Foundations and Trends® in
Computer Graphics and Vision
Vol. 3, No. 3 (2007) 177–280

DOI: 10.1561/0600000017

Local Invariant Feature Detectors: A Survey

Tinne Tuytelaars[1] and Krystian Mikolajczyk[2]

[1] *Department of Electrical Engineering, Katholieke Universiteit Leuven,
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium,
Tinne.Tuytelaars@esat.kuleuven.be*

[2] *School of Electronics and Physical Sciences, University of Surrey,
Guildford, Surrey, GU2 7XH, UK, K.Mikolajczyk@surrey.ac.uk*

Abstract

In this survey, we give an overview of invariant interest point detectors, how they evolved over time, how they work, and what their respective strengths and weaknesses are. We begin with defining the properties of the ideal local feature detector. This is followed by an overview of the literature over the past four decades organized in different categories of feature extraction methods. We then provide a more detailed analysis of a selection of methods which had a particularly significant impact on the research field. We conclude with a summary and promising future research directions.

1

Introduction

In this section, we discuss the very nature of local (invariant) features. What do we mean with this term? What is the advantage of using local features? What can we do with them? What would the ideal local feature look like? These are some of the questions we attempt to answer.

1.1 What are Local Features?

A local feature is an image pattern which differs from its immediate neighborhood. It is usually associated with a change of an image property or several properties simultaneously, although it is not necessarily localized exactly on this change. The image properties commonly considered are intensity, color, and texture. Figure 1.1 shows some examples of local features in a contour image (left) as well as in a grayvalue image (right). Local features can be points, but also edgels or small image patches. Typically, some measurements are taken from a region centered on a local feature and converted into descriptors. The descriptors can then be used for various applications.

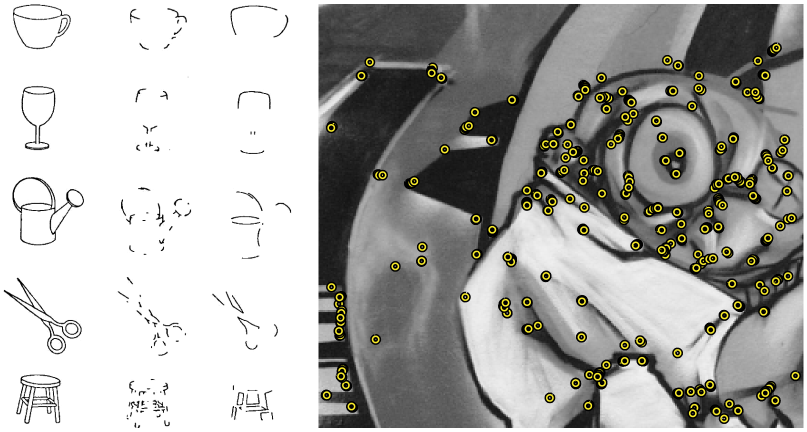

Fig. 1.1 Importance of corners and junctions in visual recognition [20] and an image example with interest points provided by a corner detector (cf. Section 3.2).

1.2 Why Local Features?

As discussed shortly in the preface, local (invariant) features are a powerful tool, that has been applied successfully in a wide range of systems and applications.

In the following, we distinguish three broad categories of feature detectors based on their possible usage. It is not exhaustive or the only way of categorizing the detectors but it emphasizes different properties required by the usage scenarios. *First*, one might be interested in a specific type of local features, as they may have a specific semantic interpretation in the limited context of a certain application. For instance, edges detected in aerial images often correspond to roads; blob detection can be used to identify impurities in some inspection task; etc. These were the first applications for which local feature detectors have been proposed. *Second*, one might be interested in local features since they provide a limited set of well localized and individually identifiable anchor points. What the features actually represent is not really relevant, as long as their location can be determined accurately and in a stable manner over time. This is for instance the situation in most matching or tracking applications, and especially for camera calibration or 3D reconstruction. Other application domains include pose estimation, image alignment or mosaicing. A typical example here are the features used in the KLT tracker [228]. *Finally*, a set of local features can be used as a robust image representation, that allows to recognize objects or scenes without the need for segmentation. Here again, it does not really matter what the features actually represent. They do not even have to be localized precisely, since the goal is not to match them on an individual basis, but rather to analyze their statistics. This way of exploiting local features was first reported in the seminal work of [213] and [210] and soon became very popular, especially in the context of object recognition (both for specific objects as well as for category-level recognition). Other application domains include scene classification, texture analysis, image retrieval, and video mining.

Clearly, each of the above three categories imposes its own constraints, and a good feature for one application may be useless in the context of a different problem. These categories can be considered when searching for suitable feature detectors for an application at hand. In this survey, we mainly focus on the second and especially the third application scenario.

Finally, it is worth noting that the importance of local features has also been demonstrated in the context of object recognition by the human visual system [20]. More precisely, experiments have shown that removing the corners from images impedes human recognition, while removing most of the straight edge information does not. This is illustrated in Figure 1.1.

1.3 A Few Notes on Terminology

Before we discuss feature detectors in more detail, let us explain some terminology commonly used in the literature.

1.3.1 Detector or Extractor?

Traditionally, the term *detector* has been used to refer to the tool that extracts the features from the image, e.g., a corner, blob or edge detector. However, this only makes sense if it is *a priori* clear what the corners, blobs or edges in the image are, so one can speak of "false detections" or "missed detections." This only holds in the first usage scenario mentioned earlier, not for the last two, where *extractor* would probably be semantically more correct. Still, the term *detector* is widely used. We therefore also stick to this terminology.

1.3.2 Invariant or Covariant?

A similar discussion holds for the use of "invariant" or "covariant." A function is invariant under a certain family of transformations if its value does not change when a transformation from this family is applied to its argument. A function is covariant when it commutes with the transformation, i.e., applying the transformation to the argument of the function has the same effect as applying the transformation to the output of the function. A few examples may help to explain the difference. The area of a 2D surface is invariant under 2D rotations, since rotating a 2D surface does not make it any smaller or bigger. But the orientation of the major axis of inertia of the surface is covariant under the same family of transformations, since rotating a 2D surface will affect the orientation of its major axis in exactly the same way. Based on these definitions, it is clear that the so-called local scale and/or affine invariant features are in fact only covariant. The descriptors derived from them, on the other hand, are usually invariant, due to a normalization step. Since the term local invariant feature is so widely used, we nevertheless use "invariant" in this survey.

1.3.3 Rotation Invariant or Isotropic?

A function is isotropic at a particular point if it behaves the same in all directions. This is a term that applies to, e.g., textures, and should not be confused with rotational invariance.

1.3.4 Interest Point, Region or Local Feature?

In a way, the ideal local feature would be a point as defined in geometry: having a location in space but no spatial extent. In practice however, images are discrete with the smallest spatial unit being a pixel and discretization effects playing an important role. To localize features in images, a local neighborhood of pixels needs to be analyzed, giving all local features some implicit spatial extent. For some applications (e.g., camera calibration or 3D reconstruction) this spatial extent is completely ignored in further processing, and only the location derived from the feature extraction process is used (with the location sometimes determined up to sub-pixel accuracy). In those cases, one typically uses the term *interest point*.

However, in most applications those features also need to be described, such that they can be identified and matched, and this again calls for a local neighborhood of pixels. Often, this neighborhood is taken equal to the neighborhood used to localize the feature, but this need not be the case. In this context, one typically uses the term *region* instead of interest point. However, beware: when a local neighborhood of pixels is used to describe an interest point, the feature extraction process has to determine not only the location of the interest point, but also the size and possibly the shape of this local neighborhood. Especially in case of geometric deformations, this significantly complicates the process, as the size and shape have to be determined in an invariant (covariant) way.

In this survey, we prefer the use of the term *local feature*, which can be either points, regions or even edge segments.

1.4 Properties of the Ideal Local Feature

Local features typically have a spatial extent, i.e., the local neighborhood of pixels mentioned above. In contrast to classical segmentation, this can be any subset of an image. The region boundaries do not have to correspond to changes in image appearance such as color or texture. Also, multiple regions may overlap, and "uninteresting" parts of the image such as homogeneous areas can remain uncovered.

Ideally, one would like such local features to correspond to semantically meaningful object parts. In practice, however, this is unfeasible, as this would require high-level interpretation of the scene content, which is not available at this early stage. Instead, detectors select local features directly based on the underlying intensity patterns.

Good features should have the following properties:

- *Repeatability*: Given two images of the same object or scene, taken under different viewing conditions, a high percentage of the features detected on the scene part visible in both images should be found in both images.
- *Distinctiveness/informativeness*: The intensity patterns underlying the detected features should show a lot of variation, such that features can be distinguished and matched.
- *Locality*: The features should be local, so as to reduce the probability of occlusion and to allow simple model approximations of the geometric and photometric deformations between two images taken under different viewing conditions (e.g., based on a local planarity assumption).
- *Quantity*: The number of detected features should be sufficiently large, such that a reasonable number of features are detected even on small objects. However, the optimal number of features depends on the application. Ideally, the number of detected features should be adaptable over a large range by a simple and intuitive threshold. The density of features should reflect the information content of the image to provide a compact image representation.
- *Accuracy*: The detected features should be accurately localized, both in image location, as with respect to scale and possibly shape.
- *Efficiency*: Preferably, the detection of features in a new image should allow for time-critical applications.

Repeatability, arguably the most important property of all, can be achieved in two different ways: either by invariance or by robustness.

- *Invariance*: When large deformations are to be expected, the preferred approach is to model these mathematically if possible, and then develop methods for feature detection that are unaffected by these mathematical transformations.
- *Robustness*: In case of relatively small deformations, it often suffices to make feature detection methods less sensitive to

> such deformations, i.e., the accuracy of the detection may decrease, but not drastically so. Typical deformations that are tackled using robustness are image noise, discretization effects, compression artifacts, blur, etc. Also geometric and photometric deviations from the mathematical model used to obtain invariance are often overcome by including more robustness.

1.4.1 Discussion

Clearly, the importance of these different properties depends on the actual application and settings, and compromises need to be made.

Repeatability is required in all application scenarios and it directly depends on the other properties like invariance, robustness, quantity etc. Depending on the application increasing or decreasing them may result in higher repeatability.

Distinctiveness and locality are competing properties and cannot be fulfilled simultaneously: the more local a feature, the less information is available in the underlying intensity pattern and the harder it becomes to match it correctly, especially in database applications where there are many candidate features to match to. On the other hand, in case of planar objects and/or purely rotating cameras (e.g., in image mosaicing applications), images are related by a global homography, and there are no problems with occlusions or depth discontinuities. Under these conditions, the size of the local features can be increased without problems, resulting in a higher distinctiveness.

Similarly, an increased level of *invariance* typically leads to a reduced *distinctiveness*, as some of the image measurements are used to lift the degrees of freedom of the transformation. A similar rule holds for *robustness versus distinctiveness*, as typically some information is disregarded (considered as noise) in order to achieve robustness. As a result, it is important to have a clear idea on the required level of invariance or robustness for a given application. It is hard to achieve high invariance and robustness at the same time and invariance, which is not adapted to the application, may have a negative impact on the results.

Accuracy is especially important in wide baseline matching, registration, and structure from motion applications, where precise correspondences are needed to, e.g., estimate the epipolar geometry or to calibrate the camera setup.

Quantity is particularly useful in some class-level object or scene recognition methods, where it is vital to densely cover the object of interest. On the other hand, a high number of features has in most cases a negative impact on the computation time and it should be kept within limits. Also robustness is essential for object class recognition, as it is impossible to model the intra-class variations mathematically, so full invariance is impossible. For these applications, an accurate localization is less important. The effect of inaccurate localization of a feature detector can be countered, up to some point, by having an extra robust descriptor, which yields a feature vector that is not affected by small localization errors.

1.5 Global versus Local Features

Local invariant features not only allow to find correspondences in spite of large changes in viewing conditions, occlusions, and image clutter (wide baseline matching), but also yield an interesting description of the image content for image retrieval and object or scene recognition tasks (both for specific objects as well as categories). To put this into context, we briefly summarize some alternative strategies to compute image representations including global features, image segments, and exhaustive and random sampling of features.

1.5.1 Global Features

In the field of image retrieval, many global features have been proposed to describe the image content, with color histograms and variations thereof as a typical example [237]. This approach works surprisingly well, at least for images with distinctive colors, as long as it is the overall composition of the image as a whole that the user is interested in, rather than the foreground object. Indeed, global features cannot distinguish foreground from background, and mix information from both parts together.

Global features have also been used for object recognition, resulting in the first appearance-based approaches to tackle this challenging problem. Turk and Pentland [245] and later Murase and Nayar [160] proposed to compute a principal component analysis of a set of model images and to use the projections onto the first few principal components as descriptors. Compared to the purely geometry-based approaches tried before, the results of the novel appearance-based approach were striking. A whole new range of natural objects could suddenly be recognized. However, being based on a global description, image clutter and occlusions again form a major problem, limiting the usefulness of the system to cases with clean backgrounds or where the object can be segmented out, e.g., relying on motion information.

1.5.2 Image Segments

An approach to overcome the limitations of the global features is to segment the image in a limited number of regions or segments, with each such region corresponding to a single object or part thereof. The best known example of this approach is the blobworld system, proposed in [31], which segments the image based on color and texture, then searches a database for images with similar "image blobs." An example based on texture segmentation is the wide baseline matching work described in [208].

However, this raises a chicken-and-egg problem as image segmentation is a very challenging task in itself, which in general requires a high-level understanding of the image content. For generic objects, color and texture cues are insufficient to obtain meaningful segmentations.

1.5.3 Sampled Features

A way to deal with the problems encountered with global features or image segmentations, is to *exhaustively sample* different subparts of the image at each location and scale. For each such image subpart, global features can then be computed. This approach is also referred to as a *sliding window* based approach. It has been especially popular in the context of face detection, but has also been applied for the recognition of specific objects or particular object classes such as pedestrians or cars.

By focusing on subparts of the image, these methods are able to find similarities between the queries and the models in spite of changing backgrounds, and even if the object covers only a small percentage of the total image area. On the downside, they still do not manage to cope with partial occlusions, and the allowed shape variability is smaller than what is feasible with a local features based approach. However, by far the biggest drawback is the inefficiency of this approach. Each and every subpart of the image must be analyzed, resulting in thousands or even millions of features per image. This requires extremely efficient methods which significantly limits the scope of possible applications.

To overcome the complexity problems more sparse *fixed grid sampling* of image patches was used (e.g., [30, 62, 246, 257]). It is however difficult to achieve invariance to geometric deformations for such features. The approach can tolerate some deformations due to dense sampling over possible locations, scales, poses etc. 00, but the individual features are not invariant. An example of such approach are multi-scale interest points. As a result, they cannot be used when the goal is to find precise correspondences between images. However, for some applications such as scene classification or texture recognition, they may well be sufficient. In [62], better results are reported with a fixed grid of patches than with patches centered on interest points, in the context of scene classification work. This can be explained by the dense coverage, as well as the fact that homogeneous areas (e.g., sky) are also taken into account in the fixed grid approach which makes the representation more complete. This dense coverage is also exploited in [66], where a fixed grid of patches was used on top of a set of local invariant features in the context of specific object recognition, where the latter supply an initial set of correspondences, which then guide the construction of correspondences for the former.

In a similar vein, rather than using a fixed grid of patches, a *random sampling* of image patches can also be used (e.g., [97, 132, 169]). This gives a larger flexibility in the number of patches, the range of scales or shapes, and their spatial distribution. Good scene recognition results are shown in [132] based on random image patches. As in the case of fixed grid sampling, this can be explained by the dense coverage which ignores the localization properties of features. Random patches are in fact a subset of the dense patches, and are used mostly to reduce the complexity. Their repeatability is poor hence they work better as an addition to the regular features rather than as a stand alone method.

Finally, to overcome the complexity problems while still providing a large number of features with better than random localization [140, 146] proposed to sample features uniformly from edges. This proved useful for dealing with wiry objects well represented by edges and curves.

1.6 Overview of this Survey

This survey article consists of two parts. First, in Section 2, we review local invariant feature detectors in the literature, from the early days in computer vision up to the most recent evolutions. Next, we describe a few selected, representative methods in more detail. We have structured the methods in a relatively intuitive manner, based on the type of feature extracted in the image. Doing so, we distinguish between corner detectors (Section 3), blob detectors (Section 4), and region detectors (Section 5). Additionally, we added a section on various detectors that have been designed in a computationally efficient manner (Section 6). With this structure, we hope the reader can easily find the type of detector most useful for his/her application. We conclude the survey with a qualitative comparison of the different methods and a discussion of future work (Section 7).

To the novice reader, who is not very familiar with local invariant feature detectors yet, we advice to skip Section 2 at first. This section has been added mainly for the more advanced reader, to give further insight in how this field evolved and what were the most important trends and to add pointers to earlier work.

2

Local Features in the Literature

In this section, we give an overview of local feature detectors proposed in the literature, starting from the early days of image processing and pattern recognition up to the current state-of-the-art.

2.1 Introduction

The literature on local feature detection is vast and goes back as far as 1954, when it was first observed by Attneave [6] that information on shape is concentrated at dominant points having high curvature. It is impossible to describe each and every contribution to over 50 years of research in detail. Instead, we provide pointers to the literature where the interested reader can find out more. The main goal of this section is to make the reader aware of the various great ideas that have been proposed, especially in the pre-internet era. All too often, these are overlooked and then re-invented. We would like to give proper credit to all those researchers who contributed to the current state-of-the-art.

2.1.1 Early Work on Local Features

It is important to mention the beginnings of this research area and the first publications which appeared after the observation on the importance of corners and junctions in visual recognition [6] (see Figure 1.1). Since then a large number of algorithms have been suggested for extracting interest points at the extrema of various functions computed on the digital shape. Also, it has been understood early on in the image processing and visual pattern recognition field that intersections of straight lines and straight corners are strong indications of man made structures. Such features have been used in a first series of applications from line drawing images [72] and photomosaics [149]. First monographs on digital image processing by Rosenfeld [191] and by Duda and Hart [58] as well as their later editions served to establish the field on a sound theoretical foundation.

2.1.2 Overview

We identified a number of important research directions and structured the subsections of this section accordingly. First, many authors have studied the curvature of contours to find corners. Their work is described in Section 2.2. Others directly analyze the image intensities, e.g., based on derivatives or regions with high variance. This is the topic of Section 2.3. Another line of research has been inspired by the human visual system and aims at reproducing the processes in the human brain — see Section 2.4. Methods focussing on the exploitation of color information are discussed in Section 2.5, while Section 2.6 describes model-based approaches. More recently, there has been a trend toward feature detection with invariance against various geometric transformations, including multi-scale approaches and scale or affine invariant methods. These are discussed in Section 2.7. In Section 2.8, we focus on segmentation-based methods and Section 2.9 describes methods which build on machine learning techniques. Finally, Section 2.10 gives an overview of different evaluation and comparison schemes proposed in the literature.

2.2 Contour Curvature Based Methods

A first category of interest point detectors are the contour curvature based methods. Originally, these were mainly applied to line drawings, piecewise constant regions, and cad–cam images rather than natural scenes. The focus was especially on the accuracy of point localization. They were most popular of the end of the 1970s and most of the 1980s.

2.2.1 High Curvature Points

Contour intersections and junctions often result in bi-directional signal changes. Therefore, a good strategy to detect features consists of extracting points along the contour with high curvature. Curvature of an analog curve is defined as the rate at which the unit tangent vector changes with respect to arc length. Contours are often encoded in chains of points or represented in a parametric form using splines.

Several techniques have been developed which involve detecting and chaining edges so as to find corners in the chain by analyzing the chain code [205], finding maxima of curvature [108, 136, 152], change in direction [83], or change in appearance [42]. Others avoid chaining edges and instead look for maxima of curvature [254] or change in direction [104] at places where the gradient is large.

Several methods for detecting edges based on gray-level gradient and angular changes in digital curves were proposed in [193, 195, 196, 197]. Other solutions for line-drawing images include methods for detecting corners in a chain-coded plane curve [73, 74]. In these works, a measure for the cornerness of a point is based on mean angular differences between successive segment positions along the chain.

One general approach to feature extraction is to detect the dominant points directly through angle or corner detection, using various schemes for approximating discrete curvature such as cosine [192, 193] or local curvature [18, 74] which define corners as discontinuities of an average curve slope. Other parametric representation like B-splines curves are commonly used in rendering a curve in computer graphics, compression and coding, CAD–CAM systems, and also for curve fitting and shape description [175]. In [108], cubic polynomials are fit to a curve and discontinuities are detected in such curve to localize interest points. Spline approximations of line images are used in [85] in combination with a dynamic programming technique to find the knots of a spline. Pseudo coding of line figures and a complicated vector finder to obtain interest points are proposed in [164].

In [207], dominant points are computed at the maximum global curvature, based on the iterative averaging of local discretized curvature at each point with respect to its immediate neighbors. In [3], tangential deflection and curvature of discrete curves are defined based on the geometrical and statistical properties associated with the eigenvalue–eigenvector structure of sample covariance matrices computed on chain-codes.

Another approach is to obtain a piecewise linear polygonal approximation of the digital curve subject to certain constraints on the quality of fit [60, 174, 176]. Indeed, it has been pointed out in [174] that piecewise linear polygonal approximation with variable breakpoints will tend to locate vertices at actual corner points. These points correspond approximately to the actual or extrapolated intersections of adjacent line segments of the polygons. A similar idea was explored in [91]. More recently, [95] estimates the parameters of two lines fitted to the two segments neighboring to the corner point. A corner is declared if the parameters are statistically significantly different. A similar approach is to identify edge crossings and junctions [19] by following image gradient maxima or minima and finding gaps in edge maps.

2.2.2 Dealing with Scale

Corner detection methods by curvature estimation normally use a set of parameters to eliminate contour noise and to obtain the corners at a given scale, although object corners can be found at multiple natural scales. To solve this problem, some detectors apply their algorithms iteratively within a certain range of parameters, selecting points which appear in a fixed set of iterations. The stability of the points and the time spent for their detection is closely related to the number of iterations.

Initial attempts to deal with discretization and scale problems via an averaging scheme can be found in [207]. The curvature primal sketch (CPS) proposed in [5] is a scale-space representation of significant changes in curvature along contours. The changes are classified as basic or compound primitives such as corners, smooth joints, ends, cranks, bumps, and dents. The features are detected at different scales, resulting in a multiple-scale representation of object contours. A similar idea was explored in [151, 152] and later in [86], where the curvature scale space analysis was performed to find the local scale of curves. They find inflection points of the curves and represent shapes in parametric forms. A B-spline based algorithm was also proposed in [108, 136]. The general idea is to fit a B-Spline to the curve, then to measure the curvature around each point directly from the B-spline coefficients.

Another algorithm [238] dealing with scale for detecting dominant points on a digital closed curve is motivated by the angle detection procedure from [193]. They indicate that the detection of dominant points relies primarily on the precise determination of the region of support rather than on the estimation of discrete curvature. First, the region of support for each point based on its local properties is determined. Then a measure of relative curvature [238] or local symmetry [170] of each point is computed. The Gaussian filter is the most commonly used filter in point detection. However, if the scale of a Gaussian filter is too small, the result may include some redundant points which are unnecessary details, i.e., due to noise. If the scale is too large, the points with small support regions will tend to be smoothed out. To solve the problems existing in Gaussian filtering with fixed scale, *scale-space procedures* based on multiple-scale discrete curvature representation and searching are proposed in [4, 181]. The scheme is based on a stability criterion that states that the presence of a corner must concur with a curvature maximum observable at a majority of scales. Natural scales of curves were studied in [199] to avoid exhaustive representation of curves over a full range of scales. A successful *scale selection* mechanism for Gaussian filters with a theoretical formulation was also proposed in [119, 120].

In [264] a nonlinear algorithm for critical point detection is presented. They establish a set of criteria for the design of a point detection algorithm to overcome the problems arising from curvature approximation and Gaussian filtering. Another approach to boundary smoothing is based on simulated annealing for curvature estimation [233]. In [152] the corner points are localized at the maxima of absolute curvature of edges. The corner points are tracked through multiple curvature scale levels to improve localization. Chang and Horng [33] proposed an algorithm to detect corner points using a nest moving average filter is investigated in [33]. Corners are detected on curves by computing the difference of blurred images and observing the shift of high curvature points. More detailed analysis of various methods for determining natural scales of curves can be found in [125, 199, 200].

2.2.3 Discussion

Although theoretically well founded for analog curves, the contour curvature calculation is less robust in case of discrete curves [194, 238]. Possible error sources in digital curvature estimation were investigated in [259].

Furthermore, the objectives for the above discussed detectors were different than the ones we typically have nowadays. It was considered disadvantageous if a method detected corners on circular shapes, multiple corners at junctions etc. At that time, a much stricter definition of interest points/corners was used, with only points corresponding to true corners in 3D being considered as relevant. Nowadays, in most practical applications of interest points, the focus is on robust, stable, and distinctive points, irrespective of whether they correspond to true corners or not (see also our earlier discussion in Section 1.2).

There has been less activity in this area recently (over the past ten years), due to complexity and robustness problems, while methods based directly on image intensity attracted more attention.

2.3 Intensity Based Methods

Methods based on image intensity have only weak assumptions and are typically applicable to a wide range of images. Many of these approaches are based on first- and second-order gray-value derivatives, while others use heuristics to find regions of high variance.

2.3.1 Differential Approaches

Hessian-based approaches. One of the early intensity based detectors is the rotation invariant *Hessian*-based detector proposed by

Beaudet [16]. It explores the second-order Taylor expansion of the intensity surface, and especially the Hessian matrix (containing the second order derivatives). The determinant of this matrix reaches a maximum for blob-like structures in the image. A more detailed description of this method can be found in Section 4.1. It has been extended in [57] and [266], where the interest points are localized at the zero crossing of a curve joining local extrema of the Hessian determinant around a corner.

Similarly, high curvature points can be localized by computing Gaussian curvature of the image surface, i.e., saddle points in image brightness. In [104], a local quadratic surface was fit to the image intensity function. The parameters of the surface were used to determine the gradient magnitude and the rate of change of gradient direction. The resulting detector uses the curvature of isophotes computed from first- and second-order derivatives scaled by image gradient to make it more robust to noise. A similar idea was proposed in [61, 229].

A detailed investigation in [168, 167, 224] and later in [83] shows that the detectors of [16, 57, 104, 163, 266] all perform the same measurements on the image and have relatively low reliability according to criteria based on localization precision. Nevertheless, the trace and determinant of the Hessian matrix were successfully used later on in scale and affine invariant extensions of interest point detectors [121, 143] when other feature properties became more important.

Gradient-based approaches. Local feature detection based on first-order derivatives is also used in various applications. A corner detector which returns points at the local maxima of a directional variance measure was first introduced in [154, 155, 156] in the context of mobile robot navigation. It was a heuristic implementation of the auto-correlation function also explored in [41]. The proposed corner detector investigates a local window in the image and determines the average change of intensity which results from shifting the window by a few pixels in various directions. This idea is taken further in [69, 70] and formalized by using first-order derivatives in a so-called *second moment matrix* to explore local statistics of directional image intensity variations. The method separates corner candidate detection and localization to improve the accuracy to subpixel precision, at the cost of higher computational complexity. Harris and Stephens [84] improved the approach by Moravec [155] by performing analytical expansion of the average intensity variance. This results in a second moment matrix computed with Sobel derivatives and a Gaussian window. A function based on the determinant and trace of that matrix was introduced which took into account both eigenvalues of the matrix. This detector is widely known today as the *Harris* detector or *Plessey* detector,[1] and is probably the best known interest point detector around. It is described in more detail in Section 3.2. It has been extended in numerous papers, e.g., by using Gaussian derivatives [212], combinations of first- and second-order derivatives [263], or an edge based second moment matrix [45] but the underlying idea remains the same.

The Harris detector was also investigated in [167] and demonstrated to be optimal for L junctions. Based on the assumption of an affine image deformation, an analysis in [228] led to the conclusion that it is more convenient to use the smallest eigenvalue of the autocorrelation matrix as the corner strength function.

More recently, the second moment matrix has also been adopted to scale changes [59] by parameterizing Gaussian filters and normalizing them with respect to scale, based on scale-space theory [115, 117]. Also, the Harris detector was extended with search over scale and affine space in [13, 142, 209], using the Laplacian operator and eigenvalues of the second moment matrix, inspired by the pioneering work of Lindeberg [117, 118] (see Section 3.4 for details).

The approach from [263] performs an analysis of the computation of the second moment matrix and its approximations. A speed increase is achieved by computing only two smoothed images, instead of the three previously required. A number of other suggestions have been made for how to compute the corner strength from the second-order matrix [84, 101, 167, 228], and these have all been shown to be equivalent to various matrix norms [102, 265]. A generalization to images with multi-dimensional pixels was also proposed in [102].

[1] Plessey Electronic Research Ltd.

In [242], the Harris corner detector is extended to yield stable features under more general transformations than pure translations. To this end, the auto-correlation function was studied under rotations, scalings, up to full affine transformations.

2.3.2 Intensity Variations

A different category of approaches based on intensity variations applies mathematical morphology to extract high curvature points. The use of zero-crossings of the shape boundary curvature in binary images, detected with a morphological opening operator was investigated in [36]. Mathematical morphology was also used to extract convex and concave points from edges in [107, 114, 168]. Later on a parallel algorithm based on an analysis of morphological residues and corner characteristics was proposed in [262].

Another approach [173] indicates that for interest points the median value over a small neighborhood is significantly different from the corner point value. Thus the difference in intensity between the center and median gives a strong indication for corners. However, this method cannot deal with more complex junctions or smooth edges.

A simple and efficient detector named SUSAN was introduced in [232] based on earlier work from [82]. It computes the fraction of pixels within a neighborhood which have similar intensity to the center pixel. Corners can then be localized by thresholding this measure and selecting local minima. The position of the center of gravity is used to filter out false positives. More details on the SUSAN detector can be found in Section 3.3. A similar idea was explored in [112, 240] where pixels on a circle are considered and compared to the center of a patch.

More recently, [203] proposed the FAST detector. A point is classified as a corner if one can find a sufficiently large set of pixels on a circle of fixed radius around the point such that these pixels are all significantly brighter (resp. darker) than the central point. Efficient classification is based on a decision tree. More details on FAST can be found in Section 6.3.

Local radial symmetry has been explored in [127] to identify interest points and its real-time implementation was also proposed. Wavelet transformation was also investigated in the context of feature point extraction with successful results based on multi-resolution analysis in [35, 111, 218].

2.3.3 Saliency

The idea of saliency has been used in a number of computer vision algorithms. The early approach of using edge detectors to extract object descriptions embodies the idea that the edges are more significant than other parts of the image. More explicit uses of saliency can be divided into those that concentrate on low-level local features (e.g., [215]), and those that compute salient groupings of low-level features (e.g., [223]); though some approaches operate at both levels (e.g., [147]).

The technique suggested in [211], is based on the maximization of descriptor vectors across a particular image. These salient points are the points on the object which are almost unique. Hence they maximize the discrimination between the objects. A related method [253] identifies salient features for use in automated generation of Statistical Shape/Appearance Models. The method aims to select those features which are less likely to be mismatched. Regions of low density in a multidimensional feature space, generated from the image, are classified as highly salient.

A more theoretically founded approach based on variability or complexity of image intensity within a region was proposed in [79]. It was motivated by visual saliency and information content, which we revise in the next section. The method from [79] defines saliency in terms of local signal complexity or unpredictability; more specifically the use of Shannon entropy of local attributes is suggested. The idea is to find a point neighborhood with high complexity as a measure of saliency or information content. The method measures the change in entropy of a gray-value histogram computed in a point neighborhood. The search was extended to scale [98] and affine [99] parameterized regions, thus providing position, scale, and affine shape of the region neighborhood. For a detailed discussion, we refer to Section 4.3.

2.4 Biologically Plausible Methods

Most systems proposed in the previous sections were mainly concerned with the accuracy of interest point localization. This is important in the context of fitting parametric curves to control points or image matching for recovering the geometry. In contrast, the biologically plausible methods reviewed in this section were mainly proposed in the context of artificial intelligence and visual recognition. Most of them did not have a specific application purpose and their main goal was to model the processes of the human brain. Numerous models of human visual attention or saliency have been discussed in Cognitive Psychology and Computer Vision. However, the vast majority were only of theoretical interest and only few were implemented and tested on real images.

2.4.1 Feature Detection as Part of the Pre-attentive Stage

One of the main models for early vision in humans, attributed to Neisser [165], is that it consists of a pre-attentive and an attentive stage. Biologically plausible methods for feature detection usually refer to the idea that certain parts of a scene are pre-attentively distinctive and create some form of immediate response within the early stages of the human visual system. In the pre-attentive stage, only "pop-out" features are detected. These are local regions of the image which present some form of spatial discontinuity. In the attentive stage, relationships between these features are found, and grouping takes place. This model has widely influenced the computer vision community (mainly through the work of Marr [133]) and is reflected in the classical computer vision approach — feature detection and perceptual grouping, followed by model matching and correspondence search. Activities in the models of attention started in the mid-1980s following progress in neurophysiological and psychological research.

One approach inspired by neuro-biological mechanisms was proposed in [87, 198]. They apply Gabor like filters to compute local energy of the signal. Maxima of the first- and second-order derivatives of that energy indicate the presence of interest points. The idea of using Gabor filter responses from different scales was further explored in [131, 186]. The approach developed in [182] was motivated by psychophysical experiments. They compute a symmetry score of the signal at each image pixel in different directions. Regions with significant symmetry are then selected as interest points.

Theory on texture recognition and the idea of textons as simple local structures like blobs, corners, junctions, line ends etc. was introduced in [96]. He suggested that statistics over texton distributions play an important role in recognition. The extraction of simple textons is done in the pre-attentive stage and the construction of relations in the attentive stage. A feature integration theory based on these principles was proposed in [241]. He distinguished between a *disjunctive* case where the distinctive features can be directly localized in a feature map and a *conjunctive* case where the feature can be extracted only by processing various feature maps simultaneously. This model was implemented by combining bottom up and top down measures of interest [32]. The bottom up method merges various feature maps and looks for interesting events, while in the top down process, knowledge about the target is exploited.

The main goal of the above systems was to provide computationally plausible models of visual attention. Their interest was mainly theoretical. However, those systems served as source of inspiration for practical solutions for real images once machine learning techniques like neural networks had grown mature enough. In [206], image processing operators were combined with the attentive models to make it applicable to more realistic images. He applies a Laplacian-of-Gaussians (LoG) like operator to feature maps to model the receptive fields and enhance the interesting events. The image was analyzed at multiple scales. The approach from [78] uses a set of feature templates and correlates them with the image to produce feature maps which are then enhanced with LoG. Temporal derivatives were used to detect moving objects.

Koch and Ullman [105] proposed a very influential computational model of visual attention which accounts for several psychophysical phenomena. They proposed to build a set of maps based on orientation, color, disparity and motion, and to simulate the lateral inhibition mechanism by extracting locations which differ significantly from their neighborhood. Information from different maps is then merged into a single saliency map. A winner-take-all (WTA) network was used to select the active location in the maps in a hierarchical manner using a pyramidal strategy. The hypotheses suggested in [105, 241] were first implemented in [34]. A similar implementation of the WTA model was proposed in [49].

The extraction of globally salient structures like object outlines was investigated in [223] by grouping local information such as contour fragments but no relation to pre-attentive vision was claimed.

2.4.2 Non-uniform Resolution and Coarse-To-Fine Processing

Also non-uniform resolution of the retina and coarse-to-fine processing strategies have been studied in biologically plausible models. These have been simulated mostly via scale-space techniques [9, 10, 187, 255]. However, these systems were mostly focused on the engineering and realtime aspects rather than its biological plausibility. One of the first systems to perform interest point detection in scale-space was proposed in [27]. They built a Laplacian pyramid for coarse-to-fine feature selection. Templates were used to localize the objects in the LoG space. Templates were also employed for building features maps which were then combined by a weighted sum [39]. Difference-of-Gaussians (DoG) filters were used in the system designed in [76] to accelerate the computation.

Biologically inspired systems developed in [81] explored the idea of using boundary and interest point detectors based on DoG filters as well as directional differences of offset Gaussians (DOOG) to simulate simple cells in V1.

The system proposed in [130] was mainly concerned with classification of textures studied earlier in [96]. The feature extraction part used a bank of filters based on oriented kernels (DoG and DOOG) to produce feature maps similar to [81]. The next stage corresponds to a WTA mechanism to suppress weak responses and simulate lateral inhibition. Finally, all the responses are merged to detect texture boundaries.

2.4.3 Spatial Event Detection

Robust statistics have also been used to detect outliers in a set of image primitives. The idea is based on the observation that textures can be represented by their statistics and the locations which violate those statistics represent interesting events. For example, texture primitives are represented by a number of attributes using histograms and RANSAC in [148].

First order statistics over feature maps computed from zero crossings of DoG at different scales are used in [23]. For each point, a histogram of gradient orientations is then constructed, and the local histograms are combined into a global one, which is similar in spirit to the more recent SIFT descriptor [124, 126]. Local histograms are then compared with the global one to provide a measure of interest.

Another statistical model was proposed in [172]. They measure the edge density at a range of distances from the interest point to build an edge distribution histogram. This idea has been used later in the shape context descriptor of [17].

Cells that respond only to edges and bars which terminate within their receptive field have first been found in [92]. A corner detection algorithm based on a model for such end-stopped cells in the visual cortex was presented in [87, 260]. Furthermore, the notion of end-stopped cells was generalized to color channels in a biologically plausible way based on color opponent processes [260].

A more recent visual attention system also motivated by the early primate visual system, is presented in [94]. Multiscale image features detected at local spatial discontinuities in intensity, color, and orientation are combined into a single topographical saliency map and a neural network selects locations depending on the saliency.

Other recent visual recognition systems inspired by a model of visual cortex V1 which follow models from [185] can be found in [162, 221, 222]. These methods attempt to implement simple and complex cells from visual cortex which are multiscale Gabor and edgel detectors followed by local maxima selection methods.

2.5 Color-based Methods

Color provides additional information which can be used in the process of feature extraction. Several biologically plausible methods reviewed in the previous section use color for building saliency maps [93, 94, 105, 260].

Given the high performance of Harris corners [84], a straightforward extension of the second moment matrix to RGB color space was introduced in [80, 153], incorporating color information in the Harris corner extraction process.

Salient point detection based on color distinctiveness has been proposed in [250]. Salient points are the maxima of the saliency map, which represents distinctiveness of color derivatives in a point neighborhood. In related work [217] they argue that the distinctiveness of color-based salient points is much higher than for the intensity ones. Color ratios between neighboring pixels are used to obtain derivatives independent of illumination, which results in color interest points that are more robust to illumination changes.

Most of the proposed approaches based on color are simple extensions of methods based on the intensity change. Color gradients are usually used to enhance or to validate the intensity change so as to increase the stability of the feature detectors but the pixel intensities remain the main source of information for feature detection.

2.6 Model-based Methods

There have been a few attempts to do an analytical study of corner detection by giving a formal representation of corner points in an image based on differential geometry techniques [82] or contour curvature [53]. For instance, it was found that a gray-level corner point can be found as the point of maximal planar curvature on the line of the steepest gray-level slope [82, 188]. An analytical expression for an optimal function whose convolution with an image has significant values at corner points was investigated in [180].

The methods presented in [82, 201] assume that a corner resembles a blurred wedge, and finds the characteristics of the wedge (the amplitude, angle, and blur) by fitting it to the local image. Several models of junctions of multiple edges were used in [188]. The assumption is that the junctions are formed by homogeneous regions. Parameterized masks are used to fit the intensity structure including position, orientation, intensity, blurring, and edges. The residual is then minimized during the detection. The accuracy is high provided a good initialization of the parameters. The efficiency of the approach in [188] was improved in [52] by using a different blurring function and a method to initialize the parameters. Fitting a corner model to image data was also considered in [137, 171]. For each possible intersection of lines a template was constructed based on the angle, orientation, and scale of the hypothesized corner. The template was then matched to the image in a small neighborhood of the interest point to verify the model. A template-based method for locating the saddle-points was also described in [128], where the corner points correspond to the intersections of saddle-ridge and saddle-valley structures.

A set of fuzzy patterns of contour points were established in [113] and the corner detection was characterized as a fuzzy classification problem of the patterns.

Other model-based methods, aimed at improving the detection accuracy of the Hessian-based corner detector [16], were proposed in [54, 266]. To this end, the responses of the corner detector on a theoretical model over scale-space were analyzed. It was observed that the operator responses at different scales move along the bisector line. It is worth to note that this observation is also valid for the popular Harris corner detector [84]. The exact position of the corner was then computed from two responses indicating the bisector and its intersection with the zero-crossing of the Laplacian response. An affine transformation was also used to fit a model of a corner to an image [22].

A different model-based approach is proposed in [77]. For each type of feature, a parametric model is developed to characterize the local intensity in an image. Projections of intensity profile onto a set of orthogonal Zernike moment-generating polynomials are used to estimate model-parameters and generate the feature map.

An interesting technique is to find corners by fitting a parameterized model with the Generalized Hough transform [51, 226]. In images with extracted edges two lines appear in a parameter space for each corner and the peak occurs at the crossover. Real corner models in the form of templates were considered in [229]. A similarity measure and several alternative matching schemes were applied. Detection and localization accuracy was improved by merging the output of the different matching techniques.

In general, only relatively simple feature models were considered in the above methods and the generalization to images other than polygonal is not obvious. The complexity is also a major drawback in such approaches.

2.7 Toward Viewpoint Invariant Methods

Most of the detectors described so far extract features at a single scale, determined by the internal parameters of the detector. At the end of the 1990s, as local features were more and more used in the context of wide baseline matching and object recognition, there was a growing need for features that could cope with scale changes or even more general viewpoint changes.

2.7.1 Multi-Scale Methods

Most of the detectors described so far extract features at a single scale, determined by the internal parameters of the detector. To deal with scale changes, a straightforward approach consists of extracting points over a range of scales and using all these points together to represent the image. This is referred to as a *multi-scale* or multi-resolution approach [48].

In [59], a scale adapted version of the Harris operator was proposed. Interest points are detected at the local maxima of the Harris function applied at several scales. Thanks to the use of normalized derivatives, a comparable strength of the cornerness measure is obtained for points detected at different scales, such that a single threshold can be used to reject less significant corners over all scales. This scale adapted detector significantly improves the repeatability of interest points under scale changes. On the other hand, when prior knowledge on the scale change between two images is given, the detector can be adapted so as to extract interest points only at the selected scales. This yields a set of points, for which the respective localization and scale perfectly reflect the real scale change between the images.

In general, multi-scale approaches suffer from the same problems as dense sampling of features (cf. Section 1.5). They cannot cope well with the case where a local image structure is present over a range of scales, which results in multiple interest points being detected at each scale within this range. As a consequence, there are many points, which represent the same structure, but with slightly different localization and scale. The high number of points increases the ambiguity and the computational complexity of matching and recognition. Therefore, efficient methods for selecting accurate correspondences and verifying the results are necessary at further steps of the algorithms. In contrast to structure from motion applications, this is less of an issue in the context of recognition where a single point can have multiple correct matches.

2.7.2 Scale-Invariant Detectors

To overcome the problem of many overlapping detections, typical of multiscale approaches, *scale-invariant* methods have been introduced. These automatically determine both the location and scale of the local features. Features are typically circular regions, in that case.

Many existing methods search for maxima in the 3D representation of an image (x,y and $scale$). This idea for detecting local features in scale-space was introduced in the early 1980s [47]. The pyramid representation was computed with low pass filters. A feature point is detected if it is at a local maximum of a surrounding 3D cube and if its absolute value is higher than a certain threshold. Since then many methods for selecting points in scale-space have been proposed. The existing approaches mainly differ in the differential expression used to build the scale-space representation.

A normalized LoG function was applied in [116, 120] to build a scale space representation and search for 3D maxima. The scale-space representation is constructed by smoothing the high resolution image with derivatives of Gaussian kernels of increasing size. *Automatic scale selection* (cf. Section 3.4) is performed by selecting local maxima in scale-space. The LoG operator is circularly symmetric. It is therefore naturally invariant to rotation. It is also well adapted for detecting blob-like structures. The experimental evaluation in [138] shows this function is well suited for automatic scale selection. The scale invariance of interest point detectors with automatic scale selection has also been explored in [24]. Corner detection and blob detection with automatic scale selection were also proposed in a combined framework in [24] for feature tracking with adaptation to spatial and temporal size variations. The interest point criterion that is being optimized for localization need not be the same as the one used for optimizing the scale. In [138], a scale-invariant corner detector, coined Harris-Laplace, and a scale-invariant blob detector, coined Hessian-Laplace were introduced. In these methods, position and scale are iteratively updated until convergence [143]. More details can be found in Sections 3.4 and 4.2.

An efficient algorithm for object recognition based on local 3D extrema in the scale-space pyramid built with DoG filters was introduced in [126]. The local 3D extrema in the pyramid representation determine the localization and the scale of interest points. This method is discussed further in Section 6.1.

2.7.3 Affine Invariant Methods

An affine invariant detector can be seen as a generalization of the scale-invariant ones to non-uniform scaling and skew, i.e., with a different scaling factor in two orthogonal directions and without preserving angles. The non-uniform scaling affects not only the localization and the scale but also the shape of characteristic local structures. Therefore, scale-invariant detectors fail in the case of significant affine transformations.

Affine invariant feature detection, matching, and recognition have been addressed frequently in the past [50, 90, 204]. Here, we focus on the methods which deal with invariant interest point detection.

One category of approaches was concerned with the localization accuracy under affine and perspective transformations. An affine invariant algorithm for corner localization was proposed in [2] which builds on the observations made in [54]. Affine morphological multi-scale analysis is applied to extract corners. The evolution of a corner is given by a linear function formed by the scale and distance of the detected points from the real corner. The location and orientation of the corner is computed based on the assumption that the multiscale points move along the bisector line and the angle indicates the true location. However, in natural scenes a corner can take any form of a bi-directional signal change and in practice the evolution of a point rarely follows the bisector. The applicability of the method is therefore limited to a polygonal like world.

Other approaches were concerned with simultaneous detection of location, size and affine shape of local structures. The method introduced in [247], coined EBR (Edge-Based Regions) starts from Harris corners and nearby intersecting edges. Two points moving along the edges together with the Harris point determine a parallelogram. The points stop at positions where some photometric quantities of the texture covered by the parallelogram reach an extremum. The method can be categorized as a model-based approach as it looks for a specific structure in images, albeit not as strict as most methods described in Section 2.6. More details can be found in Section 3.5. A similar scheme has been explored in [12].

An intensity-based method (IBR, Intensity-Based Regions) was also proposed in [248]. It starts with the extraction of local intensity extrema. The intensity profiles along rays emanating from a local extremum are investigated. A marker is placed on each ray in the place, where the intensity profile significantly changes. Finally, an ellipse is fitted to the region determined by the markers. This method is further discussed in Section 5.1. Somewhat similar in spirit are the Maximally Stable Extremal Regions or MSER proposed in [134] and described in the next section.

A method to find blob-like affine invariant features using an iterative scheme was introduced in [121], in the context of shape from texture. This method based on the affine invariance of shape adapted fixed points was also used for estimating surface orientation from binocular data (shape from disparity gradients). The algorithm explores the properties of the second moment matrix and iteratively estimates the affine deformation of local patterns. It effectively estimates the transformation that would project the patch to a frame in which the eigenvalues of the second moment matrix are equal. This work provided a theoretical background for several other affine invariant detectors.

It was combined with the Harris corner detector and used in the context of matching in [13], hand tracking in [109], fingerprint recognition [1] and for affine rectification of textured regions in [208]. In [13], interest points are extracted at several scales using the Harris detector and then the shape of the regions is adapted to the local image structure using the iterative procedure from [118]. This allows to extract affine invariant descriptors for a given fixed scale and location — that is, the scale and the location of the points are not extracted in an affine invariant way. Furthermore, the multi-scale Harris detector extracts many points which are repeated at the neighboring scale levels. This increases the probability of a mismatch and the complexity.

The Harris-Laplace detector introduced in [141] was extended in [142, 209] by affine normalization with the algorithm proposed in [13, 118, 121]. This detector suffers from the same drawbacks, as the initial location and scale of points are not extracted in an affine invariant way, although the uniform scale changes between the views are handled by the scale-invariant Harris-Laplace detector.

Beyond affine transformations. A scheme that goes even beyond affine transformations and is invariant to projective transformations was introduced in [236]. However, on a local scale, the perspective effect is usually neglectable. More damaging is the effect of non-planarities or non-rigid deformations. This is why a theoretical framework to extend the use of local features to non-planar surfaces has been proposed in

[251], based on the definition of equivalence classes. However, in practice, they have only shown results on straight corners. Simultaneously, an approach invariant to general deformations was developed in [122], by embedding an image as a 2D surface in 3D space and exploiting geodesic distances.

2.8 Segmentation-based Methods

Segmentation techniques have also been employed in the context of feature extraction. These methods were either applied to find homogeneous regions to localize junctions on their boundaries or to directly use these regions as local features. For the generic feature extraction problem, mostly bottom-up segmentation based on low level pixel grouping was considered, although in some specific tasks top-down methods can also be applied. Although significant progress has been made in the analysis and formalization of the segmentation problem, it remains an unsolved problem in the general case. Optimal segmentation is intractable in general due to the large search space of possible feature point groups, in particular in algorithms based on multiple image cues. Moreover, a multitude of definitions of optimal segmentation even for the same image makes it difficult to solve. Nonetheless, several systems using segmentation based interest regions have been developed, especially in the context of retrieval, matching, and recognition.

In early years of computer vision polygonal approximations of images were popular in scene analysis and medical image analysis [9, 25, 58]. These algorithms often involved edge detection and subsequent edge following for region identification. In [63, 64], the vertices of a picture are defined as those points which are common in three or more segmented regions. It can be seen as one of the first attempts to extract interest points using segmentation. Simple segmentation of patches into two regions is used in [123] and the regions are compared to find corners. Unfortunately, the two region assumption makes the usefulness of the method limited.

Another set of approaches represent real images through segmentation [129]. Well performing image segmentation methods are based on graph cuts, where graphs represent connected image pixels [21, 75, 225, 227]. These methods allow to obtain segmentation at the required level of detail. Although semantic segmentation is not reliable, over-segmenting the image can produce many regions which fit to the objects. This approach was explored in [31, 46] and it is particularly appealing for image retrieval problems where the goal is to find similar images via regions with similar properties. In [44], the goal is to create interest operators that focus on homogeneous regions, and compute local image descriptors for these regions. The segmentation is performed on several feature spaces using kernel-based optimization methods. The regions can be individually described and used for recognition but their distinctiveness is low. This direction has recently gained more interest and some approaches use bottom-up segmentation to extract interest regions or so-called superpixels [159, 184] (see also Section 5.3).

In general the disadvantages of this representation are that the segmentation results are still unstable and inefficient for processing large amounts of images. An approach which successfully deals with these problems was taken in [134]. Maximally Stable Extremal Regions (MSER) are extracted with a watershed like segmentation algorithm. The method extracts homogeneous intensity regions which are stable over a wide range of thresholds. The regions are then replaced by ellipses with the same shape moments up to the second-order. Recently, a variant of this method was introduced in [177], which handles the problems with blurred region boundaries by using region isophotes. In a sense, this method is also similar to the IBR method described in Section 5.1, as very similar regions are extracted. More details on MSER can be found in Section 5.2. The method was extended in [56, 161] with tree like representation of watershed evolution in the image.

2.9 Machine Learning-based Methods

The progress in the domain of machine learning and the increase of available computational power allowed learning techniques to enter the feature extraction domain. The idea of learning the attributes of local features from training examples and then using this information to extract features in other images has been around in the vision community for some time but only recently it was more broadly used in real applications. The success of these methods is due to the fact that efficiency, provided by classifiers, became a more desirable property than accuracy of detection.

In [37, 55], a neural network is trained to recognize corners where edges meet at a certain degree, near to the center of an image patch. This is applied to images after edge detection. A similar idea was explored in [244] to improve the stability of curvature measurement of digital curves.

Decision trees [178] have also been used successfully in interest point detection tasks. The idea of using intensity differences between the central points and neighboring points [173, 232, 240] has been adopted in [202, 203]. They construct a decision tree to classify point neighborhoods into corners [203]. The main concern in their work is the efficiency in testing only a fraction of the many possible differences and the tree is trained to optimize that. The approach of [203] was also extended with LoG filters to detect multiscale points in [112]. They use a feature selection technique based on the repeatability of individual interest points over perspective projected images.

A hybrid methodology that integrates genetic algorithms and decision tree learning in order to extract discriminatory features for recognizing complex visual concepts is described in [11]. In [243], interest point detection is posed as an optimization problem. They use a Genetic Programming based learning approach to construct operators for extracting features. The problem of learning an interest point operator was posed differently in [103] where human eye movement was studied to find the points of fixation and to train an SVM classifier.

One can easily generalize the feature detection problem to a classification problem and train a recognition system on image examples provided by one or a combination of the classical detectors. Any machine learning approach can be used for that. Haar like filters implemented with integral images to efficiently approximate multiscale derivatives were used in [15]. A natural extension would be to use the learning scheme from Viola and Jones [252] successfully applied to face detection, to efficiently classify interest points.

The accuracy of machine learning based methods in terms of localization, scale, and shape estimation is in general lower than for the generic detectors [84, 134, 143] but in the context of object recognition the efficiency is usually more beneficial.

2.10 Evaluations

Given the multitude of interest point approaches the need for independent performance evaluations was identified early on and many experimental tests have been performed over the last three decades. Various experimental frameworks and criteria were used. One of the first comparisons of corner detection techniques based on chain coded curves was presented in [205]. In the early papers very often only visual inspection was done [104]. Others performed more quantitative evaluations providing scores for individual images or for small test data [57, 266].

Corner detectors were often tested on artificially generated images with different types of junctions with varying angle, length, contrast, noise, blur etc. [41, 179]. Different affine photometric and geometric transformations were used to generate the test data and to evaluate corner detectors in [37, 100, 124, 128]. This approach simplifies the evaluation process but cannot model all the noise and deformations which affect the detector performance in a real application scenario, thus the performance results are often over-optimistic. A somewhat different approach is taken in [150]. There, performance comparison is approached as a general recognition problem. Corners are manually annotated on affine transformed images and measures like consistency and accuracy similar to detection rate and recall are used to evaluate the detectors.

In [88], sets of points are extracted from polyhedral objects and projective invariants are used to calculate a manifold of constraints on the coordinates of the corners. They estimate the variance of the distance from the point coordinates to this manifold independently of camera parameters and object pose. Nonlinear diffusion was used to remove the noise and the method from [188] performed better than the one proposed in [104]. The idea of using planar invariants is also explored in [40] to evaluate corner detectors based on edges. Theoretical properties of features and localization accuracy were also tested in [54, 88, 188, 189, 190] based on a parametric L-corner model to evaluate localization accuracy. Also a randomized generator of corners has been used to test the localization error [261].

State-of-the-art curve based detectors [3, 74, 193, 197, 207] are evaluated in [238]. A quantitative measure of the quality of the detected dominant points is defined as the pointwise error between the digital curve and the polygon approximated from interest points. The performance of the proposed scale adapted approach is reported better than of the other methods.

The repeatability rate and information content measures were introduced in [83]. They consider a point in an image interesting if it has two main properties: distinctiveness and invariance. This means that a point should be distinguishable from its immediate neighbors. Moreover, the position as well as the selection of the interest point should be invariant with respect to the expected geometric and radiometric distortions. From a set of investigated detectors [57, 84, 104, 232, 266], Harris [84] and a corner later described as SUSAN [232] perform best.

Systematic evaluation of several interest point detectors based on repeatability and information content measured by the entropy of the descriptors was performed in [215]. The evaluation shows that a modified Harris detector provides the most stable results on image pairs with different geometric transformations. The repeatability rate and information content in the context of image retrieval were also evaluated in [218] to show that a wavelet-based salient point extraction algorithm outperforms the Harris detector [84].

Consistency of the number of corners and accuracy criteria were introduced as evaluation criteria in [43]. This overcomes the problems with the repeatability criterion of favoring detectors providing more features. The introduced criterion instead favors detectors which provide similar number of points regardless of the object transformation even though the number of details in the image changes with scale and resolution. Several detectors [84, 104, 152, 232] are compared with the best performance reported for a modified implementation of [152].

Tracking and the number of frames over which the corners are detected during tracking was used to compare detectors in [239, 240]. Similarly Bae et al. [8] uses correlation and matching to find repeated corners between frames and compare their numbers to the reference frame.

Extensive evaluation of commonly used feature detectors and descriptors has been performed in [144, 145]. The repeatability on image pairs representing planar scenes related by various geometric transformations was computed for different state-of-the-art scale and affine invariant detectors. The MSER region detector [134] based on watershed segmentation showed the highest accuracy and stability on various structured scenes. The data collected by Mikolajczyk and Tuytelaars [145] became a standard benchmark for evaluating interest point detectors and descriptors.[2]

Recently, the performance of feature detectors and descriptors from [144, 145] has been investigated in [71, 157, 158] in the context of matching 3D object features across viewpoints and lighting conditions. A method based on intersecting epipolar constraints provides ground truth correspondences automatically. In this evaluation, the affine invariant detectors introduced in [143] are most robust to viewpoint changes. DoG detector from [124] was reported the best in a similar evaluation based on images of natural 3D scenes in [256].

Feature detectors were also evaluated in the context of recognition in [99, 139, 235] using object category training data where direct correspondence cannot be automatically verified. Clustering properties and compactness of feature clusters were measured in [139]. Some specific recognition tasks like pedestrian detection were also used to compare the performance of different features in [220].

[2] see http://www.robots.ox.ac.uk/~vgg/research/affine.

3

Corner Detectors

A large number of corner detector methods have been proposed in the literature. To guide the reader in finding an approach suitable for a given application, representative methods have been selected based on the underlying extraction technique (e.g., based on image derivatives, morphology, or geometry), as well as based on the level of invariance (translations and rotations, scale or affine invariant). For each category, we describe the feature extraction process for some of the best performing and representative methods.

3.1 Introduction

It is important to note that the term *corner* as used here has a specific meaning. The detected points correspond to points in the 2D image with high curvature. These do not necessarily correspond to projections of 3D corners. Corners are found at various types of junctions, on highly textured surfaces, at occlusion boundaries, etc. For many practical applications, this is sufficient, since the goal is to have a set of stable and repeatable features. Whether these are true corners or not is considered irrelevant.

We begin this section with a derivatives-based approach, the Harris corner detector, described in Section 3.2. Next, we explain the basic ideas of the SUSAN detector (Section 3.3), which is an example of a method based on efficient morphological operators. We then move on to detectors with higher levels of invariance, starting with the scale and affine invariant extensions of the Harris detector: Harris-Laplace and Harris-Affine (Section 3.4). This is followed by a discussion of Edge-Based Regions in Section 3.5. Finally, we conclude the section with a short discussion (Section 3.6).

3.2 Harris Detector

The Harris detector, proposed by Harris and Stephens [84], is based on the *second moment matrix*, also called the auto-correlation matrix, which is often used for feature detection and for describing local image structures. This matrix describes the gradient distribution in a local neighborhood of a point:

$$M = \sigma_D^2\, g(\sigma_I) * \begin{bmatrix} I_x^2(\mathbf{x},\sigma_D) & I_x(\mathbf{x},\sigma_D)I_y(\mathbf{x},\sigma_D) \\ I_x(\mathbf{x},\sigma_D)I_y(\mathbf{x},\sigma_D) & I_y^2(\mathbf{x},\sigma_D) \end{bmatrix} \tag{3.1}$$

with

$$I_x(\mathbf{x},\sigma_D) = \frac{\partial}{\partial x} g(\sigma_D) * I(\mathbf{x}) \tag{3.2}$$

$$g(\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.3}$$

The local image derivatives are computed with Gaussian kernels of scale σ_D (the *differentiation scale*). The derivatives are then averaged in the neighborhood of the point by smoothing with a Gaussian window of scale σ_I (the *integration scale*). The eigenvalues of this matrix represent the principal signal changes in two orthogonal directions in a neighborhood around the point defined by σ_I. Based on this property, corners can be found as locations in the image for which the image signal varies significantly in both directions, or in other words, for which both eigenvalues are large. In practice, Harris proposed to use the following measure for *cornerness*, which combines the two eigenvalues in a single measure and is computationally less expensive:

$$\text{cornerness} = \det(M) - \lambda\ \text{trace}(M) \tag{3.4}$$

with $\det(M)$ the determinant and $\text{trace}(M)$ the trace of the matrix M. A typical value for λ is 0.04. Since the determinant of a matrix is equal to the product of its eigenvalues and the trace corresponds to the sum, it is clear that high values of the cornerness measure correspond to both eigenvalues being large. Adding the second term with the trace reduces the response of the operator on strong straight contours. Moreover, computing this measure based on the determinant and the trace is computationally less demanding than actually computing the eigenvalues. This seems less relevant now, but it was important back in 1988 when the computational resources were still very limited.

Subsequent stages of the corner extraction process are illustrated in Figure 3.1. Given the original image $I(x,y)$ (upper left), the first step consists of computing the first-order derivatives I_x and I_y (lower left). Next, one takes the product of these gradient images (lower right). Then, the images are smoothed with a Gaussian kernel. These

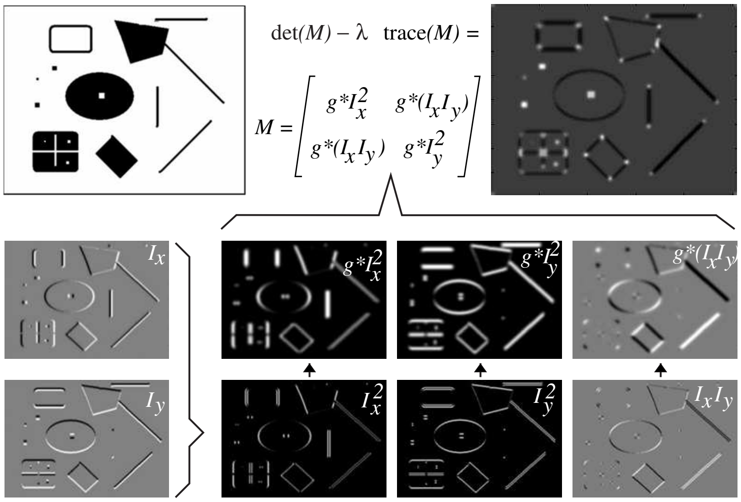

Fig. 3.1 Illustration of the components of the second moment matrix and Harris cornerness measure.

images contain the different elements of the Hessian matrix, which are then in a final step combined into the cornerness measure, following Equation (3.4) (upper right).

When used as an interest point detector, local maxima of the cornerness function are extracted, using non-maximum suppression. Such points are translation and rotation invariant. Moreover, they are stable under varying lighting conditions. In a comparative study of different interest point detectors [214, 215], the Harris corner was proven to be the most repeatable and most informative. Additionally, they can be made very precise: sub-pixel precision can be achieved through quadratic approximation of the cornerness function in the neighborhood of a local maximum.

3.2.1 Discussion

Figure 3.2 shows the corners detected with this measure for two example images related by a rotation. Note that the features found correspond to locations in the image showing two dimensional variations in the intensity pattern. These may correspond to real "corners", but the detector also fires on other structures, such as T-junctions, points with high curvature, etc. This equally holds for all other corner detectors described in this chapter. When true corners are desirable, model-based approaches are certainly more appropriate.

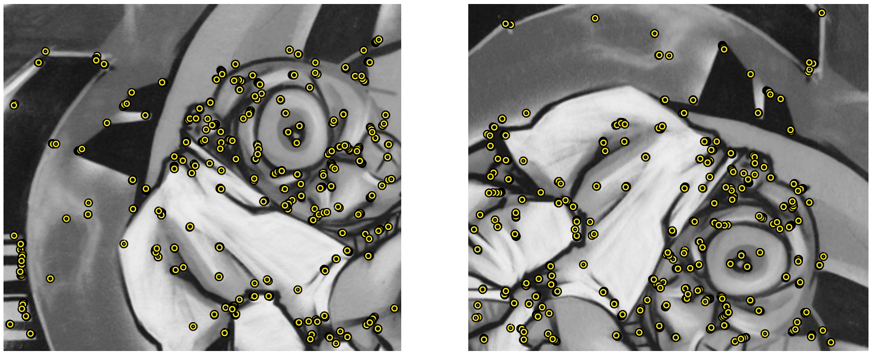

Fig. 3.2 Harris corners detected on rotated image examples.

As can be seen in the figure, many but not all of the features detected in the original image (left) have also been found in the rotated version (right). In other words, the *repeatability* of the Harris detector under rotations is high.

Additionally, features are typically found at locations which are *informative*, i.e., with a high variability in the intensity pattern. This makes them more discriminative and easier to bring into correspondence.

3.3 SUSAN Detector

The SUSAN corner detector has been introduced by Smith and Brady [232] and relies on a different technique. Rather than evaluating local gradients, which might be noise-sensitive and computationally more expensive, a morphological approach is used.

SUSAN stands for *Smallest Univalue Segment Assimilating Nucleus*, and is a generic low-level image processing technique, which apart from corner detection has also been used for edge detection and noise suppression. The basic principle goes as follows (see also Figure 3.3). For each pixel in the image, we consider a circular neighborhood of fixed radius around it. The center pixel is referred to as the *nucleus*, and its intensity value is used as reference. Then, all other

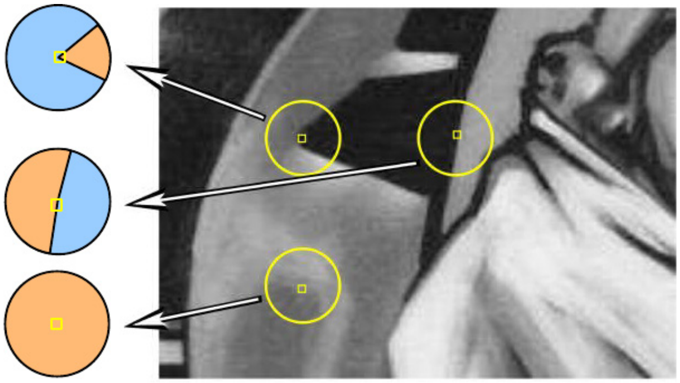

Fig. 3.3 SUSAN corners are detected by segmenting a circular neighborhood into "similar" (orange) and "dissimilar" (blue) regions. Corners are located where the relative area of the "similar" region (USAN) reaches a local minimum below a certain threshold.

pixels within this circular neighborhood are partitioned into two categories, depending on whether they have “similar” intensity values as the nucleus or “different” intensity values. In this way, each image point has associated with it a local area of similar brightness (coined USAN), whose relative size contains important information about the structure of the image at that point (see also Figure 3.3). In more or less homogeneous parts of the image, the local area of similar brightness covers almost the entire circular neighborhood. Near edges, this ratio drops to 50%, and near corners it decreases further to about 25%. Hence, corners can be detected as locations in the image where the number of pixels with similar intensity value in a local neighborhood reaches a local minimum and is below a predefined threshold. To make the method more robust, pixels closer in value to the nucleus receive a higher weighting. Moreover, a set of rules is used to suppress qualitatively “bad” features. Local minima of the SUSANs (Smallest USANs) are then selected from the remaining candidates. An example of detected SUSAN corners is shown in Figure 3.4.

3.3.1 Discussion

The features found show a high repeatability for this (artificially rotated) set of images. However, many of the features are located on edge structures and not on corners. For such points, the localization

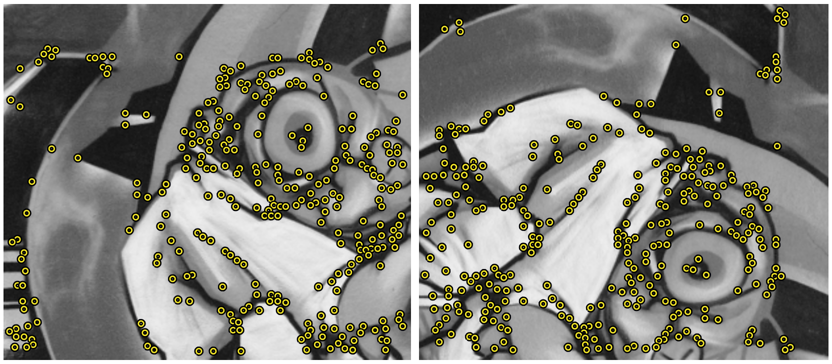

Fig. 3.4 SUSAN corners found for our example images.

is sensitive to noise. Moreover, edge based points are also less discriminative.

The two detectors described so far are invariant under translation and rotation only. This means that corners will be detected at corresponding locations only if the images are related by a translation and/or rotation. In the next sections, we will describe detectors with higher levels of viewpoint invariance, that can withstand scale changes or even affine deformations. Apart from better matching across transformed images, these also bring the advantage of detecting features over a range of scales or shapes.

Alternatively, this effect can be obtained by using a *multiscale approach*. In that case, a detector which is not scale invariant is applied to the input image at different scales (i.e., after smoothing and sampling).

3.4 Harris-Laplace/Affine

Mikolajczyk and Schmid developed both a scale invariant corner detector, referred to as *Harris-Laplace*, as well as an affine-invariant one, referred to as *Harris-Affine* [143].

3.4.1 Harris-Laplace

Harris-Laplace starts with a multiscale Harris corner detector as initialization to determine the location of the local features. The characteristic scale is then determined based on scale selection as proposed by Lindeberg et al. [116, 120]. The idea is to select the *characteristic* scale of a local structure, for which a given function attains an extremum over scales (see Figure 3.5). The selected scale is characteristic in the quantitative sense, since it measures the scale at which there is maximum similarity between the feature detection operator and the local image structures. The size of the region is therefore selected independently of the image resolution for each point. As the name Harris-Laplace suggests, the Laplacian operator is used for scale selection. This has been shown to give the best results in the experimental comparison of [141] as well as in [38]. These results can be explained by the circular shape

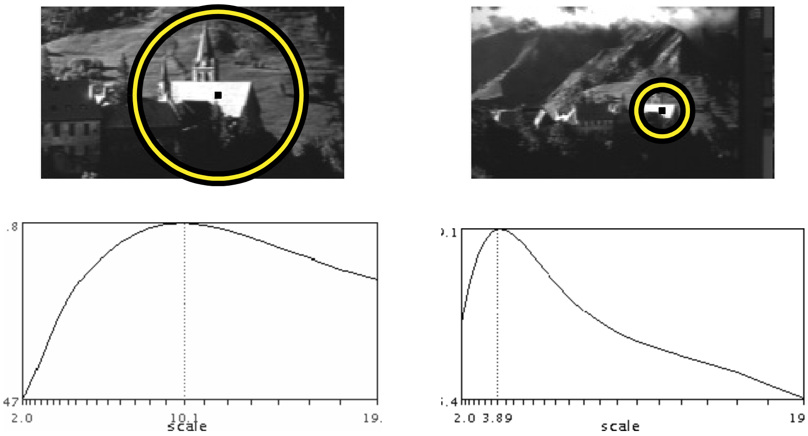

Fig. 3.5 Example of characteristic scales. The top row shows images taken with different zoom. The bottom row shows the responses of the Laplacian over scales for two corresponding points. The characteristic scales are 10.1 and 3.9 for the left and right images, respectively. The ratio of scales corresponds to the scale factor (2.5) between the two images. The radius of displayed regions in the top row is equal to 3 times the selected scales.

of the Laplacian kernel, which acts as a matched filter [58] when its scale is adapted to the scale of a local image structure.

Figure 3.6 shows the scale-invariant local features obtained by applying the Harris-Laplace detector, for two images of the same scene related by a scale change. In order not to overload the images, only some of the corresponding regions that were detected in both images

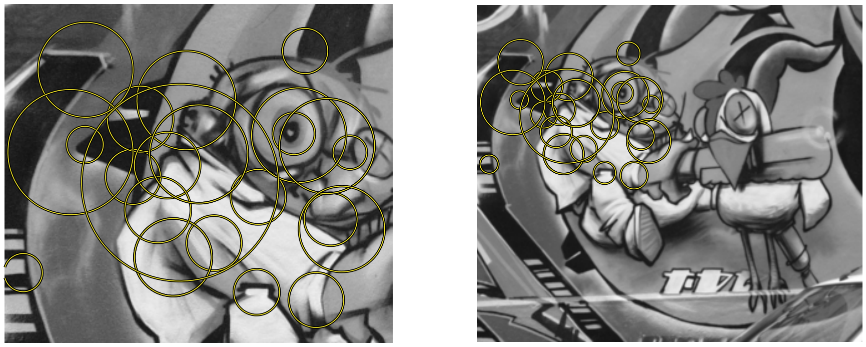

Fig. 3.6 Corresponding features found with the Harris-Laplace detector. Only a subset of corresponding features is displayed to avoid clutter. The circles indicate the scale of the features.

are shown. A similar selection mechanism has been used for all subsequent image pairs shown in this survey.

3.4.2 Harris-Affine

Given a set of initial points extracted at their characteristic scales based on the Harris-Laplace detection scheme, the iterative estimation of elliptical affine regions as proposed by Lindeberg et al. [118, 121] allows to obtain affine invariant corners. Instead of circular regions, these are ellipses. The procedure consists of the following steps:

(1) Detect the initial region with the Harris-Laplace detector.
(2) Estimate the affine shape with the second moment matrix.
(3) Normalize the affine region to a circular one.
(4) Re-detect the new location and scale in the normalized image.
(5) Go to step 2 if the eigenvalues of the second moment matrix for the new point are not equal.

The iterations are illustrated in Figure 3.7.

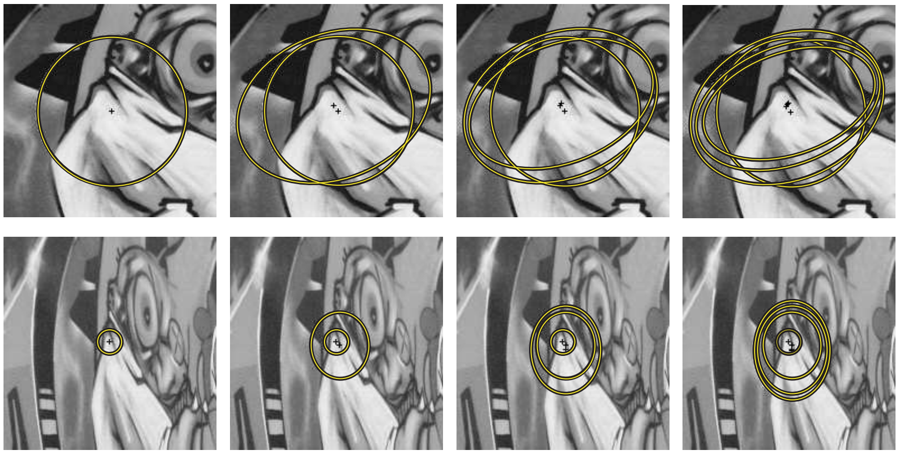

Fig. 3.7 Iterative detection of an affine invariant interest point in the presence of an affine transformation (top and bottom rows). The first column shows the points used for initialization. The consecutive columns show the points and regions after iterations 1, 2, 3, and 4. Note that the regions converge after 4 iterations to corresponding image regions.

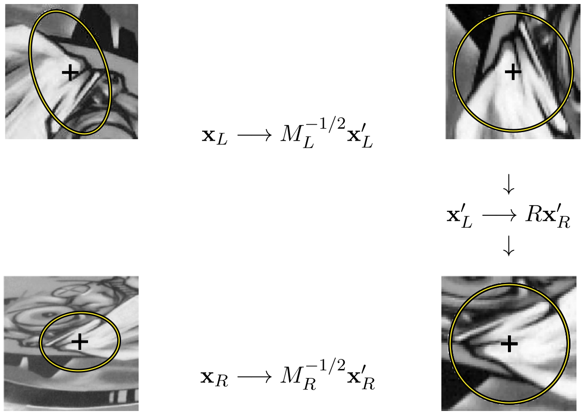
$\mathbf{x}_L \longrightarrow M_L^{-1/2}\mathbf{x}'_L$
$\downarrow$
$\mathbf{x}'_L \longrightarrow R\mathbf{x}'_R$
$\downarrow$
$\mathbf{x}_R \longrightarrow M_R^{-1/2}\mathbf{x}'_R$

Fig. 3.8 Diagram illustrating the affine normalization using the second moment matrices. Image coordinates are transformed with matrices $M_L^{-1/2}$ and $M_R^{-1/2}$.

The eigenvalues of the second moment matrix (see Equation (3.3)) are used to measure the *affine shape* of the point neighborhood. More precisely, we determine the transformation that projects the intensity pattern of the point neighborhood to one with equal eigenvalues. This transformation is given by the square root of the second moment matrix $M^{1/2}$. It can be shown that if the neighborhoods of two points $\mathbf{x}_R$ and $\mathbf{x}_L$ are related by an affine transformation, then their normalized versions, $\mathbf{x}_R = M_R^{-1/2}\mathbf{x}'_R$ and $\mathbf{x}_L = M_L^{-1/2}\mathbf{x}'_L$, are related by a simple rotation $\mathbf{x}'_L = R\mathbf{x}'_R$ [13, 121]. This process is illustrated in Figure 3.8. The matrices M'_L and M'_R computed in the normalized frames are rotation matrices as well. Note that rotation preserves the eigenvalue ratio for an image patch, therefore, the affine deformation can be determined only up to a rotation factor.

The estimation of affine shape can be applied to any initial point given that the determinant of the second moment matrix is larger than zero and the signal to noise ratio is sufficiently large. We can therefore use this technique to estimate the shape of initial regions provided by the Harris-Laplace detector.

The output of the Harris-Affine detector on two images of the same scene is shown in Figure 3.9. Apart from the scale, also the shape of the regions is now adapted to the underlying intensity patterns, so as

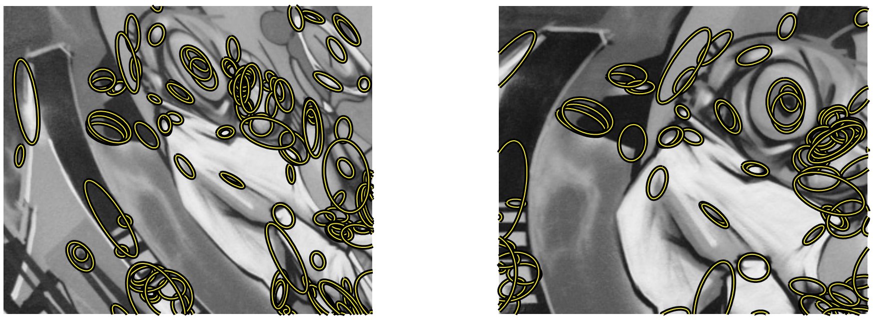

Fig. 3.9 Harris-Affine regions generated for two different views of a planar scene (subset). In spite of the affine deformation, the region shapes clearly correspond.

to ensure that the same part of the object surface is covered in spite of the deformations caused by the viewpoint change.

3.5 Edge-based Regions

A more heuristic technique to obtain affine invariance is to exploit the geometry of the edges that can usually be found in the proximity of a Harris corner. Such a method has been proposed by Tuytelaars and Van Gool [247, 249]. The rationale behind this approach is that edges are typically rather stable image features, that can be detected over a range of viewpoints, scales and illumination changes. Moreover, by exploiting the edge geometry, the dimensionality of the problem can be significantly reduced. Indeed, as will be shown next, the 6D search problem over all possible affinities (or 4D, once the center point is fixed) can be reduced to a one-dimensional problem by exploiting the nearby edges geometry. In practice, we start from a Harris corner point $\mathbf{p}$ (see Section 3.2) [84] and a nearby edge, extracted with the Canny edge detector [29]. To increase the robustness to scale changes, these basic features are extracted at multiple scales. Two points $\mathbf{p}_1$ and $\mathbf{p}_2$ move away from the corner in both directions along the edge, as shown in Figure 3.10. Their relative speed is coupled through the equality of relative affine invariant parameters l_1 and l_2:

$$l_i = \int \mathrm{abs}\left(|\mathbf{p_i}^{(1)}(s_i)\mathbf{p} - \mathbf{p_i}(s_i)|\right) ds_i \tag{3.5}$$

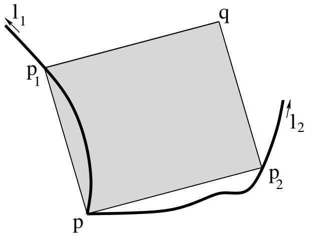

Fig. 3.10 The edge-based region detector starts from a corner point $\mathbf{p}$ and exploits nearby edge information.

with s_i an arbitrary curve parameter (in both directions, $i = 1,2$), $\mathbf{p_i}^{(1)}(s_i)$ the first derivative of $\mathbf{p_i}(s_i)$ with respect to s_i, abs( ) the absolute value and $|\cdots|$ the determinant. This condition prescribes that the areas between the joint $\langle \mathbf{p}, \mathbf{p_1} \rangle$ and the edge and between the joint $\langle \mathbf{p}, \mathbf{p_2} \rangle$ and the edge remain identical. From now on, we simply use l when referring to $l_1 = l_2$.

For each value l, the two points $\mathbf{p_1}(l)$ and $\mathbf{p_2}(l)$ together with the corner $\mathbf{p}$ define a parallelogram $\Omega(l)$: the parallelogram spanned by the vectors $\mathbf{p_1}(l) - \mathbf{p}$ and $\mathbf{p_2}(l) - \mathbf{p}$ (see Figure 3.10). This yields a one-dimensional family of parallelogram-shaped regions as a function of l. From this 1D family one (or a few) parallelogram(s) are selected for which the following photometric quantities of the texture go through an extremum.

$$\begin{aligned}
\mathrm{Inv}_1 &= \mathrm{abs}\left(\frac{|\mathbf{p_1} - \mathbf{p_g} \quad \mathbf{p_2} - \mathbf{p_g}|}{|\mathbf{p} - \mathbf{p_1} \quad \mathbf{p} - \mathbf{p_2}|}\right) \frac{M^1_{00}}{\sqrt{M^2_{00} M^0_{00} - (M^1_{00})2}} \\
\mathrm{Inv}_2 &= \mathrm{abs}\left(\frac{|\mathbf{p} - \mathbf{p_g} \quad \mathbf{q} - \mathbf{p_g}|}{|\mathbf{p} - \mathbf{p_1} \quad \mathbf{p} - \mathbf{p_2}|}\right) \frac{M^1_{00}}{\sqrt{M^2_{00} M^0_{00} - (M^1_{00})2}}
\end{aligned} \tag{3.6}$$

with

$$\begin{aligned}
M^n_{pq} &= \int_\Omega I^n(x,y) x^p y^q \;\; dxdy \\
\mathbf{p_g} &= \left(\frac{M^1_{10}}{M^1_{00}}, \frac{M^1_{01}}{M^1_{00}}\right)
\end{aligned} \tag{3.7}$$

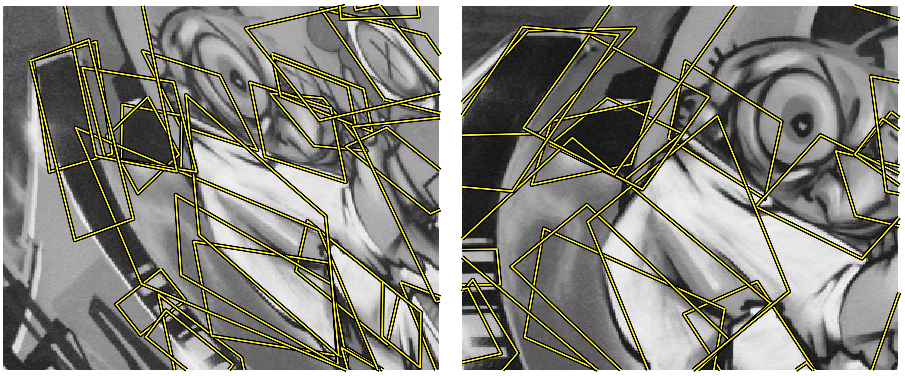

Fig. 3.11 Originally detected region shapes for the edge-based regions (subset).

with M^n_{pq} the nth *order*, $(p+q)$th *degree* moment computed over the region $\Omega(l)$, $\mathbf{p_g}$ the center of gravity of the region, weighted with intensity $I(x,y)$, and $\mathbf{q}$ the corner of the parallelogram opposite to the corner point $\mathbf{p}$ (see Figure 3.10). The second factor in these formula has been added to ensure invariance under an intensity offset.

For straight edges, $l = 0$ along the entire edge. In that case, the two photometric quantities given in Equation (3.7) are combined and locations where *both* functions reach a minimum value are taken to fix the parameters s_1 and s_2. Moreover, instead of relying on the Harris corner detection, the straight lines intersection point can be used instead. Examples of detected regions are displayed in Figure 3.11.

3.5.1 From Parallelograms to Ellipses

Note that the regions found with this method are parallelograms. This is in contrast to many other affine invariant detectors (for example those based on the second moment matrix) for which the output shape is an ellipse. For uniformity and convenience in comparison, it is sometimes advantageous to convert these parallelogram-shaped regions into ellipses. This can be achieved by selecting an ellipse with the same first- and second-order moments as the originally detected region, which is an affine covariant construction method. The elliptical regions generated with this procedure are shown in Figure 3.12. Note though that some

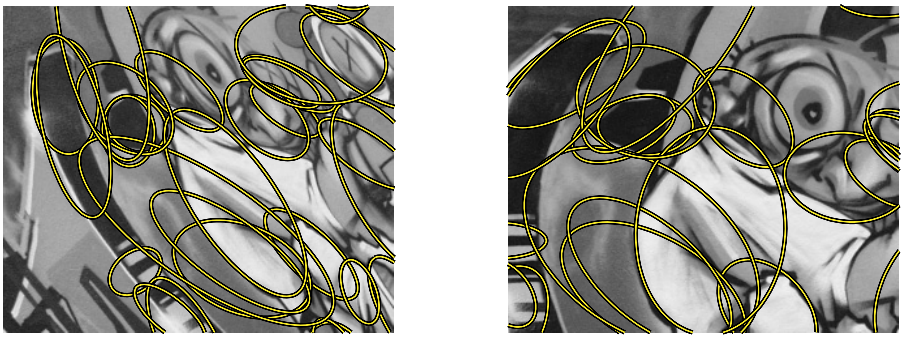

Fig. 3.12 Edge-based regions generated for the two example images, represented with ellipses (subset).

information is lost during this conversion, as ellipses have a rotational degree of freedom, which was fixed in the original representation.

3.6 Discussion

Several methods for corner detection have been described in this chapter. As discussed earlier, corner based features do not necessarily correspond to real corners in the 3D world. Indeed, the goal is to extract stable features, that can be matched well in spite of changes in viewing conditions.

The Harris detector was identified as the most stable one in many independent evaluations [83, 138, 215]. There are also multi-scale as well as scale and affine invariant extensions of this approach. It is a convenient tool for providing a large number of features. Alternatively, the SUSAN detector can be used. It is more efficient but also more sensitive to noise. An optimized SUSAN detector using machine learning techniques is described in Section 6.3. As discussed in Sections 2.2 and 2.3 contour based corner detectors are suitable for line drawing images but in natural scenes intensity based methods are typically more stable.

It is important to note that the affine transformation model only holds for viewpoint changes in case of locally planar regions and assuming the camera is relatively far from the object. However, corners are often found near object boundaries as this is where the intensity change usually occurs. Hence, the region extraction process is often based on measurements on non-planar structures, e.g., including background or another facet of the object. In these cases, the viewpoint invariance will be limited and also the robustness to background changes will be affected. A possible way out has been indicated in the work of [251]. Detectors that search for region boundaries like EBR are less affected by this phenomenon. The measurement regions can then be delimited by the detected contours, thus excluding the non-planar parts in many practical situations.

On the positive side, compared to other types of features, corners are typically better localized in the image plane. This localization accuracy can be important for some applications, e.g., for camera calibration or 3D reconstruction. Their scale however is not well defined as a corner structure changes very little over a wide range of scales. The reason why scale selection still works with the Harris detector is that the feature point is localized not exactly on the corner edge but slightly inside the corner structure.

4

Blob Detectors

After corners, the second most intuitive local features are blobs. As it was the case in the previous section, we select a few methods that have proved successful in many applications and describe these in more detail. These methods typically provide complementary features to the ones discussed in the previous chapter. We start with a derivative-based method: the Hessian detector (Section 4.1). Next, we consider the scale-invariant and affine invariant extensions of this method, coined Hessian-Laplace and Hessian-Affine (Section 4.2). Finally, we describe the salient region detector, which is based on the entropy of the intensity probability distribution (Section 4.3). We conclude the chapter with a short discussion.

4.1 Hessian Detector

The second 2 × 2 matrix issued from the Taylor expansion of the image intensity function $I(\mathbf{x})$ is the *Hessian* matrix:

$$H = \begin{bmatrix} I_{xx}(\mathbf{x}, \sigma_D) & I_{xy}(\mathbf{x}, \sigma_D) \\ I_{xy}(\mathbf{x}, \sigma_D) & I_{yy}(\mathbf{x}, \sigma_D) \end{bmatrix} \tag{4.1}$$

with I_{xx} etc. second-order Gaussian smoothed image derivatives. These encode the shape information by describing how the normal to an isosurface changes. As such, they capture important properties of local image structure. Particularly interesting are the filters based on the determinant and the trace of this matrix. The latter is often referred to as the *Laplacian*. Local maxima of both measures can be used to detect blob-like structures in an image [16].

The Laplacian is a separable linear filter and can be approximated efficiently with a Difference of Gaussians (DoG) filter. The Laplacian filters have one major drawback in the context of blob extraction though. Local maxima are often found near contours or straight edges, where the signal change is only in one direction [138]. These maxima are less stable because their localization is more sensitive to noise or small changes in neighboring texture. This is mostly an issue in the context of finding correspondences for recovering image transformations. A more sophisticated approach, solving this problem, is to select a location and scale for which the trace *and* the determinant of the Hessian matrix simultaneously assume a local extremum.

This gives rise to points, for which the second order derivatives detect signal changes in two orthogonal directions. A similar idea is explored in the Harris detector, albeit for first-order derivatives only.

The feature detection process based on the Hessian matrix is illustrated in Figure 4.1. Given the original image (upper left), one first computes the second-order Gaussian smoothed image derivatives (lower part), which are then combined into the determinant of the Hessian (upper right).

The interest points detected with the determinant of the Hessian for an example image pair are displayed in Figure 4.2. The second-order derivatives are symmetric filters, thus they give weak responses exactly in the point where the signal change is most significant. Therefore, the maxima are localized at ridges and blobs for which the size of the Gaussian kernel σ_D matches by the size of the blob structure.

4.2 Hessian-Laplace/Affine

The Hessian-Laplace and Hessian-Affine detectors are similar in spirit as their Harris-based counterparts Harris-Laplace and Harris-Affine,

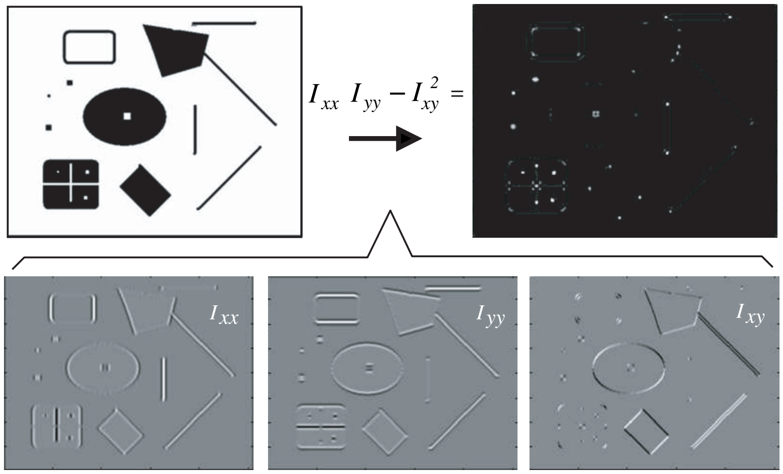

Fig. 4.1 Illustration of the components of the Hessian matrix and Hessian determinant.

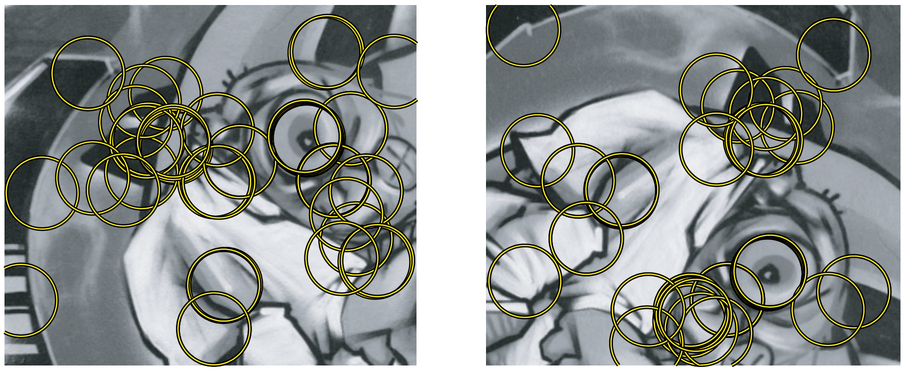

Fig. 4.2 Output of the Hessian detector applied at a given scale to example images with rotation (subset).

described in Section 3.4, except that they start from the determinant of the Hessian rather than the Harris corners. This turns the methods into viewpoint invariant blob-detectors. They have also been proposed by Mikolajczyk and Schmid [143], and are complementary to their Harris-based counterparts, in the sense that they respond to a different type of feature in the image. An example of the detection result is shown in Figures 4.3 and 4.4 for the scale-invariant Hessian-Laplace and affine invariant Hessian-Affine, respectively.

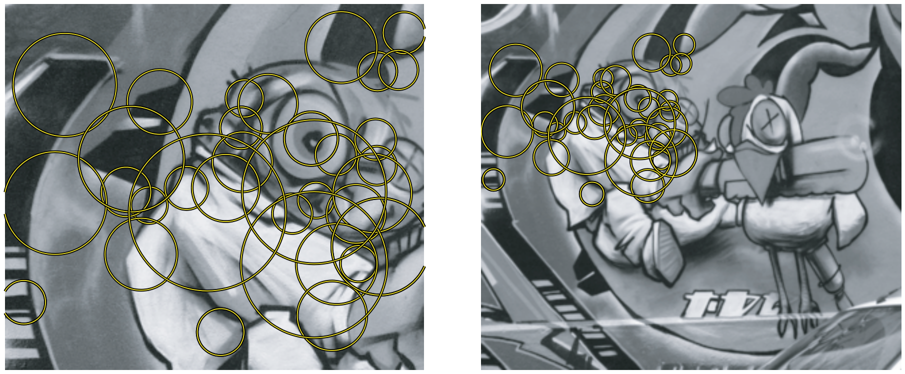

Fig. 4.3 Output of Hessian-Laplace detector applied to example images with scale change (subset).

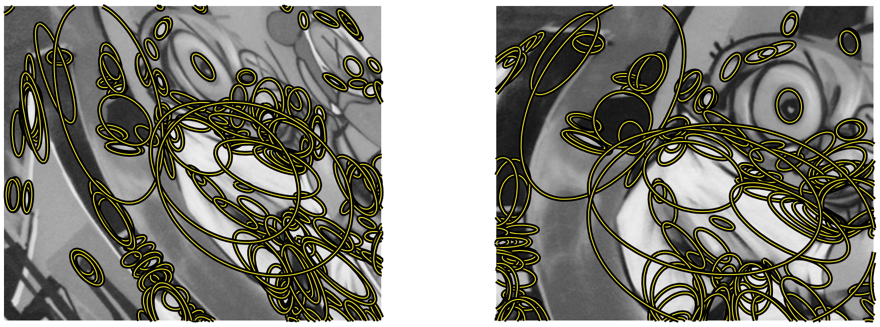

Fig. 4.4 Hessian-Affine regions generated for two views of the example scene (subset).

Like in the Harris based detector the number of regions found with the Hessian-Laplace detector can be controlled by thresholding the Hessian determinant as well as the Laplacian response. Typically a large number of features can be extracted resulting in a good coverage of the image, which is one of the advantages of the Hessian-based detector.

Furthermore, this detector also responds to some corner structures at fine scale (see Figure 4.1). The returned locations, however, are more suitable for scale estimation than the Harris points due to the use of similar filters for spatial and scale localization, both based on second-order Gaussian derivatives. One of the possible extensions of this work is to explore the Hessian matrix to use additional shape information encoded by the eigenvalues of this matrix.

4.3 Salient Regions

Rather than building on the derivative information in the image, the salient region detector proposed by Kadir and Brady [98] is inspired by information theory. The basic idea behind this feature detector is to look for *salient* features, where saliency is defined as local complexity or unpredictability. It is measured by the entropy of the probability distribution function of intensity values within a local image region. However, looking at entropy alone does not suffice to accurately localize the features over scales, so as an additional criterion, the *self-dissimilarity* in scale-space of the feature is added as an extra weighting function, favoring well-localized complex features.

Detection proceeds in two steps: first, at each pixel $\mathbf{x}$ the entropy $\mathcal{H}$ of the probability distribution $p(I)$ is evaluated over a range of scales s.

$$\mathcal{H} = -\sum_{I} p(I) \log p(I).$$

The probability distribution $p(I)$ is estimated empirically based on the intensity distribution in a circular neighbourhood of radius s around $\mathbf{x}$. Local maxima of the entropy are recorded. These are candidate salient regions. Second, for each of the candidate salient regions the magnitude of the derivative of $p(I)$ with respect to scale s is computed as

$$\mathcal{W} = \frac{s^2}{2s-1} \sum_{I} \left| \frac{\partial p(I;s)}{\partial s} \right|.$$

The saliency $\mathcal{Y}$ is then computed as

$$\mathcal{Y} = \mathcal{W}\mathcal{H}.$$

The candidate salient regions over the entire image are ranked by their saliency $\mathcal{Y}$, and the top P ranked regions are retained.

Also an affine invariant version of the detector has been proposed, where local maxima over the scale s and the shape parameters (orientation θ and ratio of major to minor axes λ of an elliptical region)

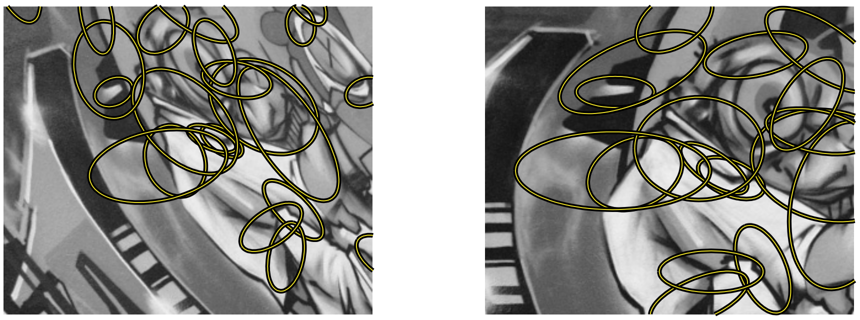

Fig. 4.5 Salient regions found for the two example images related to a change in viewpoint (subset).

are sought simultaneously. However, this seriously slows down the computation.

Examples of detected regions, using the affine invariant version, are displayed in Figure 4.5. More details about this method can be found in [99].

4.3.1 Discussion

Because of the weighting factor measuring the self-dissimilarity over scale, the detector typically fires on blob-like structures in the image. That is why we have catalogued the method as a blob detector. But note that in contrast to other blob detectors the contrast of the blobs does not have any influence on the detection.

The number of features found with this method is typically relatively low. Unlike for many other detectors the ranking of the extracted features is meaningful due to the entropy based criteria with the ones from the top the most stable. This property has been explored in the context of category-level object recognition, and especially in combination with classifiers where the complexity largely depends on the number of features (e.g., [65]).

4.4 Discussion

Blob detectors have been used widely in different application domains. Apart from the methods described above, also DoG (difference-of-

Gaussians) [124] and SURF (speeded-up robust features) [15] can be catalogued as blob detectors. However, since their extraction processes are focussed on efficiency, we postpone their discussion until Section 6.

Some of the methods described in Section 5 also share common characteristics with blob detectors. Especially IBR (intensity-based regions) [248] and MSER (maximally stable extremal regions) [134] often find blob-like structures in the image. However, apart from blob-like structures, they also detect other, more irregularly shaped patterns, which we consider their distinctive property.

Blob detectors are in a sense complementary to corner detectors. As a result, they are often used together. By using several complementary feature detectors, the image is better covered and the performance becomes less dependent on the actual image content. This has been exploited, e.g., in [110, 140, 231].

In general, blob-like structures tend to be less accurately localized in the image plane than corners, although their scale and shape are better defined than for corners. The location of a corner can be identified by a single point while blobs can only be localized by their boundaries, which are often irregular. On the other hand, the scale estimation of a corner is ill-defined as for example an intersection of edges exists at a wide range of scales. The boundaries of a blob however, even if irregular, give a good estimate of the size thus scale of the blob. This makes them less suited for, e.g., camera calibration or 3D reconstruction. For object recognition, on the other hand, a precise image localization is often not necessary, since the entire recognition process is very noisy. A robust descriptor such as SIFT [124] can match such features nevertheless. The scales of matched blobs allow then to hypothesize the size of the objects [140], which makes them very useful in recognition applications.

Finally, the number of features detected with the methods described above varies greatly. There is often just a few tens of salient regions found in an image, whereas the Hessian-Laplace or Hessian-Affine methods allow to extract up to several hundreds or thousands of features. Depending on the application and algorithms used, either case can be advantageous. We refer to Section 7 for a further discussion on this issue.

5

Region Detectors

In this chapter, we discuss a number of feature detectors which, directly or indirectly, are concerned with extraction of image regions. First, we describe the intensity-based regions (Section 5.1), followed by maximally stable extremal regions (Section 5.2). At the end, we discuss superpixels (Section 5.3). These regions are provided by different methods but focus on similar image structures and share similar properties. Superpixels are traditionally not considered as local features and have limited robustness to changes in viewing conditions but they are currently more and more used in the context of image recognition. We therefore include all the above features in the same category.

5.1 Intensity-based Regions

Here we describe a method proposed by Tuytelaars and Van Gool [248, 249] to detect affine invariant regions. It starts from intensity extrema (detected at multiple scales), and explores the image around them in a radial way, delineating regions of arbitrary shape, which are then replaced by ellipses.

More precisely, given a local extremum in intensity, the intensity function along rays emanating from the extremum is studied (see

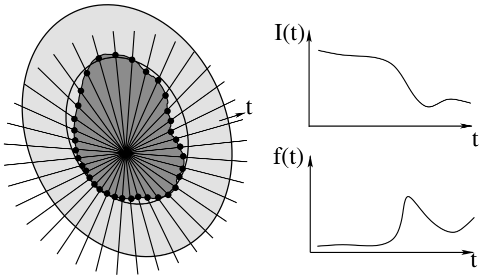

Fig. 5.1 Construction of intensity-based regions.

Figure 5.1). The following function is evaluated along each ray:

$$f(t) = \frac{\text{abs}(I(t) - I_0)}{\max\left(\frac{\int_0^t \text{abs}(I(t) - I_0) dt}{t}, d\right)}$$

with t an arbitrary parameter along the ray, $I(t)$ the intensity at position t, I_0 the intensity value at the extremum, and d a small number which has been added to prevent a division by zero. The point for which this function reaches an extremum is invariant under affine geometric and linear photometric transformations (given the ray). Typically, a maximum is reached at positions where the intensity suddenly increases or decreases. The function $f(t)$ is in itself already invariant. Nevertheless, points are selected where this function reaches an extremum to make a robust selection. Next, all points corresponding to maxima of $f(t)$ along rays originating from the same local extremum are linked to enclose an affine invariant region. This often irregularly shaped region is replaced by an ellipse having the same shape moments up to the second-order. This ellipse fitting is an affine covariant construction. An example of regions detected with this method is shown in Figure 5.2.

5.2 Maximally Stable Extremal Regions

MSER or Maximally Stable Extremal Regions have been proposed by Matas et al. [134]. A Maximally Stable Extremal Region is a connected component of an appropriately thresholded image. The word "extremal" refers to the property that all pixels inside the MSER have

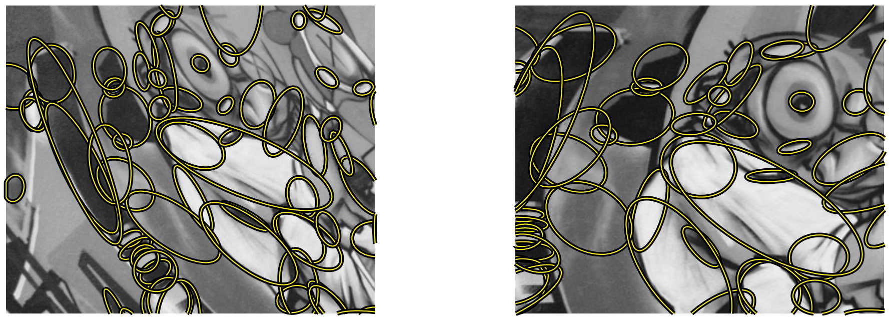

Fig. 5.2 Intensity-based regions found for the graffiti images (subset).

either higher (bright extremal regions) or lower (dark extremal regions) intensity than all the pixels on its outer boundary. The “maximally stable” in MSER describes the property optimized in the threshold selection process.

The set of extremal regions $\mathcal{E}$, i.e., the set of all connected components obtained by thresholding, has a number of desirable properties. First, a monotonic change of image intensities leaves $\mathcal{E}$ unchanged. Second, continuous geometric transformations preserve topology — pixels from a single connected component are transformed to a single connected component. Finally, there are no more extremal regions than there are pixels in the image. So a set of regions was defined that is preserved under a broad class of geometric and photometric changes and yet has the same cardinality as, e.g., the set of fixed-sized square windows commonly used in narrow-baseline matching.

The enumeration of the set of extremal regions $\mathcal{E}$ is very efficient, almost linear in the number of image pixels. The enumeration proceeds as follows. First, pixels are sorted by intensity. After sorting, pixels are marked in the image (either in decreasing or increasing order) and the list of growing and merging connected components and their areas is maintained using the union-find algorithm [219]. During the enumeration process, the area of each connected component as a function of intensity is stored. Among the extremal regions, the “maximally stable” ones are those corresponding to thresholds for which the relative area change as a function of relative change of threshold is at a local minimum. In other words, the MSER are the parts of the image where local binarization is stable over a large range of thresholds. The definition of MSER stability based on relative area change is invariant to affine transformations (both photometrically and geometrically).

Detection of MSER is related to *thresholding*, since every extremal region is a connected component of a thresholded image. However, no global or "optimal" threshold is sought, all thresholds are tested and the stability of the connected components evaluated. The output of the MSER detector is not a binarized image. For some parts of the image, multiple stable thresholds exist and a system of nested subsets is output in this case.

For many of the affine invariant detectors the output shape is an ellipse. However, for MSER it is not. Examples of the original regions detected are given in Figure 5.3. Using the same procedure as explained above for the IBR, an ellipse can be fitted based on the first and second shape moments. This results in a set of features as shown in Figure 5.4. Alternatively, a local affine frame can be defined based on a set of stable points along the region contour [135]. This provides an alternative scheme to normalize the region against affine deformations.

5.2.1 Discussion

The MSER features typically anchor on region boundaries, thus the resulting regions are accurately localized compared to other

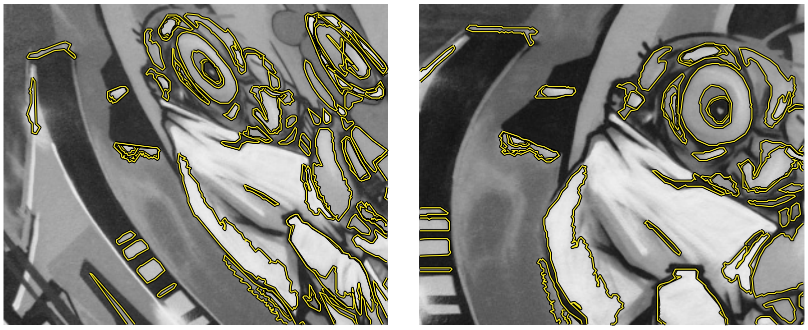

Fig. 5.3 Regions detected with MSER on the graffiti images (subset).

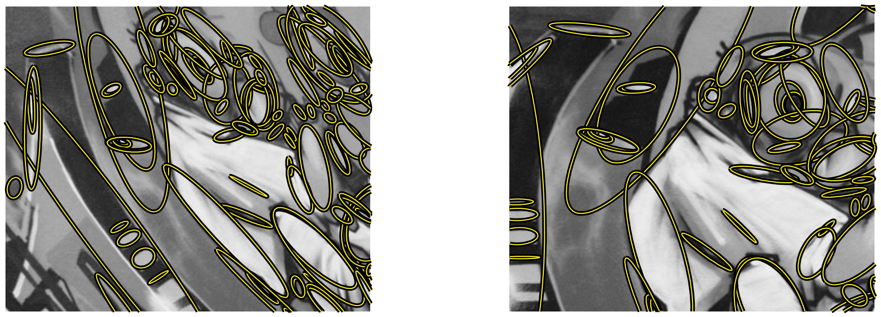

Fig. 5.4 Final MSER regions for the graffiti images (subset).

blob-detectors. The method works best for structured images which can be segmented well — ideally an image with uniform regions separated by strong intensity changes. On the downside, it was found to be sensitive to image blur [145], which can be explained by the fact that image blur undermines the stability criterion. This issue was addressed in its recent extension in [177]. The method is also relatively fast. It is currently the most efficient among the affine invariant feature detectors. It has been used mostly for recognizing or matching specific objects (e.g., [166, 231]) and showed lower performance for object class recognition [139].

5.3 Segmentation-based Methods (Superpixels)

The two methods described above extract small regions whose intensity patterns clearly stand out with respect to their immediate surroundings. This is reminiscent of traditional image segmentation techniques. However, image segments are typically relatively large — too large, in fact, to be used as local features. By increasing the number of segments a new image representation can be obtained where the image segments typically have the right trade-off between locality and distinctiveness required in most local features-based applications (see Figure 5.5). This low-level grouping of pixels into atomic regions has been advocated by Mori et al. [159] and Ren and Malik [184], who refer to the resulting atomic regions as *superpixels*. This terminology refers to the fact that

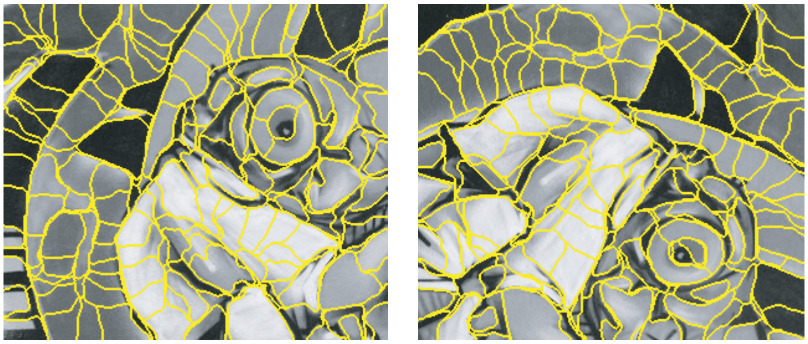

Fig. 5.5 Superpixels generated for the example image.

superpixels can be considered as a more natural and perceptually more meaningful alternative for the original image pixels.

In [159, 184], superpixels are extracted from the image using normalized cuts [227], but any data driven segmentation methods can be used here. The normalized cuts based approach is a classical image segmentation algorithm which exploits pairwise brightness, color, or texture affinities between pixels. To enforce locality, only local connections are taken into account when constructing the affinity matrix. An example of superpixels is shown in Figure 5.5.

In contrast to traditional local features, by construction superpixels cover the entire image and do not overlap. Multiple segmentations can also be used to increase the possibility of object boundaries coinciding with boundaries between adjacent superpixels, except for small contour details and invisible contours. All superpixels extracted from an image have similar scale, so the method is not scale-invariant. An alternative construction method, based on Constrained Delauney Triangulation, has been proposed to obtain robustness against scale change [183].

These features are less suited for matching or object recognition, as the regions are uniform therefore not discriminative and the repeatability of boundary extraction is low. They have been used successfully for modeling and exploiting mid-level visual cues, such as curvilinear continuity, region grouping, or figure/ground organization for semantic image segmentation.

5.4 Discussion

The local features detected with the methods described above typically represent homogeneous regions. While this is acceptable for the detection step, it may incur problems for the later description and matching. Indeed, homogeneous regions lack distinctiveness. Fortunately, this can easily be overcome by increasing the measurement region. In other words, we use a larger scale region to compute the descriptor, such that it also contains part of the surrounding image structures and captures the shape of the region boundary. This usually suffices to increase the discriminative power and match regions between images.

Intensity-based regions and maximally stable extremal regions typically give very similar features. These methods are therefore not complementary. IBR may break down when the region is non-convex, but it is more robust to small gaps in the region contour. MSER, on the other hand, has been shown to be relatively sensitive to image blur in [145], as this directly affects the stability criterion. This problem has been recently addressed in [177]. However, apart from the case of image blur, MSER scores best with respect to repeatability in [145].

As discussed earlier, region detectors often detect blob-like structures — although they are not restricted to this type of regions. As a result, they are less complementary to blobs then to corners.

Region-based detectors are typically quite accurate in their localization. They work especially well for images with a well structured scene, clearly delineated regions, such as images containing objects with printed surfaces, buildings etc.

Even though superpixels share some characteristics with the other region detectors, they are not the same. They are non-overlapping, and cover the entire image. Their repeatability suffers from the weak robustness of the segmentation methods. Most importantly, they have been developed in a different context, where the idea is to speed up the image analysis by focussing on the superpixels only instead of analyzing all pixels. Superpixels are hence considered as a bigger equivalent of pixels, which can be described to a first approximation by a single intensity or color value. This is in contrast to local features which should be distinctive and, in the ideal case, uniquely identifiable. However, using region boundaries to build distinctive descriptors may overcome the occlusion problem from which traditional interest points suffer [68].

6

Efficient Implementations

Most feature detectors described so far involve the computation of derivatives or more complex measures such as the second moment matrix for the Harris detector or entropy for the salient regions detector. Since this step needs to be repeated for each and every location in feature coordinate space which includes position, scale and shape, this makes the feature extraction process computationally expensive, thus not suitable for many applications.

In this section we describe several feature detectors that have been developed with computational efficiency as one of the main objectives. The DoGs detector approximates the Laplacian using multiple scale-space pyramids (see Section 6.1*). SURF makes use of integral images to efficiently compute a rough approximation of the Hessian matrix (Section* 6.2*). FAST evaluates only a limited number of individual pixel intensities using decision trees (see Section* 6.3*).*

6.1 Difference-of-Gaussians

The Difference-of-Gaussians detector, or DoG for short, has been proposed in [47, 76, 81, 124, 126]. It is a scale-invariant detector which extracts blobs in the image by approximating the Laplacian $L_{xx}^2 + L_{yy}^2$ (see also Section 4.1). Based on the diffusion equation in scale-space theory [117, 234, 258], it can be shown that the Laplacian corresponds to the derivative of the image in the scale direction. Since the difference between neighboring points in a given direction approximates the derivative in this direction, the difference between images at different scales approximates the derivative with respect to scale. Furthermore, Gaussian blurring is often applied to generate images at various scales. Hence, the DoG images produce responses which approximate the LoG. The computation of second-order derivatives in x and y directions is then avoided, as illustrated in Figure 6.1.

The actual computation scheme is illustrated in Figure 6.2. The image is smoothed several times with a Gaussian convolution mask. These smoothed versions are combined pairwise to compute a set of DoG blob response maps. Local maxima in these maps are located both over space and over scales with non-maximal suppression, and the locations are further refined with quadratic interpolation. After a few smoothing steps, the image can be subsampled to process the next octave.

Since the Laplacian gives strong response on edges, an additional filtering step is added, where the eigenvalues of the full Hessian matrix are computed and their strengths evaluated. This filtering step does not affect the overall processing time too much, as it is only needed for a limited number of image locations and scales. The DoG features detected in our example images are shown in Figure 6.3. Several frames per second can be processed with this method.

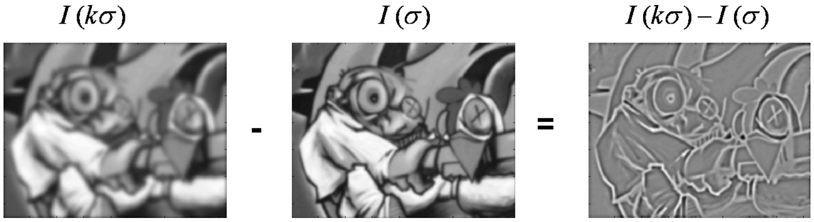

Fig. 6.1 The Laplacian can be approximated as a difference of two Gaussian smoothed images.

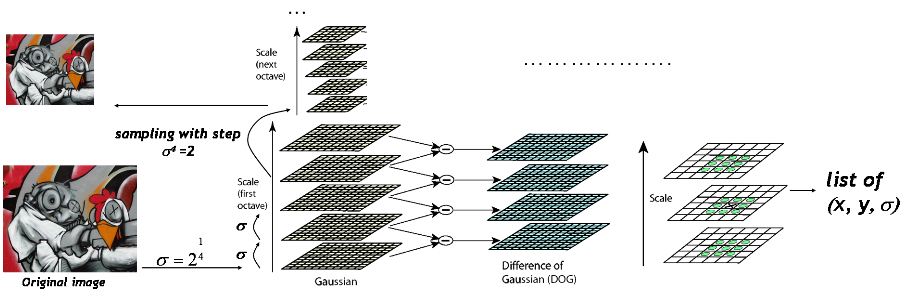

Fig. 6.2 Overview of the DoG-detection scheme.

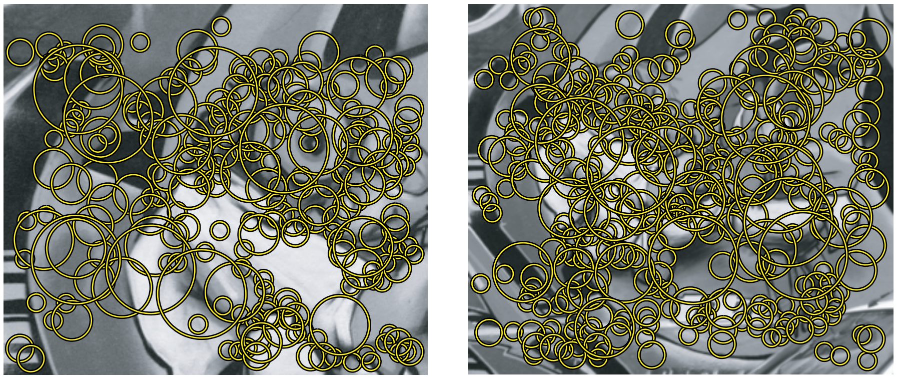

Fig. 6.3 Local features detected with the DoG-detector.

6.2 SURF: Speeded Up Robust Features

In the context of realtime face detection, Viola and Jones have proposed to use *integral images* [252], which allow for very fast computation of Haar wavelets or any box-type convolution filter. First, we will describe the basic idea of integral images. Then we show how this technique can be used to obtain a fast approximation of the Hessian matrix, as used in SURF (Speeded-Up Robust Features) [15].

6.2.1 Integral Images

The entry of an integral image $I_{\Sigma}(\mathbf{x})$ at a location $\mathbf{x} = (x, y)$ represents the sum of all pixels in the input image I of a rectangular region formed

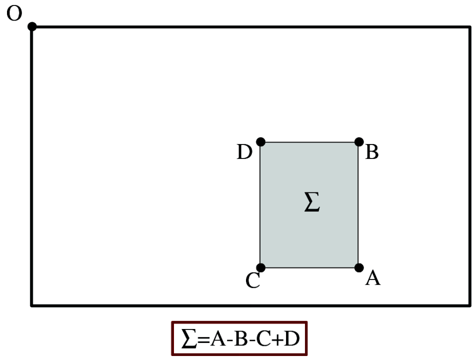

Fig. 6.4 Using integral images, it takes only four operations to calculate the area of a rectangular region of any size.

by the origin and **x**.

$$I_{\Sigma}(\mathbf{x}) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i,j). \tag{6.1}$$

Once the integral image has been computed, it takes four additions to calculate the sum of the intensities over any upright, rectangular area, as shown in Figure 6.4. Moreover, the calculation time is independent of the size of the rectangular area.

6.2.2 SURF

SURF or Speeded Up Robust Features have been proposed by Bay et al. [15, 14]. It is a scale-invariant feature detector based on the Hessian-matrix, as is, e.g., the Hessian-Laplace detector (see Section 4.2). However, rather than using a different measure for selecting the location and the scale, the determinant of the Hessian is used for both. The Hessian matrix is roughly approximated, using a set of box-type filters, and no smoothing is applied when going from one scale to the next.

Gaussians are optimal for scale-space analysis [7, 67, 106, 117], but in practice they have to be discretized (Figure 6.5 left) which introduces artifacts, in particular in small Gaussian Kernels. SURF pushes the approximation even further, using the box filters as shown in the right

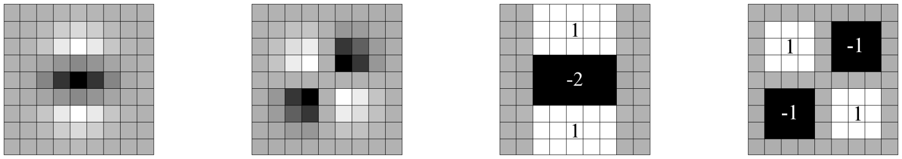

Fig. 6.5 Left to right: the (discretised and cropped) Gaussian second-order partial derivative in y-direction and xy-direction respectively; SURF's box-filter approximation for the second-order Gaussian partial derivative in y-direction and xy-direction. The gray regions are equal to zero.

half of Figure 6.5. These approximate second-order Gaussian derivatives, and can be evaluated very fast using integral images, independently of their size. Surprisingly, in spite of the rough approximations, the performance of the feature detector is comparable to the results obtained with the discretized Gaussians. Box filters can produce a sufficient approximation of the Gaussian derivatives as there are many other sources of significant noise in the processing chain.

The 9 × 9 box filters in Figure 6.5 are approximations for a Gaussian with $\sigma = 1.2$ and represent the finest scale (i.e., highest spatial resolution). We will denote them by D_{xx}, D_{yy}, and D_{xy}. The weights applied to the rectangular regions are kept simple for computational efficiency, but we need to further balance the relative weights in the expression for the Hessian's determinant with $\frac{|L_{xy}(1.2)|_F|Dxx/yy(9)|_F}{|L_{xx/yy}(1.2)|_F|D_{xy}(9)|_F} = 0.616\ldots \simeq 0.6$ for the smallest scale, where $|x|_F$ is the Frobenius norm. This yields

$$\det(\mathcal{H}_{\text{approx}}) = D_{xx}D_{yy} + (0.6D_{xy})^2. \tag{6.2}$$

The approximated determinant of the Hessian represents the blob response in the image at location x. These responses are stored in a blob response map, and local maxima are detected and refined using quadratic interpolation, as with DoG (see Section 6.1). Figure 6.6 shows the result of the SURF detector for our example images. SURF has been reported to be more than five times faster than DoG.

6.3 FAST: Features from Accelerated Segment Test

The FAST detector, introduced by Rosten and Drummond in [202, 203] builds on the SUSAN detector [232] previously discussed in Section 3.3. SUSAN computes the fraction of pixels within a neighborhood which

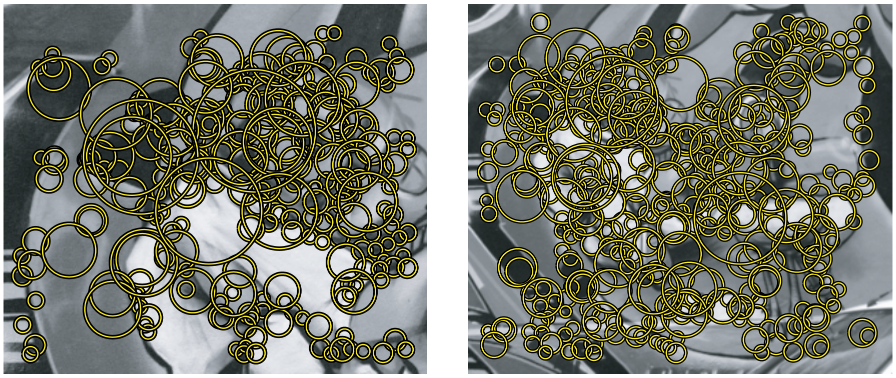

Fig. 6.6 Local features detected with the SURF-detector.

have similar intensity to the center pixel. This idea is taken further by FAST, which compares pixels only on a circle of fixed radius around the point. The test criterion operates by considering a circle of 16 pixels around the corner candidate (see Figure 6.7). Initially pixels 1 and 2 are compared with a threshold, then 3 and 4 as well as the remaining ones at the end. The pixels are classified into dark, similar, and brighter subsets. The ID3 algorithm from [178] is used to select the pixels which yield the most information about whether the candidate pixel is a corner. This is measured by the entropy of the positive and

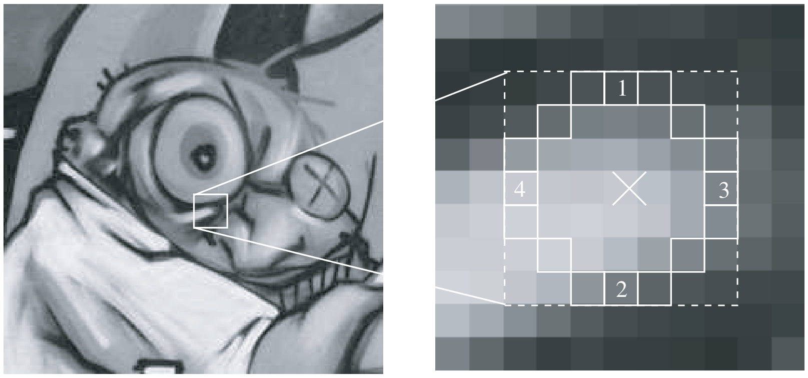

Fig. 6.7 Illustration of pixels examined by the FAST detector.

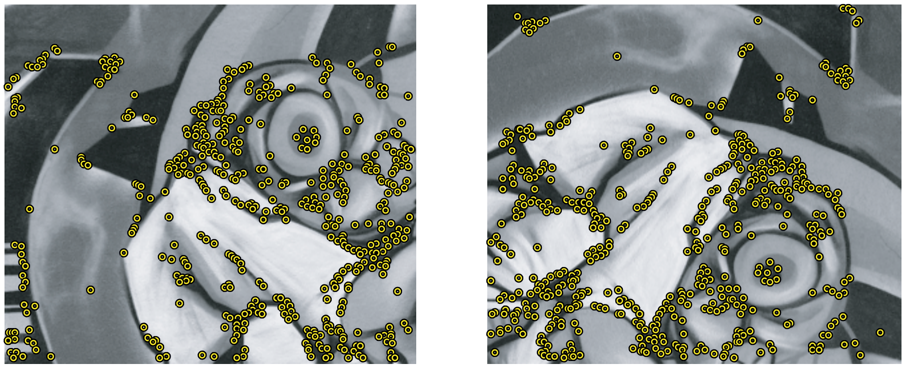

Fig. 6.8 Local features detected with the FAST detector.

negative corner classification responses based on this pixel. The process is applied recursively on all three subsets and terminates when the entropy of a subset is zero. The decision tree resulting from this partitioning is then converted into C-code, creating a long string of nested if-then-else statements which is compiled and used as a corner detector. Finally non-maxima suppression is applied on the sum of the absolute difference between the pixels in the circle and the center pixel. This results in a very efficient detector which is up to 30 times faster than the DoG detector discussed in Section 6.1 — albeit not invariant to scale changes. The FAST features found in our example images are displayed in Figure 6.8.

An extension to a multi-scale detector by scale selection with the Laplacian function was proposed in [112]. They estimate the Laplacian using gray-level differences between pixels on the circle and the central one and retain only the locations where this estimate is largest. This proves to be sufficient to produce a large number of keypoint candidates from which the unstable ones are filtered out during the recognition process.

6.4 Discussion

The ultimate goal of methods focussing on efficiency is often realtime processing of a video stream or dealing with large amounts of data.

However, to some extent, this is a moving target. Computation power increases rapidly over time but so does the number of features we extract or the size of the databases we deal with. Moreover, feature detection is not the final goal, but just the first step in a processing chain, followed by matching, tracking, object recognition, etc. In many applications significant performance improvement can be obtained just by increasing the number of training examples. Efficiency is therefore one of the major properties equally important to invariance or robustness which should be considered when designing or selecting a feature detector.

Coming back to the first point, especially the advent of powerful graphical processing units opens up new possibilities. Apart from the methods described above, which obtain a speedup by platform independent algorithmic changes, further speedups become possible by exploiting the special structure and parallelism that can be realized with GPUs. Some examples of such work can be found in [89, 230]. An FPGA-based implementation of the Harris-Affine feature detector (see Section 3.4) is discussed in [28] and of the DoG detector (see Section 6.1 in [216]. This significantly reduces the time needed to compute all features on a normal-sized image and enables video frame rate processing. In spite of this new trend, the basic ideas and methods described in this section still hold, as they are sufficiently general and widely applicable.

Finally, more efficient methods usually come at a price. A tradeoff has to be made between efficiency on the one hand and accuracy or repeatability on the other hand. Surprisingly, the DoG, SURF, and FAST detectors are competitive with the standard, more computationally expensive feature detectors and may produce better results for some applications.

7

Discussion and Conclusion

In this final section of our survey, we give an overview of the previously discussed methods and highlight their respective strengths and weaknesses. We give some hints on how to use these features, and on how to select the appropriate feature detector for a given application. Finally, we discuss some open issues and future research directions.

7.1 How to Select Your Feature Detector?

Below, we give a few guidelines on what feature detector to use for a given application. This does not give a precise and definitive answer, but indicates a few points one needs to consider when searching for a suitable detector. We refer the reader to Section 1.4 where we define the properties of local features often mentioned here.

First, we organized the feature detectors in this survey based on the *type of image structures* they extract — corners, blobs or regions. Depending on the image content, some of these image structures are more common than others, thus the number of features found with a given detector may vary for different image categories. If little is known about the image content in advance, it is generally recommended to combine different complementary detectors, i.e., extracting different types of features.

Second, feature detectors can be distinguished based on the *level of invariance*. There have been many evaluations which focus on this property [145, 157, 215]. One might be tempted to always select the highest level of invariance available, so as to compensate for as much variability as possible. However, the discriminative power of features is reduced at increased levels of invariance. As more patterns are to be judged equivalent, there is more parameters to estimate, thus more possible sources of noise. Also, the feature detection process becomes more complex, which affects both the computational complexity as well as the repeatability. As a result, a basic rule of thumb is to use no more invariance than what is truly needed by the application at hand. Moreover, if the expected transformations are relatively small, it is often better to count on the robustness of the feature detection and description rather than to increase the level of invariance. That is also the reason why feature detectors invariant to perspective transformations are of little use.

All detectors discussed in this survey are invariant to translations and rotations. The former automatically follows from the use of local features. The latter can be achieved relatively easily at limited extra cost. Sometimes, rotation invariance is not required — e.g., if all images are taken upright and the objects are always upright as well (buildings, cars, etc.). In these cases, the rotation invariant detectors can be combined with a rotation variant descriptor, to ensure good discriminative power. In all other cases, a descriptor with at most the same level of invariance as the detector is preferred.

Finally, there are a number of *qualitative properties* of the detectors to consider. Depending on the application scenario, some of these properties are more crucial than others. When dealing with category-level object recognition, robustness to small appearance variations is important to deal with the within-class variability. When fitting a parametric model to the data, as for camera calibration or 3D modeling, the localization accuracy is essential. For online applications or applications where a large amount of data needs to be processed, efficiency is the most important criterion.

7.2 Summary on the Detectors

Table 7.1 gives an overview of the most important properties for the feature detectors described in Sections 3–6.

The feature detectors in Table 7.1 are organized in 4 groups according to their invariance: rotation, similarity, affine, and perspective. We compare the properties within each group. For rotation invariant features the highest repeatability and localization accuracy in many tests has been obtained by the Harris detector. The Hessian detector finds blobs which are not as well localized and requires second-order derivatives to be computed. The SUSAN detector avoids computation of derivatives and is known for its efficiency, however the absence of smoothing makes it more susceptible to noise. All the rotation invariant methods are suitable for applications where only the spatial location of the features is used and no large scale changes are expected, e.g., structure from motion or camera calibration.

In the scale-invariant group Harris-Laplace shows high repeatability and localization accuracy inherited from the Harris detector [215]. However, its scale estimation is less accurate due to the multiscale nature of corners. Hessian-Laplace is more robust than its single scale version [145]. This is due to the fact that blob-like structures are better localized in scale than corners and the detector benefits from multiscale analysis although it is less accurately localized in the image plane. DoG and SURF detectors were designed for efficiency and the other properties are slightly compromised. However, for most applications they are still more than sufficient. Quantity and good coverage of the image are crucial in recognition applications, where localization accuracy is less important. Thus, Hessian-Laplace detectors have been successful in various categorization tasks although there are detectors with higher repeatability rate. Random and dense sampling also provide good results in this context which confirms the coverage requirements of recognition methods [169] — although they result in far less compact representations than the interest points. DoG detector performs extremely well in matching [26] and image retrieval [124] probably due to a good balance between spatial localization and scale estimation accuracy.

Table 7.1 Overview of feature detectors.

Feature Detector	Corner	Blob	Region	Rotation invariant	Scale invariant	Affine invariant	Repeatability	Localization accuracy	Robustness	Efficiency
Harris	√			√			+++	+++	+++	++
Hessian		√		√			++	++	++	+
SUSAN	√			√			++	++	++	+++
Harris-Laplace	√	(√)		√	√		+++	+++	++	+
Hessian-Laplace	(√)	√		√	√		+++	+++	+++	+
DoG	(√)	√		√	√		++	++	++	++
SURF	(√)	√		√	√		++	++	++	+++
Harris-Affine	√	(√)		√	√	√	+++	+++	++	++
Hessian-Affine	(√)	√		√	√	√	+++	+++	+++	++
Salient Regions	(√)	√		√	√	(√)	+	+	++	+
Edge-based	√			√	√	√	+++	+++	+	+
MSER			√	√	√	√	+++	+++	++	+++
Intensity-based			√	√	√	√	++	++	++	++
Superpixels			√	√	(√)	(√)	+	+	+	+

Note that for the scale and affine invariant detectors, the difference between corner and blob detectors becomes less outspoken, with most detectors detecting a mixture of both feature types — although they still show a preference for either type.

The affine invariant Harris and Hessian follow the observations from previous groups. Salient regions require to compute a histogram and its entropy for each region candidate in scale or affine space, which results in large computational cost [145]. On the positive side the regions can be ranked according to their complexity or information content. Some applications exploit this and use only a small subset of the salient regions while still obtaining a good performance in, e.g., recognition [65]. Originally, they were only scale-invariant, but later they have been extended to affine invariance. The edge based regions focus on corners formed by edge junctions which gives good localization accuracy and repeatability but the number of detected features is small.

The region detectors are based on the idea of segmenting boundaries of uniform regions. Intensity based regions use a heuristic method and find similar regions to MSER. Superpixels are typically based on segmentation methods which are computationally expensive like normalized cuts. The level of invariance of superpixels depends mostly on the segmentation algorithm used. In contrast to superpixels, MSER selects only the most stable regions which results in high repeatability. MSER is also efficient due to the use of a watershed segmentation algorithm. Affine invariant detectors are beneficial in cases where extreme geometric deformations are expected. Otherwise their scale-invariant counterparts usually perform better, in particular for category recognition [139]. This can be understood from the fact that viewpoint changes up to 30 degrees can usually be dealt with by robustness instead of invariance. Affine deformations are more frequent when the same objects are observed from significantly different viewpoints, e.g., in the context of matching or retrieval. In the case of category recognition variability of object appearance dominates over deformations due to viewpoint changes and affine invariance typically brings little improvement.

7.3 Future Work

So far no theory emerged which would provide guidance in what features should be extracted from images or how they should be sampled regardless the application. It is not clear whether it is possible to have a more principled theory on generic feature extraction.

Since memory requirements became less of an issue, brute-force approaches which extract various types of features, densely covering images seem to obtain better and better results, e.g., in object category recognition. However, it has been shown frequently that careful design of image measurements leads to better performance regardless the subsequent components of the system. Even though a lot of progress has been made in the domain of feature extraction — especially with respect to the level of invariance, and even though impressive applications have been built using local features, they still have a number of shortcomings. We would like to emphasize again that for different applications different feature properties may be important and the success of an approach largely depends on the appropriate selection of features. For example in an application like registration of a scene observed from different viewpoints the underlying principles are very clear and repeatability, invariance as well as quantity of features, as defined in Section 1.4, are crucial. In category recognition it is hard to define and measure the repeatability therefore robustness to small appearance variations matters more.

7.3.1 Limited Repeatability

In spite of their success, the repeatability of the local feature detectors is still very limited, with repeatability scores below 50% being quite common (see e.g., [145]). This indicates there is still room for improvement.

7.3.2 Limited Robustness

One of the major drawbacks of all feature detectors is a weak robustness to various transformations in particular in estimation of local scale and shape. The extraction methods are very unstable for small regions. They produce stable scale and shape estimates for large support regions but then other effects like occlusion and background clutter start to affect the results. Methods that extract stable features over a wide range of scales will be very beneficial for various applications.

7.3.3 Lack of Semantic Interpretation

After vector quantization, these local features are often referred to as *visual words* or *object parts*. Yet, this is over-optimistic, as they do not have any semantic connotation. They are just local image fragments which sometimes correspond to meaningful object parts (e.g., wheels of a car) only by coincidence. From a purely bottom-up approach, this is all one can expect. Yet, bringing in top-down information or external knowledge about the world, semantically meaningful object parts could probably be discovered. This could be in the form of an intermediate level representation, or as novel category-specific local features that are learnt from a set of training data.

7.3.4 Automatic Selection of the Optimal Feature Detector

There is a range of feature detectors available, which all have their strengths and weaknesses. Which one performs best depends not only on the application, but also on the image content. To circumvent this problem, researchers often use several detectors in parallel. However, this has a negative impact on the needed computation time. A tool that could quickly gather some image statistics and suggest the most suitable detector would be a valuable instrument for time-critical applications.

7.3.5 Complementary Features

New image measurements with a focus on complementarity of features is another direction to explore. Overcomplete representations, which result from the simultaneous use of multiple detectors provide a temporary solution only in spite of efficient multi-type feature detectors. An efficient combination of complementary detectors or a multi-type detector providing complementary features for compact representation would be much more useful given the increasing amounts of data to process.

7.3.6 Performance Evaluation

By far the most frequently used evaluation measure is the repeatability of features. While this is one of the most important properties it does not guarantee high performance in a given application. New criteria should take into account all the properties discussed in Section 1.4. Another property that is crucial for generative recognition approaches and should also be addressed in general performance evaluations is the reconstruction capability of features. Furthermore, complementarity measures of various detectors are still to be defined. In general, a more principled and probabilistic way of evaluating features which would give a good performance prediction in various applications would be valuable. There is sufficient amount of test data which emphasizes various aspects of features and was explicitly created for evaluating feature detectors. It would be useful to organize it in a common evaluation framework with well defined tests and criteria. Automatic tools which would perform extensive evaluations given a feature detector with specified input and output format, would be of great use.

7.4 Conclusions

Local features are a popular tool for image description nowadays. They are the standard representation for wide baseline matching and object recognition, both for specific objects as well as for category-level schemes.

In this survey, we gave an overview of some of the most widely used detectors, with a qualitative evaluation of their respective strengths and weaknesses, which can be found at the end of the sections and chapters. We also put the work on local feature detection in context, by summarizing the progress in feature detection from the early days of computer vision up to now. These early works from the pre-internet era tend to be forgotten. Yet they contain valuable insights and ideas, that can inspire future research on local features and avoid a waste of resources by reinventing the wheel. The literature is huge, and we could only touch the different contributions without going into details. Yet, we hope to provide the right pointers so those who are interested have a starting point and can delve deeper if they want to.

Acknowledgments

The authors would like to acknowledge support from the Network of Excellence PASCAL and the Flemish Fund for Scientific Research FWO. This work was also supported by EPSRC EP/F003420/1 and VIDI-Video IST-2-045547 project.

References

[1] A. Almansa and T. Lindeberg, "Fingerprint enhancement by shape adaptation of scale–space operators with automatic scale selection," *IEEE Transactions on Image Processing*, vol. 9, no. 12, pp. 2027–2042, 2000.

[2] L. Alvarez and F. Morales, "Affine morphological multiscale analysis of corners and multiple junctions," *International Journal of Computer Vision*, vol. 2, no. 25, pp. 95–107, 1997.

[3] I. M. Anderson and J. C. Bezdek, "Curvature and tangential deflection of discrete arcs: A theory based on the commutator of scatter matrix pairs and its application to vertex detection in planar shape data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 27–40, 1984.

[4] N. Ansari and E. J. Delp, "On detecting dominant points," *Pattern Recognition*, vol. 24, no. 5, pp. 441–451, 1991.

[5] H. Asada and M. Brady, "The curvature primal sketch," *Pattern Analysis and Applications*, vol. 8, no. 1, pp. 2–14, 1986.

[6] F. Attneave, "Some informational aspects of visual perception," *Psychological Review*, vol. 61, pp. 183–193, 1954.

[7] 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*, vol. 8, no. 1, pp. 26–33, 1986.

[8] S. C. Bae, I.-S. Kweon, and C. Don Yoo, "COP: A new corner detector," *Pattern Recognition Letters*, vol. 23, no. 11, pp. 1349–1360, 2002.

[9] R. Bajcsy, "Computer identification of visual surface," *Computer and Graphics Image Processing*, vol. 2, pp. 118–130, 1973.

[10] R. Bajcsy and D. A. Rosenthal, *Visual and Conceptual Focus of Attention Structured Computer Vision.* Academic Press, 1980.

[11] J. Bala, K. DeJong, J. Huang, H. Vafaie, and H. Wechsler, "Using learning to facilitate the evolution of features for recognizing visual concepts," *Evolutionary Computation*, vol. 4, no. 3, pp. 297–311, 1996.

[12] J. Bauer, H. Bischof, A. Klaus, and K. Karner, "Robust and fully automated image registration using invariant features," *International Society for Photogrammetry and Remote Sensing*, 2004.

[13] A. Baumberg, "Reliable feature matching across widely separated views," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 774–781, 2000.

[14] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *International Journal on Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[15] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proceedings of the European Conference on Computer Vision*, pp. 404–417, 2006.

[16] P. R. Beaudet, "Rotationally invariant image operators," in *Proceedings of the International Joint Conference on Pattern Recognition*, pp. 579–583, 1978.

[17] S. Belongie, J. Malik, and J. Puzicha,, "Shape context: A new descriptor for shape matching and object recognition," in *Proceedings of the Neural Information Processing Systems*, pp. 831–837, 2000.

[18] H. L. Beus and S. S. H. Tiu, "An improved corner detection algorithm based on chain-coded plane curves," *Pattern Recognition*, vol. 20, no. 3, pp. 291–296, 1987.

[19] D. J. Beymer, "Finding junctions using the image gradient," in *International Conference Computer Vision and Pattern Recognition*, pp. 720–721, 1991.

[20] I. Biederman, "Recognition-by-components: A theory of human image understanding," *Psychological Review*, vol. 2, no. 94, pp. 115–147, 1987.

[21] Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N–D images," in *Proceedings of the International Conference on Computer Vision,* vol. 1, pp. 105–112, 2001.

[22] P. Brand and R. Mohr, "Accuracy in image measure," *SPIE Conference on Videometrics III*, vol. 2350, pp. 218–228, 1994.

[23] V. Brecher, R. Bonner, and C. Read, "A model of human preattentive visual detection of edge orientation anomalies," in *Proceedings of the SPIE Conference of Visual Information Processing: From Neurons to Chips*, vol. 1473, pp. 39–51, 1991.

[24] L. Bretzner and T. Lindeberg, "Feature tracking with automatic selection of spatial scales," *Computer Vision and Image Understanding*, vol. 71, no. 3, pp. 385–392, 1998.

[25] C. R. Brice and C. L. Fennema, "Scene analysis using regions," *Artificial Intelligence*, vol. 1, pp. 205–226, 1970.

[26] M. Brown and D. Lowe, "Recognising panoramas," in *Proceedings of the International Conference on Computer Vision*, pp. 1218–1227, 2003.

[27] P. J. Burt and E. H. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 9, no. 4, pp. 532–540, 1983.

[28] C. Cabani and W. J. MacLean, "Implementation of an affine-covariant feature detector in field-programmable gate arrays," in *Proceedings of the International Conference on Computer Vision Systems*, 2007.

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

[30] P. Carbonetto, N. de Freitas, and K. Barnard, "A statistical model for general contextual object recognition," in *Proceedings of the European Conference on Computer Vision, part I*, pp. 350–362, 2004.

[31] C. Carson, S. Belongie, S. Greenspan, and J. Malik, "Blobworld: Image segmentation using expectation-maximization and its application to image querying," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1026–1038, 2002.

[32] K. R. Cave and J. M. Wolfe, "Modeling the role of parallel processing in visual search," *Cognitive Psychology*, vol. 22, pp. 225–271, 1990.

[33] S. P. Chang and J. H. Horng, "Corner point detection using nest moving average," *Pattern Recognition*, vol. 27, no. 11, pp. 1533–1537, 1994.

[34] D. Chapman, "Vision, instruction and action," Technical Report AI-TR-1204, AI Laboratory, MIT, 1990.

[35] C.-H. Chen, J.-S. Lee, and Y.-N. Sun, "Wavelet transformation for gray-level corner detection," *Pattern Recognition*, vol. 28, no. 6, pp. 853–861, 1995.

[36] M. Chen and P. Yan, "A multiscaling approach based on morphological filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, no. 7, pp. 694–700, 1989.

[37] W.-C. Chen and P. Rockett, "Bayesian labelling of corners using a grey–level corner image mode," in *Proceedings of the International Conference on Image Processing*, pp. 687–690, 1997.

[38] O. Chomat, V. Colin deVerdière, D. Hall, and J. Crowley, "Local scale selection for gaussian based description techniques," in *Proceedings of the European Conference on Computer Vision, Dublin, Ireland*, pp. 117–133, 2000.

[39] J. J. Clark and N. J. Ferrier, "Modal control of attentive vision system," in *Proceedings of the International Conference on Computer Vision*, pp. 514–523, 1988.

[40] C. Coelho, A. Heller, J. L. Mundy, D. A. Forsyth, and A. Zisserman, *An Experimental Evaluation of Projective Invariants.* Cambridge, MA: MIT Press, 1992.

[41] J. Cooper, S. Venkatesh, and L. Kitchen, "The dissimilarity corner detectors," *International Conference on Advanced Robotics*, pp. 1377–1382, 1991.

[42] J. Cooper, S. Venkatesh, and L. Kitchen, "Early jump-out corner detectors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 8, pp. 823–828, 1993.

[43] T. F. Cootes and C. Taylor, "Performance evaluation of corner detection algorithms under affine and similarity transforms," in *Proceedings of the British Machine Vision Conference*, 2001.

[44] J. J. Corso and G. D. Hager, "Coherent regions for concise and stable image description," in *Proceedings of the Conference on Computer Vision and Pattern Recognition,* vol. 2, pp. 184–190, 2005.

[45] J. C. Cottier,, “Extraction et appariements robustes des points d’intérêt de deux images non etalonnées,” Technical Report, LIFIA-IMAG-INRIA, Rhone-Alpes, 1994.

[46] T. Cour and J. Shi, “Recognizing objects by piecing together the segmentation puzzle,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2007.

[47] J. L. Crowley and A. C. Parker, “A representation for shape based on peaks and ridges in the difference of low pass transform,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 2, pp. 156–170, 1984.

[48] J. L. Crowley and A. C. Sanderson, “Multiple resolution representation and probabilistic matching of 2D gray–scale shape,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 1, pp. 113–121, 1987.

[49] S. M. Culhane and J. Tsotsos, “An attentional prototype for early vision,” in *Proceedings of the European Conference on Computer Vision*, pp. 551–560, 1992.

[50] D. Cyganski and J. A. Or, “Application of tensor theory to object recognition and orientation determination,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, pp. 662–673, 1985.

[51] E. R. Davies, “Application of the generalised hough transform to corner detection,” *IEE Proceedings*, vol. 135, no. 1, pp. 49–54, 1988.

[52] R. Deriche and T. Blaszka, “Recovering and characterizing image features using an efficient model-based approach,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 530–535, 1993.

[53] R. Deriche and G. Giraudon, “Accurate corner detection: An analytical study,” in *Proceedings International Conference on Computer Vision*, pp. 66–70, 1990.

[54] R. Deriche and G. Giraudon, “A computational approach for corner and vertex detection,” *International Journal of Computer Vision*, vol. 10, no. 2, pp. 101–124, 1993.

[55] P. Dias, A. Kassim, and V. Srinivasan, “A neural network based corner detection method,” *in IEEE International Conference on Neural Networks*, vol. 4, pp. 2116–2120, 1995.

[56] M. Donoser and H. Bischof, “Efficient maximally stable extremal region (MSER) tracking,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 553–560, 2006.

[57] L. Dreschler and H. Nagel, “Volumetric model and 3D-trajectory of a moving car derived from monocular TV-frame sequence of a street scene,” *In Computer Graphics and Image Processing*, vol. 20, pp. 199–228, 1982.

[58] R. Duda and P. Hart, *Pattern Classification and Scene Analysis.* Wiley–Interscience, 1973.

[59] Y. Dufournaud, C. Schmid, and R. Horaud, “Matching images with different resolutions,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 612–618, 2000.

[60] J. Dunham, “Optimum uniform piecewise linear approximation of planar curves,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 67–75, 1986.

[61] J. Q. Fang and T. S. Huang, “A corner finding algorithm for image analysis and registration,” in *Proceedings of AAAI Conference*, pp. 46–49, 1982.

[62] L. FeiFei and P. Perona, “A Bayesian hierarchical model for learning natural scene categories,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition,* vol. 2, pp. 524–531, 2005.

[63] F. Y. Feng and T. Pavlidis, “Finding ‘vertices’ in a picture,” *Computer Graphics and Image Processing*, vol. 2, pp. 103–117, 1973.

[64] F. Y. Feng and T. Pavlidis, “Decomposition of polygons into simpler components: Feature generation for syntactic pattern recognition,” *IEEE Transactions on Computers*, vol. C-24, 1975.

[65] 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*, pp. 264–271, 2003.

[66] V. Ferrari, T. Tuytelaars, and L. Van Gool, “Simultaneous object recognition and segmentation by image exploration,” in *Proceedings of the European Conference on Computer Vision*, pp. 40–54, 2004.

[67] 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*, vol. 10, pp. 376–388, 1992.

[68] P.-E. Forssen and D. Lowe, “Shape descriptors for maximally stable extremal regions,” in *Proceedings of the International Conference on Computer Vision*, pp. 59–73, 2007.

[69] W. Förstner, “A framework for low level feature extraction,” in *Proceedings of the European Conference on Computer Vision*, pp. 383–394, 1994.

[70] W. Förstner and E. Gülch, “A fast operator for detection and precise location of distinct points, corners and centres of circular features,” in *Intercommission Conference on Fast Processing of Photogrammetric Data*, pp. 281–305, 1987.

[71] F. Fraundorfer and H. Bischof, “Evaluation of local detectors on non-planar scenes,” in *Proceedings of the Austrian Association for Pattern Recognition Workshop*, pp. 125–132, 2004.

[72] H. Freeman, “A review of relevant problems in the processing of line-drawing data,” in *Automatic Interpretation and Classification of Images*, (A. Graselli, ed.), pp. 155–174, Academic Press, 1969.

[73] H. Freeman, “Computer processing of line drawing images,” *Surveys*, vol. 6, no. 1, pp. 57–97, 1974.

[74] H. Freeman and L. S. Davis, “A corner-finding algorithm for chain-coded curves,” *IEEE Transactions on Computers*, vol. 26, pp. 297–303, 1977.

[75] M. Galun, E. Sharon, R. Basri, and A. Brandt, “Texture segmentation by multiscale aggregation of filter responses and shape elements,” in *Proceedings of the International Conference on Computer Vision,* vol. 2, pp. 716–725, 2003.

[76] P. Gaussier and J. P. Cocquerez, “Neural networks for complex scene recognition: Simulation of a visual system with several cortical areas,” in *Proceedings of the International Joint Conference on Neural Networks,* vol. 3, pp. 233–259, 1992.

[77] S. Ghosal and R. Mehrotra, “Zernike moment–based feature detectors,” in *International Conference on Image Processing*, pp. 934–938, 1994.

[78] G. J. Giefing, H. Janssen, and H. Mallot, “Saccadic object recognition with an active vision system,” in *Proceedings of the European Conference on Artificial Intelligence*, pp. 803–805, 1992.

[79] S. Gilles, *Robust Description and Matching of Image.* PhD thesis, University of Oxford, 1998.

[80] V. Gouet, P. Montesinos, R. Deriche, and D. Pelé, “Evaluation de détecteurs de points d’intérêt pour la couleur,” in *12ème Congrès Francophone AFRIF–AFIA de Reconnaissance des Formes et Intelligence Artificielle*, pp. 257–266, 2000.

[81] S. Grossberg, E. Mingolla, and D. Todorovic, “A neural network architecture for preattentive vision,” *IEEE Transactions on Biomedical Engineering*, vol. 36, pp. 65–84, 1989.

[82] A. Guidicci, “Corner characterization by differential geometry techniques,” *Pattern Recognition Letters*, vol. 8, no. 5, pp. 311–318, 1988.

[83] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision.* Addison-Wesley, pp. 453–507, 1993.

[84] C. Harris and M. Stephens, “A combined corner and edge detector,” in *Alvey Vision Conference*, pp. 147–151, 1988.

[85] T. I. Hashimoto, S. Tsujimoto, and S. Arimoto, “Spline approximation of line images by modified dynamic programming,” *Transactions of IECE of Japan*, vol. J68, no. 2, pp. 169–176, 1985.

[86] X. C. He and N. H. C. Yung, “Curvature scale space corner detector with adaptive threshold and dynamic region of support,” in *International Conference on Pattern Recognition*, pp. 791–794, 2004.

[87] F. Heitger, L. Rosenthaler, R. von der Heydt, E. Peterhans, and O. Kubler, “Simulation of neural contour mechanisms: From simple to end-stopped cells,” *Vision Research*, vol. 32, no. 5, pp. 963–981, 1992.

[88] A. Heyden and K. Rohr, “Evaluation of corner extraction schemes using invariance method,” in *Proceedings of the International Conference on Pattern Recognition*, pp. 895–899, 1996.

[89] S. Heymann, K. Maller, A. Smolic, B. Froehlich, and T. Wiegand, “SIFT implementation and optimization for general-purpose GPU,” in *Proceedings of the International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2007.

[90] J. Hong and X. Tan, “A new approach to point pattern matching,” in *Proceedings of the International Conference on Pattern Recognition*, vol. 1, pp. 82–84, 1988.

[91] R. Horaud, T. Skordas, and F. Veillon, “Finding geometric and relational structures in an image,” in *Proceedings of the European Conference on Computer Vision*, pp. 374–384, 1990.

[92] D. Hubel and T. Wiesel, “Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex,” *Journal of Physiology*, vol. 160, pp. 106–154, 1962.

[93] L. Itti and C. Koch, “Computational modeling of visual attention,” *Nature Reviews Neuroscience*, vol. 2, no. 3, pp. 194–203, 2001.

[94] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 1254–1259, 1998.

[95] Q. Ji and R. M. Haralick, "Corner detection with covariance propagation," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 362–367, 1997.

[96] B. Julesz, "Textons, the elements of texture perception, and their interactions," *Nature*, vol. 290, pp. 91–97, 1981.

[97] F. Jurie and B. Triggs, "Creating efficient codebooks for visual recognition," in *Proceedings of the International Conference on Computer Vision*, pp. 604–610, 2005.

[98] T. Kadir and M. Brady, "Scale, saliency and image description," *International Journal of Computer Vision*, vol. 45, no. 2, pp. 83–105, 2001.

[99] T. Kadir, M. Brady, and A. Zisserman, "An affine invariant method for selecting salient regions in images," in *Proceedings of the European Conference on Computer Vision*, pp. 345–457, 2004.

[100] 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*, pp. 511–517, 2004.

[101] C. Kenney, B. Manjunath, M. Zuliani, G. Hewer, and A. Van Nevel, "A condition number for point matching with application to registration and post-registration error estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 11, pp. 1437–1454, 2003.

[102] C. S. Kenney, M. Zuliani, and B. S. Manjunath, "An axiomatic approach to corner detection," *International Conference on Computer Vision and Pattern Recognition*, pp. 191–197, 2005.

[103] W. Kienzle, F. A. Wichmann, B. Scholkopf, and M. O. Franz, "Learning an interest operator from human eye movements," in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshop*, pp. 1–8, 2005.

[104] L. Kitchen and A. Rosenfeld, "Gray-level corner detection," *Pattern Recognition Letters*, vol. 1, pp. 95–102, 1982.

[105] C. Koch and S. Ullman, "Shifts in selective visual attention: Towards the underlying neural circuitry," *Human Neurobiology*, vol. 4, no. 4, pp. 219–227, 1985.

[106] J. J. Koenderink, "The structure of images," *Biological Cybernetics*, vol. 50, pp. 363–396, 1984.

[107] R. Laganiere, "A morphological operator for corner detection," *Pattern Recognition*, vol. 31, no. 11, pp. 1643–1652, 1998.

[108] D. J. Langridge, "Curve encoding and detection of discontinuities," *Computer Graphics Image Processing*, vol. 20, pp. 58–71, 1982.

[109] I. Laptev and T. Lindeberg, "Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features," in *Proceedings of Scale-Space and Morphology Workshop*, pp. 63–74, Lecture Notes in Computer Science, 2001.

[110] 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*, pp. 319–324, 2003.

[111] J. S. Lee, Y. N. Sun, C. H. Chen, and C. T. Tsai, "Wavelet based corner detection," *Pattern Recognition*, vol. 26, pp. 853–865, 1993.

[112] V. Lepetit and P. Fua, "Keypoint recognition using randomized trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1465–1479, 2006.

[113] L. Li and W. Chen, "Corner detection and interpretation on planar curves using fuzzy reasoning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 1204–1210, 1999.

[114] R.-S. Lin, C.-H. Chu, and Y.-C. Hsueh, "A modified morphological corner detector," *Pattern Recognition Letters*, vol. 19, no. 3, pp. 279–286, 1998.

[115] T. Lindeberg, "Scale-space for discrete signals," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp. 234–254, 1990.

[116] T. Lindeberg, "Detecting salient blob-like image structures and their scales with a scale-space primal sketch – a method for focus-of-attention," *International Journal of Computer Vision*, vol. 11, no. 3, pp. 283–318, 1993.

[117] T. Lindeberg, *Scale-Space Theory in Computer Vision.* Kluwer Academic Publishers, 1994.

[118] T. Lindeberg, "Direct estimation of affine image deformation using visual front-end operations with automatic scale selection," in *Proceedings of the International Conference on Computer Vision*, pp. 134–141, 1995.

[119] T. Lindeberg, "Edge detection and ridge detection with automatic scale selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 79–116, 1998.

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

[121] 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*, vol. 15, no. 6, pp. 415–434, 1997.

[122] H. Ling and D. Jacobs, "Deformation invariant image matching," in *Proceedings of the International Conference on Computer Vision,* vol. 2, pp. 1466–1473, 2005.

[123] S.-T. Liu and W.-H. Tsai, "Moment preserving corner detection," *Pattern Recognition*, vol. 23, no. 5, pp. 441–460, 1990.

[124] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 2, no. 60, pp. 91–110, 2004.

[125] D. G. Lowe, "Organization of smooth image curves at multiple scales," *International Conference on Computer Vision*, pp. 558–567, 1988.

[126] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision*, pp. 1150–1157, 1999.

[127] G. Loy and A. Zelinsky, "Fast radial symmetry for detecting points of interest," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 959–973, 2003.

[128] B. Luo, A. D. J. Cross, and E. R. Hancock, "Corner detection via topographic analysis of vector potential," *Pattern Recognition Letters*, vol. 20, no. 6, pp. 635–650, 1998.

[129] J. Malik, S. Belongie, J. Shi, and T. Leung, "Textons, contours and regions: Cue integration in image segmentation," in *Proceedings of the International Conference on Computer Vision*, pp. 918–925, 1999.

[130] J. Malik and P. Perona, "Preattentive texture discrimination with early vision mechanism," *Journal of Optical Society of America A*, vol. 7, no. 5, pp. 923–932, 1990.

[131] B. S. Manjunath, C. Shekhar, and R. Chellappa, "A new approach to image feature detection with applications," *Pattern Recognition*, vol. 29, no. 4, pp. 627–640, 1996.

[132] R. Maree, P. Geurts, J. Piater, and L. Wehenkel, "Random subwindows for robust image classification," in *Proceedings of the Conference on Computer Vision and Pattern Recognition,* vol. 1, pp. 34–40, 2005.

[133] D. Marr, *Vision.* USA, San Francisco, CA: W.H. Freeman and Company, 1982.

[134] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," in *Proceedings of the British Machine Vision Conference*, pp. 384–393, 2002.

[135] J. Matas, S. Obdrzalek, and O. Chum, "Local affine frames for wide-baseline stereo," in *Proceedings of 16th International Conference Pattern Recognition,* vol. 4, pp. 363–366, 2002.

[136] G. Medioni and Y. Yasumoto, "Corner detection and curve representation using cubic B-spline," *Computer Vision, Graphics and Image Processing*, vol. 39, no. 1, pp. 267–278, 1987.

[137] R. Mehrotra, S. Nichani, and N. Ranganathan, "Corner detection," *Pattern Recognition*, vol. 23, no. 11, pp. 1223–1233, 1990.

[138] K. Mikolajczyk, *Scale and Affine Invariant Interest Point Detectors.* PhD thesis, 2002. INRIA Grenoble.

[139] K. Mikolajczyk, B. Leibe, and B. Schiele, "Local features for object class recognition," in *Proceedings of the International Conference on Computer Vision*, pp. 525–531, 2005.

[140] K. Mikolajczyk, B. Leibe, and B. Schiele, "Multiple object class detection with a generative model," in *Proceedings the Conference on Computer Vision and Pattern Recognition*, pp. 26–36, 2006.

[141] K. Mikolajczyk and C. Schmid, "Indexing based on scale-invariant interest points," in *Proceedings of the International Conference on Computer Vision*, pp. 525–531, Vancouver, Canada, 2001.

[142] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," in *Proceedings of the European Conference on Computer Vision*, pp. 128–142, 2002.

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

[144] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[145] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1/2, pp. 43–72, 2005.

[146] K. Mikolajczyk, A. Zisserman, and C. Schmid, "Shape recognition with edge based features," in *Proceedings of the British Machine Vision Conference*, pp. 779–788, 2003.

[147] R. Milanese, *Detecting Salient Regions in an Image: From Biological Evidence to Computer Implementation.* PhD thesis, University of Geneva, 1993.

[148] R. Milanese, J.-M. Bost, and T. Pun, "A bottom-up attention system for active vision," in *Proceedings of the European Conference on Artificial Intelligence*, pp. 808–810, 1992.

[149] D. Milgram, "Computer methods for creating photomosaics," *IEEE Transactions on Computers*, vol. 23, pp. 1113–1119, 1975.

[150] F. Mohanna and F. Mokhtarian, "Performance evaluation of corner detection algorithms under affine and similarity transforms," in *Proceedings of the British Machine Vision Conference*, 2001.

[151] F. Mokhtarian and A. Mackworth, "Scale-based description of plannar curves and two-dimensional shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 34–43, 1986.

[152] F. Mokhtarian and R. Suomela, "Robust image corner detection through curvature scale-space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1376–1381, 1998.

[153] P. Montesinos, V. Gouet, and R. Deriche, "Differential invariants for color images," in *Proceedings of the International Conference on Pattern Recognition*, pp. 838–840, 1998.

[154] H. Moravec, "Towards automatic visual obstacle avoidance," in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 584–590, 1977.

[155] H. Moravec, "Visual mapping by a robot rover," in *Proceedings of the International Joint Conference on Artificial Intellingence*, pp. 598–600, 1979.

[156] H. Moravec, "Rover visual obstacle avoidance," in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 785–790, 1981.

[157] P. Moreels and P. Perona, "Evaluation of features detectors and descriptors based on 3D objects," in *Proceedings of the International Conference on Computer Vision*, pp. 800–807, 2005.

[158] P. Moreels and P. Perona, "Evaluation of features detectors and descriptors based on 3D objects," *International Journal of Computer Vision*, vol. 73, no. 3, pp. 800–807, 2007.

[159] G. Mori, X. Ren, A. Efros, and J. Malik, "Recovering human body configurations: Combining segmentation and recognition," in *Proceedings of the Conference on Computer Vision and Pattern Recognition,* vol. 2, pp. 326–333, 2004.

[160] H. Murase and S. Nayar, "Visual learning and recognition of 3D objects from appearance," *International Journal on Computer Vision*, vol. 14, no. 1, pp. 5–24, 1995.

[161] E. Murphy-Chutorian and M. Trivedi, "N-tree disjoint-set forests for maximally stable extremal regions," in *Proceedings of the British Machine Vision Conference*, 2006.

[162] J. Mutch and D. G. Lowe, "Multiclass object recognition with sparse, localized features," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 11–18, 2006.

[163] H. H. Nagel, "Displacement vectors derived from second-order intensity variations in image sequences," *Computer Vision Graphics, and Image Processing*, vol. 21, pp. 85–117, 1983.

[164] M. Nakajima, T. Agui, and K. Sakamoto, "Pseudo-coding method for digital line figures," *Transactions of the IECE*, vol. J68–D, no. 4, pp. 623–630, 1985.

[165] U. Neisser, "Visual search," *Scientific American*, vol. 210, no. 6, pp. 94–102, 1964.

[166] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 2161–2168, 2006.

[167] J. A. Noble, "Finding corners," *Image and Vision Computing*, vol. 6, no. 2, pp. 121–128, 1988.

[168] J. A. Noble, *Descriptions of Image Surfaces.* PhD thesis, Department of Engineering Science, Oxford University, 1989.

[169] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," in *Proceedings of the European Conference on Computer Vision*, pp. 490–503, 2006.

[170] H. Ogawa, "Corner detection on digital curves based on local symmetry of the shape," *Pattern Recognition*, vol. 22, no. 4, pp. 351–357, 1989.

[171] C. M. Orange and F. C. A. Groen, "Model-based corner detection," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 690–691, 1993.

[172] M. Pabst, H. J. Reitboeck, and R. Eckhorn, "A model of preattentive Region definition based on texture analysis," in *Models of Brain Function*, pp. 137–150, Cambridge, England: Cambridge University Press, 1989.

[173] K. Paler, J. Foglein, J. Illingworth, and J. Kittler, "Local ordered grey levels as an aid to corner detection," *Pattern Recognition*, vol. 17, no. 5, pp. 535–543, 1984.

[174] T. Pavlidis, *Structural Pattern Recognition.* Berlin, Heidelberg, NY: Springer-Verlag, 1977.

[175] T. Pavlidis, *Algorithms for Graphics and Image Processing.* Computer Science Press, 1982.

[176] T. Pavlidis, "Problems in recognition of drawings," *Syntactic and Structural Pattern Recognition*, vol. 45, pp. 103–113, 1988.

[177] M. Perdoch, J. Matas, and S. Obdrzalek, "Stable affine frames on isophotes," in *Proceedings of the International Conference on Computer Vision*, 2007.

[178] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81–106, 1986.

[179] P. Rajan and J. Davidson, "Evaluation of corner detection algorithms," in *21th Southeastern Symposium on System Theory*, pp. 29–33, 1989.

[180] K. Rangarajan, M. Shah, and D. V. Brackle, "Optimal corner detection," *Computer Vision Graphics Image Processing*, vol. 48, pp. 230–245, 1989.

[181] A. Rattarangsi and R. T. Chin, "Scale-based detection of corners of planar curves," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 430–449, 1992.

[182] D. Reisfeld, H. Wolfson, and Y. Yeshurun, "Context free attentional operators: The generalized symmetry transform," *International Journal of Computer Vision*, vol. 14, no. 2, pp. 119–130, 1995.

[183] X. Ren, C. Fowlkes, and J. Malik, "Scale–invariant contour completion using conditional random fields," in *Proceedings of the International Conference on Computer Vision,* vol. 2, pp. 1214–1221, 2005.

[184] X. Ren and J. Malik, "Learning a classification model for segmentation," in *Proceedings of the International Conference on Computer Vision,* vol. 1, pp. 10–17, 2003.

[185] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature Neuroscience*, vol. 2, pp. 1019–1025, 1999.

[186] B. Robbins and R. A. Owens, "2D feature detection via local energy," *Image Vision Comput*, vol. 15, no. 5, pp. 353–368, 1997.

[187] V. Roberto and R. Milanese, "Matching hierarchical structures in a machine vision system," *Intelligent Autonomous Systems*, pp. 845–852, 1989.

[188] K. Rohr, "Recognizing corners by fitting parametric models," *International Journal of Computer Vision*, vol. 9, no. 3, pp. 213–230, 1992.

[189] K. Rohr, "Localization properties of direct corner detectors," *Journal of Mathematical Imaging and Vision*, vol. 4, no. 2, pp. 139–150, 1994.

[190] K. Rohr, "On the precision in estimating the location of edges and corners," *Journal of Mathematical Imaging and Vision*, vol. 7, no. 1, pp. 7–22, 1997.

[191] A. Rosenfeld, "Picture processing by computer," *ACM Computing Surveys*, vol. 1, no. 3, pp. 147–176, 1969.

[192] A. Rosenfeld, "Digital image processing and recognition," *Digital Image Processing*, pp. 1–11, 1977.

[193] A. Rosenfeld and E. Johnston, "Angle detection on digital curves," *IEEE Transactions on Computers*, vol. C-22, pp. 875–878, 1973.

[194] A. Rosenfeld and A. C. Kak, *Digital Picture Processing.* Academic Press, Second Edition, 1982.

[195] A. Rosenfeld and M. Thurston, "Edge and curve detection for digital scene analysis," *IEEE Transactions on Computers*, vol. C-20, pp. 562–569, 1971.

[196] A. Rosenfeld, M. Thurston, and Y. H. Lee, "Edge and curve detection: Further experiments," *IEEE Transactions on Computers*, vol. C-21, pp. 677–715, 1972.

[197] A. Rosenfeld and J. S. Weszka, "An improved method of angle detection on digital curves," *IEEE Transactions on Computers*, vol. 24, no. 9, pp. 940–941, 1975.

[198] L. Rosenthaler, F. Heitger, O. Kubler, and R. von der Heydt, "Detection of general edges and keypoints," in *Proceedings of the European Conference on Computer Vision*, pp. 78–86, 1992.

[199] P. L. Rosin, "Representing curves at their natural scales," *Pattern Recognition*, vol. 25, pp. 1315–1325, 1992.

[200] P. L. Rosin, "Determining local natural scales of curves," *International Conference on Pattern Recognition*, 1994.

[201] P. L. Rosin, "Measuring corner properties," *Computer Vision and Image Understanding*, vol. 73, no. 2, pp. 292–307, 1999.

[202] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," in *Proceedings of the International Conference on Computer Vision*, pp. 1508–1511, 2005.

[203] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Proceedings of the European Conference on Computer Vision*, pp. 430–443, 2006.

[204] C. A. Rothwell, A. Zisserman, D. A. Forsyth, and J. L. Mundy, "Planar object recognition using projective shape representation," *International Journal on Computer Vision*, vol. 16, pp. 57–99, 1995.

[205] W. S. Rutkowski and A. Rosenfeld, "A comparison of corner detection techniques for chain coded curves," Technical Report 623, Maryland University, 1978.

[206] P. A. Sandon, "Simulating visual attention," *Journal of Cognitive Neuroscience*, vol. 2, no. 3, pp. 213–231, 1990.

[207] P. V. Sankar and C. V. Sharma, "A parallel procedure for the detection of dominant points on digital curves," *Computer Graphics and Image Processing*, vol. 7, pp. 403–412, 1978.

[208] F. Schaffalitzky and A. Zisserman, "Viewpoint invariant texture matching and wide baseline stereo," in *Proceedings of the International Conference on Computer Vision*, pp. 636–643, 2001.

[209] F. Schaffalitzky and A. Zisserman, "Multi-view matching for unordered image sets," in *Proceedings of the European Conference on Computer Vision*, pp. 414–431, 2002.

[210] B. Schiele and J. L. Crowley, "Probabilistic object recognition using multidimensional receptive field histograms," in *Proceedings of the International Conference on Pattern Recognition,* vol. 2, pp. 50–54, 1996.

[211] B. Schiele and J. L. Crowley, "Recognition without correspondence using multi-dimensional receptive field histograms," *International Journal of Computer Vision*, vol. 36, no. 1, pp. 31–50, 2000.

[212] C. Schmid and R. Mohr, "Combining gray-value invariants with local constraints for object recognition," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 872–877, 1996.

[213] C. Schmid and R. Mohr, "Local gray-value invariants for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 5, pp. 530–534, 1997.

[214] C. Schmid, R. Mohr, and C. Bauckhage, "Comparing and evaluating interest points," in *Proceedings of the International Conference on Computer Vision*, pp. 230–235, 1998.

[215] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors," *International Journal of Computer Vision*, vol. 37, no. 2, pp. 151–172, 2000.

[216] S. Se, T. Barfoot, and P. Jasiobedzki, "Visual motion estimation and terrain modeling for planetary rovers," in *Proceedings of the International Symposium on Artificial Intelligence for Robotics and Automation in Space*, 2005.

[217] N. Sebe, T. Gevers, J. van de Weijer, and S. Dijkstra, "Corner detectors for affine invariant salient regions: Is color important," in *Proceedings of International Conference on Image and Video Retrieval*, pp. 61–71, 2006.

[218] N. Sebe, Q. Tian, E. Loupias, M. Lew, and T. Huang, "Evaluation of salient point techniques," *Image and Vision Computing*, vol. 21, no. 13–14, pp. 1087–1095, 2003.

[219] R. Sedgewick, *Algorithms.* Addison–Wesley, Second Edition, 1988.

[220] E. Seemann, B. Leibe, K. Mikolajczyk, and B. Schiele, "An evaluation of local shape-based features for pedestrian detection," in *Proceedings of the British Machine Vision Conference*, pp. 11–20, 2005.

[221] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortex-like mechanisms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 411–426, 2007.

[222] T. Serre, L. Wolf, and T. Poggio, "Object recognition with features inspired by visual cortex," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2005.

[223] A. Sha'ashua and S. Ullman, "Structural saliency: The detection of globally salient structures using a locally connected network," in *Proceedings of the International Conference on Computer Vision*, pp. 321–327, 1988.

[224] M. A. Shah and R. Jain, "Detecting time-varying corners," *Computer Vision, Graphics and Image Processing*, vol. 28, pp. 345–355, 1984.

[225] E. Sharon, A. Brandt, and R. Basri, "Segmentation and boundary detection using multiscale intensity measurements," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 469–476, 2001.

[226] F. Shen and H. Wang, "Corner detection based on modified Hough transform," *Pattern Recognition Letters*, vol. 23, no. 8, pp. 1039–1049, 2002.

[227] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[228] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 593–600, 1994.

[229] A. Singh and M. Shneier, "Gray-level corner detection a generalization and a robust real time implementation," in *Proceedings of the Computer Vision Graphics Image Processing*, vol. 51, pp. 54–69, 1990.

[230] S. N. Sinha, J. M. Frahm, M. Pollefeys, and Y. Genc, "GPU-based video feature tracking and matching," in *EDGE, Workshop on Edge Computing Using New Commodity Architectures*, 2006.

[231] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proceedings of the International Conference on Computer Vision*, pp. 1470–1478, 2003.

[232] S. M. Smith and J. M. Brady, "SUSAN — A new approach to low level image processing," *International Journal of Computer Vision*, vol. 23, no. 34, pp. 45–78, 1997.

[233] K. Sohn, J. H. Kim, and W. E. Alexander, "A mean field annealing approach to robust corner detection," *IEEE Transactions on Systems Man Cybernetics Part B*, vol. 28, pp. 82–90, 1998.

[234] J. Sporring, M. Nielsen, L. Florack, and P. Johansen, *Gaussian Scale-Space Theory.* Springer-Verlag, 1997.

[235] M. Stark and B. Schiele, "How good are local features for classes of geometric objects," in *Proceedings of the International Conference on Computer Vision*, 2007.

[236] B. Super and W. Klarquist, "Patch-based stereo in a general binocular viewing geometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, pp. 247–252, 1997.

[237] M. Swain and D. Ballard, "Color indexing," *International Journal in Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[238] C.-H. Teh and R. T. Chin, "On the detection of dominant points on digital curves," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 859–872, 1989.

[239] P. Tissainayagam and D. Suter, "Assessing the performance of corner detectors for point feature tracking applications," *Image and Vision Computing*, vol. 22, no. 8, pp. 663–679, 2004.

[240] M. Trajkovic and M. Hedley, "Fast corner detection," *Image and Vision Computing*, vol. 16, no. 2, pp. 75–87, 1998.

[241] A. M. Treisman, "Features and objects: The fourteenth Berlett memorial lecture," *Journal of Experimantal Psychology*, vol. 40A, pp. 201–237, 1988.

[242] W. Triggs, "Detecting keypoints with stable position, orientation and scale under illumination changes," in *Proceedings of the European Conference on Computer Vision,* vol. 4, pp. 100–113, 2004.

[243] L. Trujillo and G. Olague, "Synthesis of interest point detectors through genetic programming," *Genetic and Evolutionary Computation*, pp. 887–894, 2006.

[244] D. M. Tsai, "Boundary-based corner detection using neural networks," *Pattern Recognition*, vol. 30, pp. 85–97, 1997.

[245] M. A. Turk and A. P. Pentland, "Eigenfaces for face recognition," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 586–591, 1991.

[246] T. Tuytelaars and C. Schmid, "Vector quantizing feature space with a regular lattice," in *Proceedings of the International Conference on Computer Vision*, 2007.

[247] T. Tuytelaars and L. Van Gool, "Content-based image retrieval based on local affinely invariant regions," in *International Conference on Visual Information Systems*, pp. 493–500, 1999.

[248] T. Tuytelaars and L. Van Gool, "Wide baseline stereo matching based on local, affinely invariant regions," in *Proceedings of the British Machine Vision Conference*, pp. 412–425, 2000.

[249] T. Tuytelaars and L. Van Gool, "Matching widely separated views based on affine invariant regions," *International Journal of Computer Vision*, vol. 1, no. 59, pp. 61–85, 2004.

[250] J. van de Weijer, T. Gevers, and A. D. Bagdanov, "Boosting color saliency in image feature detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 150–156, 2006.

[251] A. Vedaldi and S. Soatto, "Features for recognition: Viewpoint invariance for non-planar scenes," in *Proceedings of the International Conference on Computer Vision*, pp. 1474–1481, 2005.

[252] 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,* vol. 1, pp. 511–518, 2001.

[253] K. N. Walker, T. F. Cootes, and C. J. Taylor, "Locating salient object features," in *Proceedings of the British Machine Vision Conference*, 1998.

[254] H. Wang and M. Brady, "Real-time corner detection algorithm for motion estimation," *Image and Vision Computing*, vol. 13, no. 9, pp. 695–703, 1995.

[255] R. J. Watt, "Scanning from coarse to fine spatial scales in the human visual system after the onset of a stimulus," *Journal of Optical Society of America*, vol. 4, no. 10, pp. 2006–2021, 1987.

[256] S. Winder and M. Brown, "Learning local image descriptors," *International Conference on Computer Vision and Pattern Recognition*, 2007.

[257] J. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary," in *Proceedings of the International Conference on Computer Vision,* vol. 2, pp. 1800–1807, 2005.

[258] A. P. Witkin, "Scale-space filtering," in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 1019–1023, 1983.

[259] M. Worring and A. W. M. Smeulders, "Digital curvature estimation," *CVGIP: Image Understanding*, vol. 58, no. 3, pp. 366–382, 1993.

[260] R. P. Wurtz and T. Lourens, "Corner detection in color images through a multiscale combination of end-stopped cortical cells," *Image and Vision Computing*, vol. 18, no. 6–7, pp. 531–541, 2000.

[261] X. Zhang, R. Haralick, and V. Ramesh, "Corner detection using the map technique," in *Proceedings of the International Conference on Pattern Recognition*, vol. 1, pp. 549–552, 1994.

[262] X. Zhang and D. Zhao, "A morphological algorithm for detecting dominant points on digital curves," *SPIE Proceedings, Nonlinear Image Processing 2424*, pp. 372–383, 1995.

[263] Z. Zheng, H. Wang, and E. Teoh, "Analysis of gray level corner detection," *Pattern Recognition Letters*, vol. 20, pp. 149–162, 1999.

[264] P. Zhu and P. M. Chirlian, "On critical point detection of digital shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 737–748, 1995.

[265] M. Zuliani, C. Kenney, and B. S. Manjunath, “A mathematical comparison of point detectors,” in *Proceedings of the Computer Vision and Pattern Recognition Workshop*, p. 172, 2004.

[266] O. A. Zuniga and R. Haralick, “Corner detection using the facet model,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 30–37, 1983.