- A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Washington, DC, USA*, pages 762–769, 2004. 51
- B. Triggs. Detecting keypoints with stable position, orientation and scale under illumination changes. In *Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic*, volume IV, pages 100–113, May 2004. 22, 24, 27, 44
- T. Tuytelaars and L. Van Gool. Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, 59(1):61–85, 2004. 22, 23
- S. Ullman, E. Sali, and M. Vidal-Naquet. A fragment-based approach to object representation and classification. In *4th International Workshop on Visual Form, Capri, Italy*, May 2001. 52
- L. Van Gool, T. Moons, and D. Ungureanu. Affine / photometric invariants for planar intensity patterns. In *Proceedings of the 4th European Conference on Computer Vision, Cambridge, England*, pages 642–651, 1996. 29
- C. van Rijsbergen. *Information Retrieval*. Butterworths, 1979. 60
- V. Vapnik. *The nature of statistical learning theory*. Springer-Verlag, 1995. 62
- M. Vidal-Naquet and S. Ullman. Object recognition with informative features and linear classification. In *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, pages 281–288, 2003. 52, 62
- P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, USA*, volume I, pages 511–518, 2001. 51
- P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *Proceedings of the 9th International Conference on Computer Vision, Nice, France*, volume 1, pages 734–741, 2003. 80
- M. Weber, M. Welling, and P. Perona. Towards automatic discovery of object categories. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, volume 2, pages 101–108, 2000a. 83, 90

- M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *Proceedings of the 6th European Conference on Computer Vision, Dublin, Ireland*, pages 18–32, 2000b. [51](#), [55](#), [80](#)
- J. Willamowski, D. Arregui, G. Csurka, C. R. Dance, and L. Fan. Categorizing nine visual classes using local appearance descriptors. In *International Workshop on Learning for Adaptable Visual Systems (LAVS04), Cambridge, United Kingdom*, August 2004. [51](#), [81](#), [90](#)
- J. Winn, A. Criminisi, and T. P. Minka. Object categorization by learned universal visual dictionary. In *Proceedings of the 10th International Conference on Computer Vision, Beijing, China*, volume 1, pages 1800–1807, October 2005. [81](#)
- A. Witkin. Scale-space filtering. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence, Karlsruhe, Germany*, pages 1019–1023, 1983. [26](#)
- H. Yang and J. Moody. Feature selection based on joint mutual information. In *Advances in Intelligent Data Analysis (AIDA), Computational Intelligence Methods and Applications (CIMA), International Computer Conventions, Rochester, New York, USA*, June 1999. [62](#)
- Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In D. H. Fisher, editor, *Proceedings of the 14th International Conference on Machine learning, Nashville, USA*, pages 412–420. Morgan Kaufmann Publishers, San Francisco, US, 1997. [55](#), [57](#)
- Z. Zheng, R. Srihari, and S. Srihari. A feature selection framework for text filtering. In *Proceedings of the 3rd IEEE International Conference on Data Mining, Washington, DC, USA*, page 705, Washington, DC, USA, 2003. IEEE Computer Society. ISBN 0-7695-1978-4. [64](#)
- Z. Zheng, X. Wu, and R. Srihari. Feature selection for text categorization on imbalanced data. *SIGKDD Explorations Newsletter, Special issue on learning from imbalanced datasets*, 6(1):80–89, 2004. [57](#), [64](#)