Neurocomputing 149 (2015) 736–751

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A survey of recent advances in visual feature detection ☆

Yali Li [a,b], Shengjin Wang [a,b,*], Qi Tian [c], Xiaoqing Ding [a,b]

[a] State Key Laboratory of Intelligent Technology and Systems, PR China
[b] Department of Electronic Engineering, Tsinghua University, Beijing 100084, PR China
[c] Department of Computer Science, The University of Texas at San Antonio (UTSA), United States

ARTICLE INFO

Article history:
Received 26 January 2014
Received in revised form
16 July 2014
Accepted 3 August 2014
Communicated by D. Tao
Available online 11 August 2014

Keywords:
Visual feature detection
Machine learning
Edge detection
Corner detection
Blob detection

ABSTRACT

Feature detection is a fundamental and important problem in computer vision and image processing. It is a low-level processing step which serves as the essential part for computer vision based applications. The goal of this paper is to present a survey of recent progress and advances in visual feature detection. Firstly we describe the relations among edges, corners and blobs from the psychological view. Secondly we classify the algorithms in detecting edges, corners and blobs into different categories and provide detailed descriptions for representative recent algorithms in each category. Considering that machine learning becomes more involved in visual feature detection, we put more emphasis on machine learning based feature detection methods. Thirdly, evaluation standards and databases are also introduced. Through this survey we would like to present the recent progress in visual feature detection and identify future trends as well as challenges.

1. Introduction

Visual features refer to interest image structures and primitives. They are very important within the field of computer vision and image processing. Feature detection is referred as the identification of interested image primitives (e.g. points, lines/curves, and regions), for the purpose of highlighting salient visual cues in digital images. It is a low-level processing step with pixel intensities as the input and image structures indicating different characteristic properties as the output. A variety of visual features are widely researched and applied in computer vision based applications such as object recognition, content-based image retrieval (CBIR), visual tracking [1], and wide baseline matching [2,3]. Although the application scopes of visual features are widely different, it is the ultimate goal to extract features with high stability effectively and efficiently.

The main challenge in computer vision is the semantic gap between high-level concepts and low-level visual cues. Descriptive and discriminant features are important to bridge the semantic gap, therefore can impact the system performance significantly. Although many efforts have been devoted to feature detection, the challenges still exist. They are mainly caused by the divergence of imaging conditions. Generally, the difficulties in feature detection are caused by the changes in scale, viewpoint, illumination, image quality, etc. A high-performance feature detector should show robustness to changing imaging conditions as well as satisfy human interests. Besides, the computational efficiency needs to be considered in real-time applications. Although there are existing surveys on the methods of feature detection, they focus on visual features of single kind (e.g., edge detection [4–6], interest point detection [3,7]) and lack of the relations among different visual features. Besides, some of the methods introduced in existing surveys are out of date. Also we note the new trend that machine learning algorithms are more extensively applied in visual feature detection. In this paper, we survey the recent advances and progress in feature detection. The motivation of this survey includes

(1) We aim to present the emerging development of feature detection techniques, especially the machine learning based feature detection methods.
(2) We also present the relations of different feature detection methods. Along with representative existing techniques, we draw the trends on feature detection and put emphasis on future challenges.

The remainder part of the paper is organized as follows. Section 2 introduces the basic definitions on visual features. Section 3 presents

☆This document is a collaborative effort. The work is supported by the National High Technology Research and Development Program of China (863 program) under Grant No. 2012AA011004 and the National Science and Technology Support Program under Grant No. 2013BAK02B04.

* Corresponding author.
E-mail addresses: liyali@ocrserv.ee.tsinghua.edu.cn (Y. Li), wgsgj@tsinghua.edu.cn (S. Wang), qitian@cs.utsa.edu (Q. Tian), dxq@ocrserv.ee.tsinghua.edu.cn (X. Ding).

http://dx.doi.org/10.1016/j.neucom.2014.08.003
0925-2312/

the recent advances on visual feature detection methods. In Section 4 we present the evaluation and typical databases for feature detection. The summaries and discussions are presented in Section 5.

2. The classification of feature detection methods

Visual features are much related to human perceptual organization. The psychological research of Gesalt laws indicates that human vision system is prone to group low-level image components. Human vision system groups and organizes visual stimuli according to Gesalt factors such as proximity, similarity, continuity, and closure. Since computer vision is to simulate human visual perception with cameras and computers, visual feature detection finds inspiration from human visual perception. There are several visual features applied in computer vision tasks are biologically inspired [8,9]. Visual features bridge from image pixels to computer vision tasks. Primitive features such as edges, contours, corners and regions are much related to human visual perception. To better describe the recent progress in feature detection methods, we firstly clarify the related concepts (Fig. 1).

1. *Edge* refers to pixel at which the image intensities change abruptly. Image pixels are discontinuous at different sides of edges.
2. *Contour/boundary* has ambiguous definitions. Since we focus on low-level features, we refer them as the intersecting lines/curves of different segmented regions.
3. *Corner* refers to the point at which two different edge directions occur in the local neighborhood. It is the intersection of two connected contour lines.
4. *Region* refers to a closed set of connected points. Nearby and similar pixels are grouped together to compose the interest region.

It is noteworthy that there are natural and tight connections between the above-mentioned definitions. That is, contour/boundary can be obtained by tracking and connecting neighboring edges. Corners are the intersecting points of straight edge lines. The intersection curves between different regions consist into boundaries. We follow the traditional categorization and refer the *edges*, *corners*, *regions* as the important visual features. The visual feature detection methods are classified as edge detection, corner detection and blob detection (i.e., interest point/region detection). Here blob refers to the local regions of interest. The categorization of visual feature detection methods are further illustrated in Fig. 2. And the representative methods are listed in Table 1. Edge detection is briefly divided as *differentiation based* and *learning based* methods. The outputs of edge gradients are often used as the inputs of *learning based* methods. As for corner detection, the methods can be divided as *gradient based*, *template based* and *contour based* methods. *Contour based* corner detection is based on contour/boundary detection. Blob detection is divided as interest point and region detection. Several interest point detection methods are constructed based on the multi-scale analysis of corner detection. Interest region detection is much related to segmentation techniques. Boundary based interest region detection is based on contour/boundary detection. We mainly focus on the recent advances on visual feature detection. A chronicle table of visual features emerges in recent years is given in Fig. 3. The traditional feature detection methods before 2005 are firstly listed. The newly emerged representative features are sorted and labeled by

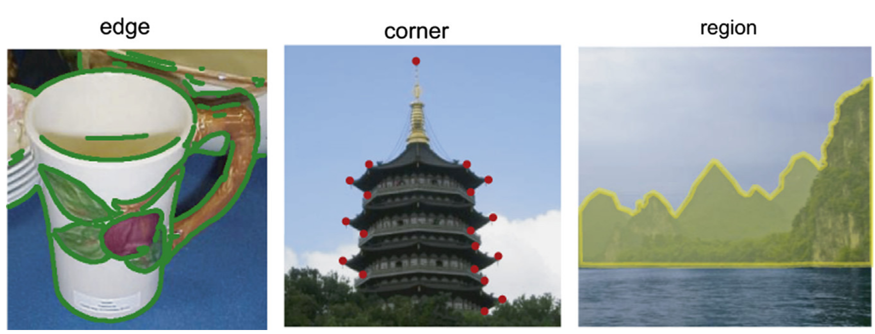

Fig. 1. The definitions of visual features in computer vision.

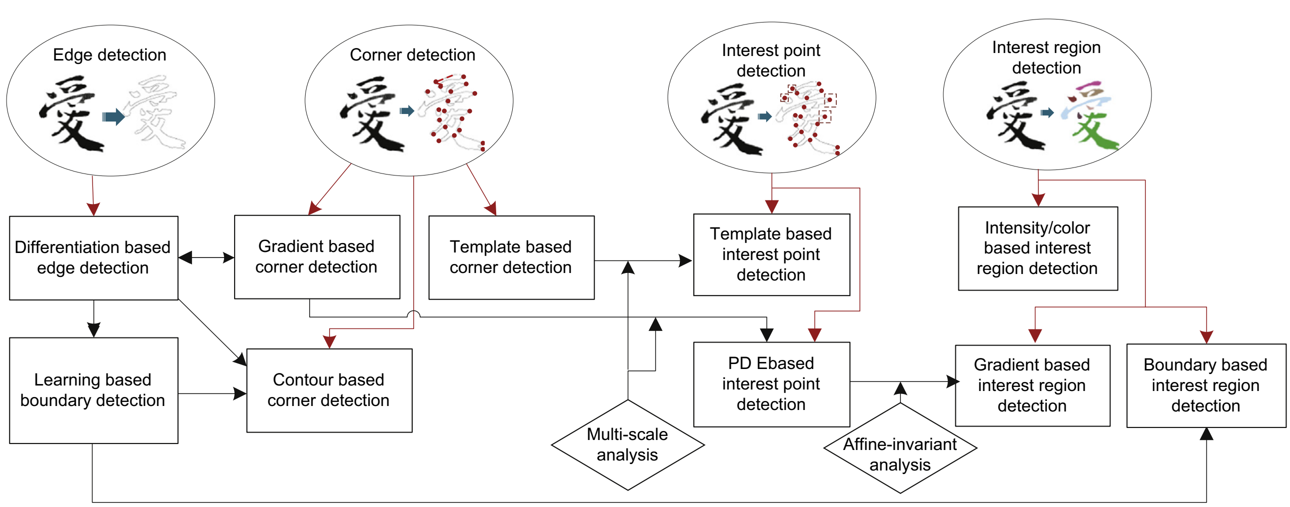

Fig. 2. The classification of visual feature detection methods. The connections of different categories are also labeled.

Table 1
The classification of covered representative feature detection methods.

Category	Classification	Methods
Edge detection	Differentiation based	Sobel, Prewitt, Roberts-cross, Laplacian of Gaussian, Oriented Energy (OE) [10], Canny edge detector [11,12], D-ISEF [13], Color boundary [14]
	Learning based	Pb [15], MS-Pb [16],gPb [17,18], tPb [19], NMX [20], BEL [21], DSC [22], Sketch Tokens [23], SCG [24], SE [25]
Corner detection	Gradient based	Harris detector [26], KLT [27], Shi-Tomasi detector [28], LOCOCO [29], S-LOCOCO [30]
	Template based	SUSAN [31],FAST [32], FAST-ER [33], AGAST [34]
	Contour based	DoG-curve [35], ANDD [36], Hyperbola fitting [37], ACJ [38]
Blob detection		
Interest point	PDE based	LoG, DoG, DoH, Hessian–Laplacian [7], SIFT [39], SURF [40], Cer-SURF [41], DART [42], Rank-SIFT [43], RLOG [44], MO-GP [45], KAZE [46], A-KAZE [47], WADE [48]
	Template based	ORB [49], BRISK [50], FREAK [51]
Interest region	Segmentation based	MSER [52], IBR [2], EBR [2], Salient region [53], PCBR [54], Beta-stable feature [55], MFD [56], MSCR [57], FLOG [58], BPLR [59]

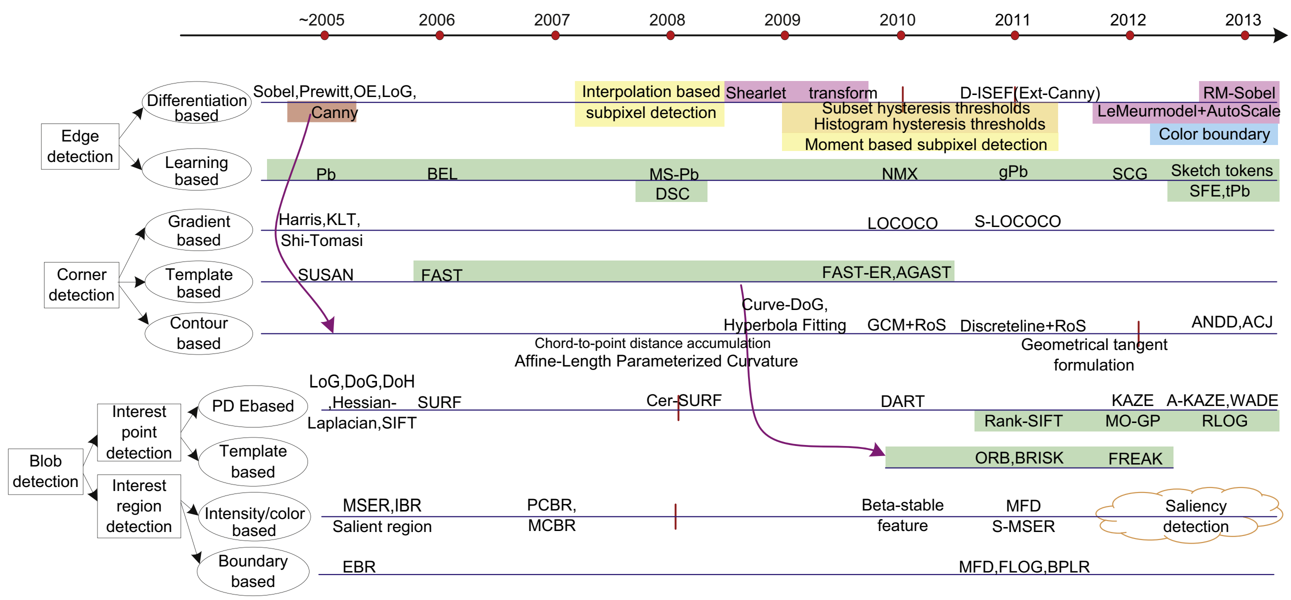

Fig. 3. Recent representative advances in visual feature detection.

the year information. We would introduce the typical feature detection methods in details in the following part of this paper.

3. Feature detection methods

In this section, we introduce the detection methods of edges, corners and blobs in details. We mainly focus on the recent progress and advances. The relationship among the detection methods for different kinds of visual features is also presented.

3.1. Edge detection

Edge refers to sharp changes in image brightness. Differential operation is used to capture the strength and position of discontinuities in image brightness. Contour/boundary can be viewed as the generalized definition of edge which indicates the intersection of different regions. Contour/boundary has been playing the important role in image interpretation. Recently efforts have been devoted to multi-resolution edge analysis, sub-pixel edge detection and hysteresis thresholding. Besides, along with the extraction of multiple low-level information, statistical machine learning is introduced into contour and boundary detection.

3.1.1. Differentiation based edge detection and the progress

Classical edge detection aims to capture discontinuities in image brightness. Differentiation based filters are convolved to identify edge points. First-order differentiation based gradient operators appear in pairs (e.g., Prewitt, Sobel, as shown in Fig. 4(a and b)). By those operators, gradients at different orientations are computed. Local maximas of gradient magnitudes are recorded as edges. Second-order differentiation filters such as Laplacian of Gaussian (LoG) (Fig. 4(c)) find zero-crossings as the edge positions. Gaussian smoothing is necessary since differential operation is sensitive to noise. Directional differentiations such as Oriented Energy (OE) [10] adopt a batch of filters at different orientations to obtain brightness changes. An early survey on differentiation based edge detection is given in [4]. Differentiation based edge detection is very simple but sensitive to noise. Currently they are seldom independently used to identify edges. Yet the filtered responses of differential filters are still widely used as low-level image cues to construct more reliable and informative features [102].

Canny edge detector [11] is based on the computational theory of edge detection. The edge detection is modeled as an optimization problem with three criteria such as good detection, good localization and single-pixel response. The steps of Canny edge detection are filtering, hysteresis thresholding, edge tracking and non-maximal suppression. Edge responses are firstly obtained by filtering with gradient operators. The edges are then traced and determined by hysteresis thresholds. Only the pixels with maximal a

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

b

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

c

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Fig. 4. Differentiation based edge detection. They are characterized by gradient operators.

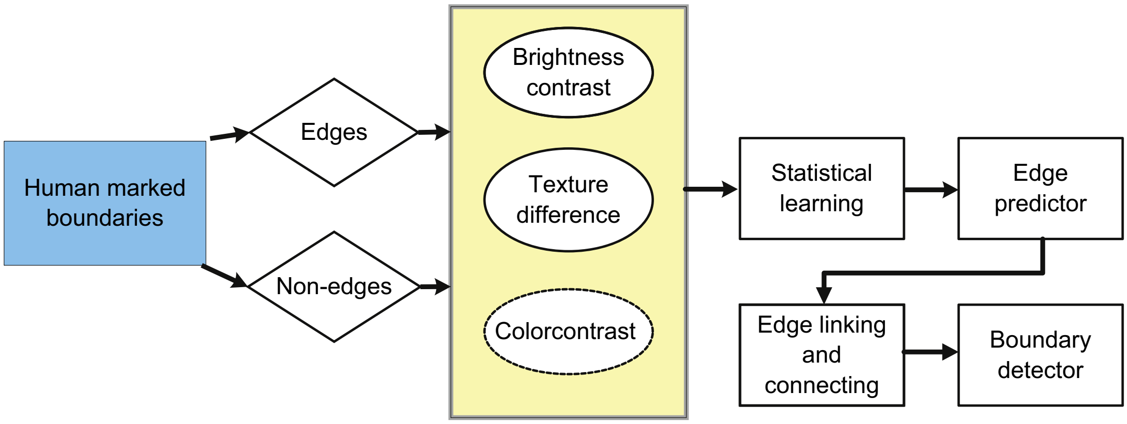

Fig. 5. A framework for statistical learning based boundary detection.

magnitude in the gradient direction can be recorded as edge points. Canny edge detector still outperforms several new detectors and is still widely applied today. An extended Canny edge detection theory has been proposed recently in [13], with the objective to amend the computational theory of Canny in order to generate theoretically finite edge curves. Besides, a distributed Canny edge detector for FPGA implementation is presented in [12].

Differentiation based edge detection proceeds with the purpose of improving detection performance, especially in natural images. The recent advances mainly concentrate on multi-resolution analysis, hysteresis thresholding and sub-pixel edge detection. Multi-resolution edge detection is inspired by the fact edge responses are scale-relevant and human vision is multi-resolutional. It aims to integrate the edge responses at different scales. Gaussian smoothing and coarse-to-fine edge tracking are combined to detect multi-scale edges. Wavelet analysis named as Shearlet transform [60] creates the multi-scale directional image representation, which are used to localize discontinuities in different scales. Sobel operators are performed in different Gaussian-smoothed images. Edge pixels are matched in descending scales to be integrated into the multi-scale edges [61]. A human visual attention inspired model is to automatically select the most suitable scale for edge detection [62].

Threshold selection is essential to binarize edge responses. Hysteresis thresholding which is used in Canny edge detection helps generate connected edge curves. Histogram of gradient magnitude is used within chosen regions to determine the high and low thresholds [63]. A subset of edge point candidates is constructed to determine the hysteresis thresholds [64]. Sub-pixel edge detection is used to increase the localization accuracy. Interpolation is proven to be effective in sub-pixel edge localization [65]. Moments are computed in region of interest to construct the model for sub-pixel accuracy [66,67]. Besides, color edge detection also attracts attention. Color model is combined with edge determination techniques and the color-opponent mechanism originates from human visual system is applied for color boundary detection [14].

3.1.2. Learning based boundary detection

Differentiation based edge detection focuses on abrupt changes in image brightness. It produces wild edge responses in textured regions. Internal edges caused by textures need to be suppressed to obtain the boundaries between different regions. Recent progress is much promoted by the Berkeley Segmentation Dataset and Benchmark (BSDS).[1] Natural images and attached human labeled boundaries are included in the dataset. Boundaries are manually marked and provided in the training and validation set. Edge detection is modeled as a machine learning based framework to discriminate edge points from smooth regions. The typical framework is presented in Fig. 5. Multiple low-level image cues are extracted and combined into the model for edge response prediction. We list the typical statistical learning based edge detection methods as well as low-level cues and applied statistical learning algorithms in Table 2.

Pb (probability-of-boundary) [15] edge detection is based on the extraction of multiple local cues including brightness gradient, texture gradient and color gradient. Logistic regression is applied to combine the χ-distance of those multiple cues and learn the discriminative model for edge response prediction. Since edge responses are scale relevant, multi-scale edge detectors based on Pb [16,17] are proposed. Similar to differentiation based edge detection, the combination and localization of edges in different scales are two important issues to be solved. MS-Pb [16] integrates the multi-scale edge responses of Pb [15], additional localization and relative contrast information to determine multi-scale edges. The localization cues represent the distance from pixels to the closest peak Pb responses at respective scales. And the relative contrast cues indicate the normalized edge responses in local regions. gPb (global Pb) [17,18] linearly combines Pb edge responses in three scales and the global information into contour detection. Multi-scale image cues are combined into an affinity matrix which defines the similarity

[1] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/

Table 2
Typical statistical learning based edge detection and the corresponding statistical learning algorithms.

Methods	Cues	Statistical learning
Pb [15]	Gradient, texture, color	Logistic regression
MS-Pb [16]	Pb edge responses, localization, relative contrast	Logistic regression
gPb [18]	Gradient, texture, color	Spectral clustering
tPb [19]	pair-wise texture variations	Linear combination
NMX [20]	Pb edge responses, SIFT [39]	*F*-measure AnyBoost
Sketch Tokens [23]	Color, gradient, oriented gradient	Random forests
BEL [21]	Histograms of DoG/DooG, Haar	Boosting
SFE [25]	Color gradients, pairwise-difference	Structured random forests
DSC [22]	Reconstruction error of discriminative sparse coding	Sparse coding, logistic regression
SCG [24]	Sparse codes	Sparse coding, SVM

between pixels. Spectral clustering is used to globally compute the eigenvectors of the affinity matrix which correspond to the contour information. tPb(texture based Pb) [19] uses the average of texture variations in randomly placed pairs of windows to estimate salient edge responses. Pb edge features are fed into the AnyBoost classifier whose optimization criterion is based on the approximation of *F*-measure (i.e., *F*-measure Boosting) [20] for boundary detection.

Except for contrasts in prior defined channels, other methods apply existing feature descriptors (e.g., Haar wavelets [68], SIFT [39]) as dictionary words and feed them into classifiers for edge responses prediction. BEL (Boosted edge learning) [21] is based on the probability Boosting by decision trees. Generic features such as histograms of DoG responses and Haar wavelets are generated at different locations and scales. Those generic features are sequentially selected by Boosting decision trees to construct the discriminative model for edge determination. Besides, structured forest based edge detection (SFE) [25] is based on the learning of randomized decision trees. The inputs are color and gradient channels as well as pair-wise differences. Each structure forest labels edge pixels inside the patch. The final edge responses are the aggregation of random forests. Both BEL [21] and SFE [25] based edge detection share the advantage of relatively low computation cost. Besides, automatic generated the dictionary words serve as the input of classifiers for edge probability determination. For instance, sparse coding representation is applied to generating dictionary words for candidate edge responses. In [22], a discriminative sparse coding algorithm is used to learn the sparse representation of images. An extra linear classifier is trained to combine multi-scale reconstruction errors of sparse coding and obtain the edge responses. SCG (sparse code gradient) detection [24] uses sparse coding to automatically produce generic features. Sparse codes serve as the input local cues and support vector machine (SVM) classification is used to learn the model to discriminate edges from non-edges.

Edge points are low-level features. Edge tracking aims to link the isolated and identified edge points. Furthermore, contour grouping is introduced to obtain continuous semantic curves and boundaries of regions in closed form. Contour grouping can also be viewed as to improve the boundary detection performance with global information. Spatial relations of dispersed edge fragments need to be considered in contour grouping. Gesalt factors such as proximity and continuity are used in the contour grouping model. Graph models have been widely applied into the formulation of grouping factors. Untangling cycles [69] exploit the topological relations of contour fragments. Normalized contour saliency is explored and contour grouping is formulated as a discrete optimization issue in [20]. Markov random Field and Conditional Random Field (CRF) models [70,71] which are widely applied segmentation techniques to capture interactions among neighboring contours and build up contour connections. Sequential Bayesian framework is applied in grouping edge fragments. In [72], the primal edge fragments are classified as edgelets by the shape information. The priors and the transitions over multiple types of edgelets are learnt in the offline step. Particle filtering framework is constructed for contour elements grouping. Besides, a sequential labeling of edges for boundary detection is presented in [19].

3.1.3. Discussions

Edge detection has always been important in image processing and computer vision. Classical differentiation based edge detection identifies pixel locations with brightness changes by gradient operators. These gradient operators are still widely used nowadays, as low-level pre-processing for further mid-level image representation. Canny edge detection is the popular choice for applications in which connected single-response edges are produced. Three important issues are concerned in recent differentiation based edge detection. They are sub-pixel edge detection, multi-scale edge analysis and hysteresis thresholding. The first aims to improve the localization accuracy and the last two aims to lift up the detection accuracy. Differentiation based edge detection methods only focus on the contrast in intensity, therefore they universally suffer from the disadvantage of noise-sensitivity. The edge responses are wild in texture regions. Using gradient information only also might lead to internal edges which might be noise for image interpretation.

Another noteworthy progress is the learning based boundary detection. It is promoted by the emergence of Berkeley Segmentation Dataset and Benchmark. The edge detection is modeled as a classification problem to distinguish boundary points. Statistical learning algorithms and multiple image cues are used to construct the model for edge response determination. The learning based methods can be further divided by the representation of multiple cues. Following by Pb edge detector [15], the distance functions are formulated to measure the pair-wise region differences in color, texture, brightness channels. Classification algorithms are used to combine the multiple distance measures. Since the distance measures are quite directly correlated with boundary points, simple linear classification can also lead to good performance. Existing experiments show that by averaging the textural distances in pairwise subwindows, comparable performance can also be achieved [19]. The other learning based methods feed automatic generated dictionary words (i.e., sparse codes, densely defined Haar features, pair-wise comparison features) into classifiers. In some cases classification algorithms such as Boosting and random forest are needed to select effective dictionary words. The computational burden can be decreased by selecting simple calculated dictionary words.

Compared with differentiation based methods, learning based methods can suppress the internal edges caused by textures.

The detected edges by learning are closer to human visual perception and more related with semantic meanings such as object boundaries. However, the computational cost is quite high. Multi-scale analysis is a shared issue in both differentiation and learning based edge detection. Yet the sub-pixel localization is only researched in differentiation based methods. In learning based methods, instead of pixels, subregions are basic elements for difference computing and analysis. The integration of information in neighboring subregions limits the localization accuracy. Differentiation based edge detection is favored in constructing mid-level feature representation from image subregions. It is now integrated into the analysis of image visual properties. Meanwhile, learning based edge detection with multiple cues can obtain the boundary fragment, which is much related with the image segmentation.

Edge tracking and contour grouping are the following steps to generate connected edge lines/curves. Edge tracking aims to connect isolated edge points into edge lines/curves. And contour grouping aims to obtain the semantic boundaries between different objects. The latter improves edge detection performance with global information. Graph model based learning algorithms are used to learn spatial and co-occurrence relations of edge fragments. The connected and grouped edges are quite useful for object recognition and image understanding. Shape information can be directly acquired from the analysis of connected edges. Therefore the responses of Canny and Pb edge detectors are widely used as the basis of object class recognition with shapes, in which the detected responses are viewed as the fragments of object boundaries. The accuracy and precision are important measurements in evaluating the edge detection performance.

3.2. Corner detection

Corner is defined as the intersecting point of two connected straight edge lines. In image understanding, it mathematically refers to the point at which two dominant and different gradient orientations exist. Wealthy information can be obtained at the neighborhood of corners. Compared with edges, corners are unique in local regions, which are favored for wide baseline matching. The definition of corners is scale-relevant. Corner detection along with multi-scale analysis is a straightforward and important way to identify interest points. Reversely, corner can also be viewed as interest point at a fixed scale. Generally, corner detection methods can be further divided into three classes. Classical *gradient based* corner detection is based on gradient calculation. *Template based* corner detection is based on the comparison of pixels. In recent years, templates are combined with machine learning techniques (i.e., decision trees) for fast corner detection. *Contour based* detection is based on the results of contour and boundary detection. It relies on the prediction of edge responses to identify corners.

3.2.1. Gradient based corner detection

Most of the corner detection methods in early literature are based on gradients computation. Here we take the representative Harris corner detector [26] as an example to illustrate the idea of gradient based corner detection. Suppose there is an image patch located at (u,v) as $I(u,v)$. Shift the image patch by (x,y), another image patch as $I(u+x,v+y)$ is obtained. The weighted sum of squared difference (SSD) between the shifted and original window can be computed as

$$E(x,y) = \sum_{u,v} w(u,v)(I(u+x,v+y)-I(u,v))^2. \tag{1}$$

where $w(u,v)$ is the window function. It can be set as the average or Gaussian function. $I(u+x,v+y)$ can be approximated by Taylor expansion as in Eq. (2):

$$I(u+x,v+y) \approx I(u,v)+I_x(u,v)x+I_y(u,v)y. \tag{2}$$

in which I_x and I_y are gradients in x and y directions. Furthermore, $E(x,y)$ can be rewritten as

$$E(x,y) \approx \sum_{u,v} w(u,v)(I_x(u,v)x+I_y(u,v)y)^2 = (x,y)M(x,y)^T. \tag{3}$$

where M is a 2×2 matrix. It can be computed from first-order gradients as

$$M = \sum_{u,v} w(u,v)\begin{bmatrix} I_x^2(u,v) & I_x(u,v)I_y(u,v) \\ I_x(u,v)I_y(u,v) & I_y^2(u,v) \end{bmatrix}. \tag{4}$$

The eigenvalues of M as λ_1 and λ_2 are used to distinguish corners. If λ_1 and λ_2 are both large, $E(x,y)$ increases significantly in all directions. That indicates $I(u,v)$ is around a corner. Besides, if $\lambda_1 \gg \lambda_2$, $I(u,v)$ is near edges. Harris cornerness measurement is constructed from determinant and trace of M. It is further used to distinguish corners, edges and smooth regions (Fig. 6).

It can be seen from the above that Harris corner detection is based on the auto-correlation of gradients on shifting windows. There are other gradient based corner detection methods proposed in early papers, such as KLT [27] and Shi-Tomasi corner detector [28]. The main difference lies in the cornerness measurement function. Since the calculation of gradients is sensitive to noise, gradient based corner detection suffers from the disadvantages of noise-sensitivity. Besides, the matrix for measurement function needs to be computed inside the window, which makes the computational complexity quite high. That is another drawback of traditional gradient based corner detection. To tackle with the high complexity, recently efforts have been devoted on the approximation of gradient based cornerness measurements. Low-complexity corner detector is proposed based on the classical corner detectors, such as LOCOCO [29]. It is based on Harris and KLT cornerness measurements. Firstly, box kernels are used to approximate the first-order Gaussian derivative kernel. Second, gradient based integral images are borrowed to compute the cornerness responses in overlapping windows in a fast way. Finally, an efficient non-maximal suppression is proposed based on the Quick-sort algorithm for further time saving. A sub-pixel version of LOCOCO as S-LOCOCO based on interpolation is recently proposed in [30].

3.2.2. Template based corner detection

Template based corner detection finds corners by comparing the intensity of surrounding pixels with that of center pixels. Templates are firstly defined and placed around the center pixels, as illustrated in Fig. 7. The cornerness measurement function is devised from the relations of surrounding/centering pixel intensities. In traditional SUSAN (Smallest Univalue Segment Assimilating Nucleus) [31], every pixel inside the circular mask is compared with the center pixel and the intensity difference is recorded. USAN measurement is defined as the number of pixels with absolute intensity difference smaller than a threshold. Points with smallest USAN value are recorded as corners. The computational cost for template based corner detection is mainly caused by the multiple times of comparison, which is relatively lower than that of gradient based methods.

In recent years, machine learning algorithms especially decision trees are involved to accelerate template based corner detection. FAST (Features from Accelerated Segment Test) [32] is based on a circular template of diameter 3.4 pixels contains 16 pixels. A point is considered as a corner only if there are at least S contiguous pixels in the circle which are brighter or darker than the value determined by the center pixel intensity and a threshold t.

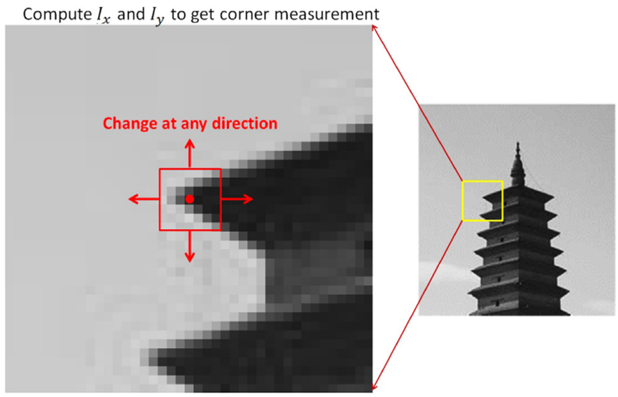

Fig. 6. Gradient-based corner detection.

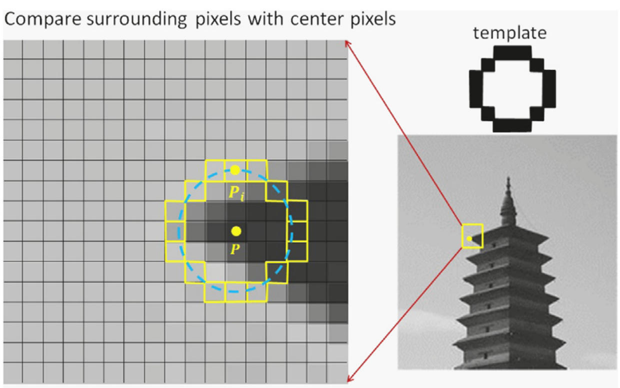

Fig. 7. Template-based corner detection.

Suppose the center is $\mathbf{p}_0$ and the pixel intensity is $I(\mathbf{p}_0)$. There must be at least S connected pixels are brighter than $I(\mathbf{p}_0)+t$ or darker than $I(\mathbf{p}_0)-t$ when $\mathbf{p}_0$ is regarded as a corner. A decision tree is learnt to determine the order of pixels for comparison, for time saving. FAST-ER [33] increases the thickness of circular template in order to increase the stability of detected corners. AGAST (Adaptive and Generic Accelerated Segment Test) [34] is another kind of FAST derivations. It applies backward induction to construct an optimal decision tree for speed improvement. Besides, a collection of different decision trees are trained with different sets of specified train images. The collection of decision trees makes AGAST more generic for different environments. Templates are quite important in determining corners. Circular templates are chosen since they are isotropic. The interpolation is needed to calculate pixel intensity in sub-pixel level for high accuracy. In general, a thicker template increases both the robustness and computational cost. Besides, although the application of machine learning helps reduce the computational cost of corner detection, it also might cause database-dependent problems.

3.2.3. Contour based corner detection

Corner is defined as the intersecting points of two adjacent straight edge lines. There are methods find corners based on contour/boundary detection. These methods aim to find the points with maximum curvature in the planar curves which compose of edges. Traditional kinds of contour based corners are specified in binary edge maps, which are the achieved results of edge detection and tracking (as introduced in Section 3.1). Curve smoothing and curvature estimation are two essential steps. Smoothing curves helps reduce the noise caused by quantized point locations

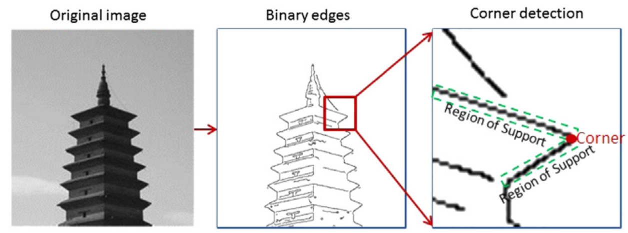

Fig. 8. Contour based corner detection (RoS based).

and Gaussian filters are the most widely used smoothing function. Difference of Gaussian (DoG) filters are convolved with curves points to find curvature corner points [73]. Anisotropic directional derivative (ANDD) filters based on Gaussian kernels [36] are applied to preserve the curves while reduce curve representation noise. Region of support (RoS) is a region smoothing way (as shown in Fig. 8). A RoS is a bounded zone of curves and it is used in [35,74]. Recent curvature estimation is obtained by Affine-Length Parameterized Curvature [75], Chord-to-point distance accumulation [75], geometrical tangent formulation [76], determinant of gradient correlation matrices [35]. A recent survey on contour based corner detection with binary edge maps is presented in [77]. Contour based corners are more applied in shape analysis and shape based image compression, rather than wide baseline matching.

There have been ambiguous bounds between the definition of corners and junctions, especially in contour based detection. The latter is often assigned with more specific information of intersecting contours. The crossed contours can be connected with *T*-, *L*-, *X*-junctions. Classical researches on contour based corner and junction detection focus on the processing of binary edges. However, it might be insufficient for applications in natural images since the binarization causes information loss. The edge responses rather than binarized edge maps are applied in recent proposed methods. Templates are firstly defined around the center point in edge images. Different from templates defined in FAST [32], AGAST [34], the templates in junction detection are more similar to RoS. They are defined with wedge shapes to serve as the branches of curves. Fitting steps are needed to determine the intensity changing information in branches. The shapes of junctions as well as the cornerness measurements are accomplished by the strength of edge responses in branches. In [37], the edge responses are computed and used to fit the hyperbolic polynomial curves. Fast corner detection is accomplished by thresholding the algebraic features from the fitted curve based shape models. In [38], fan-shaped branches are applied. The normalized gradients and discrete orientations are used to compute the edge strength in branches. Probabilistic principles are set up for robust junction detection. The accuracy of corner localization relies on edge detection. Along with the improvement in edge detection performance caused by multi-cue integration and machine learning, the contour based corner detection has been promoted. A unified framework for contour and junction detection based on multi-cue combination and statistical learning is presented in [17]. Similar to learning based boundary detection, the integration of multiple cues helps inhibit the corners in texture regions.

3.2.4. Discussions

Compared with edges, corners are stable and unique in local image regions. The corners are defined as the point where two different gradient directions exist. Intuitionally, classical gradient based corner detection methods come out. However, it is quite time-consuming and noise-sensitive. In contrast, template based corner detection is much faster since no derivative operation is employed. It is based on the comparison of pixel intensities in circular templates. Recent progress is characterized by the application of decision tree classification, which significantly accelerates the corner detection speed. Besides of time saving, the detected number of corners by template based methods such as FAST [32], FASTER [33], AGAST [34] is larger than that of gradient based methods. Since more points lead to more reliable matching, the larger number of detected corners is preferred in wide baseline matching. The comparison over pixels also tolerates the illumination changes. However, the number of detected FAST corners is not so stable under different imaging conditions [3]. Another disadvantage of template based methods is the lack of effective and precise cornerness measurements. The discrete cornerness measures are hard to satisfy the requirement of non-maximal suppression. Besides, the learning procedure might cause database-dependent problems thus the generalization performance of machine learning based methods needs to be improved. Contour based corner detection is inspired by the natural connections between edge fragments and corners. It is much different from the gradient and template based methods in detection framework and application area. Contour based detection methods depend on the contours acquired by edge detection and linking. The detected corners and junctions are more applied in shape based region approximation and representation. Rather than binary identified contour points, recently gradient magnitudes and orientations are more involved in contour based corner detection. Cornerness measurement is computed in wedge shaped templates. The classification in templates can further discriminate different kinds of junctions.

Corners are much related with edges and blobs. Firstly, corners are the points where at least two dominant edge orientations exist. The traditional cornerness measurements such as Harris measures [26], UASN [31] can also be used to identify edges. Contour based corner detection is based on edge response determination. It needs edge detection and tracking as the pre-processing steps. There are recent methods [17] which combine the tasks of contour and corner detection into a whole framework. Secondly, corners can be viewed as points of interest at a fixed scale. They can be viewed as an important kinds of blobs. Combined with pyramid construction and multi-scale analysis, interest points dispersed in different scales can be localized. The classical gradient based corner detection leads to gradient based features such as SIFT [39], SURF [40]. They are of high time and storage burden. In contrast, template based corner detection leads to binary features. With the emerges of binary descriptors such as BRIEF [78,79], corner detection based on binary decision trees are integrated into the construction of binary features such as ORB [49], BRISK [50]. ORB borrows Harris cornerness measurement to rank the detected multi-scale corners. An evaluation paper of binary features is given in [80]. The attached binary description helps reduce the storage burden and time cost for matching. Therefore binary features are time and storage saving. The low computation and storage burden make corners detected by decision trees more preferred in large-scale vision based applications, such as image retrieval. Both gradient and template based corners are widely researched and applied in wide baseline matching and stereo reconstruction. Besides, corners and interest points are often combined with feature description for high-level image interpretation.

3.3. Blob detection

Blob is defined as a region inside which the pixels are considered to be similar to each other, meanwhile be different from the surrounding neighborhoods. The definition of blobs is based on constancy of interest properties, thus blob detection is further referred as the identification of interest point (keypoint)/interest region (keyregion). Blobs are represented by regions of regular/irregular shapes. We refer interest points as local extremas in scale-location spaces, which furthermore denote regular circular or square regions. Interest regions are referred as segmented regions (in most cases irregular) with defined constancy. Interest point detection aims to find local extremas in pyramid spaces and interest region detection aims to identify regions with constancy by segmentation techniques. The stability is always the favored property for blob detection.

3.3.1. Interest point detection

Interest point can provide informative representation for digital images. It refers to local extrema in 3-dimensional scale spaces with locations and scale as axes, as shown in Fig. 9. Thus interest point can be mathematically denoted as (x, y, σ). Here (x,y) indicates the location and σ indicates the scale. Corner can be viewed as interest point at the fixed scale. Furthermore, feature descriptors can be obtained inside square or circular regions centered at (x,y) with the size determined by σ [81]. A variety of interest point detection methods are proposed and the existing literature for evaluation are given in [7,3]. Classical methods include Laplacian of Gaussian (LoG), difference of Gaussian (DoG) and Hessian–Laplacian [7] are based on Gaussian pyramid construction. Gaussian scale-space kernel is defined as

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp^{-(x^2+y^2)/2\sigma^2}. \tag{5}$$

Gaussian pyramid is constructed by increasing σ. Suppose

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y). \tag{6}$$

where $*$ indicates the convolution operation. LoG is based on the Laplacian of Gaussian filtered scale space. Each layer of LoG pyramid is defined as $\nabla^2 L = L_{xx} + L_{yy}$, in which L_{xx}, L_{yy} are the second partial derivatives. Unlike LoG, DoG layers are obtained by the difference of two nearby Gaussian smoothed layers, without the computation of second partial derivatives. DoG can be seen as the approximation of LoG with low computational cost. The DoG function is defined as

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y). \tag{7}$$

where k is a multiplicative factor. The local extremas of LoG and DoG pyramid are recorded as LoG and DoG interest points, respectively. Another classical interest point detection is based on the determinant of Hessian matrix (DoH). The Hessian matrix for the Gaussian smoothed images is

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix}. \tag{8}$$

where L_{xx},L_{xy},L_{yy} are the second partial derivatives. The scale-normalized determinant of Hessian matrix $\sigma^4 \det(H)$ is the measurement function for interest point detection. Hessian-Laplacian [7] combines LoG and DoH for interest point detection.

Interest point detection with DoG, DoH, Hessian–Laplacian is still widely used in recent computer vision algorithms. SIFT (Scale Invariant Feature Transform) [39] locates interest points with DoG pyramid and Hessian matrix. The local extremas in DoG pyramid are recorded as potential keypoints and a 3D quadratic function is to approximately locate the interpolated location of candidate keypoints. The measurement function computed with the trace and determinant of Hessian matrix is used to eliminate keypoints with strong edge responses and sub-pixel localization. Histograms of gradient orientation are the feature description. SIFT has been widely used in wide baseline matching, structure from motion, visual tracking and object recognition. SURF (speed up robust feature) [40] uses box filters to approximate the determinant of Hessian matrix and constructs fast Hessian interest point detector. A center-surround interest point detector with SURF (CerSURE) is presented in [41]. Another interest point detection by approximating

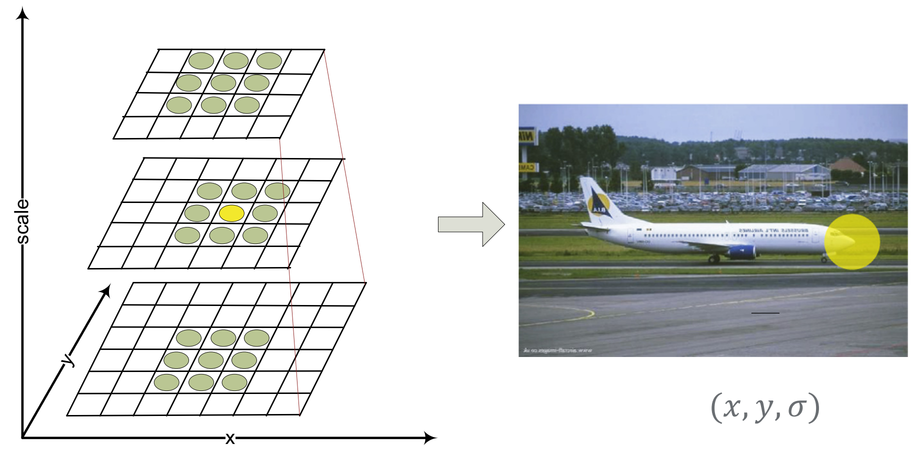

Fig. 9. Interest point detection.

the determinant of Hessian matrix is DART [42]. It uses weighted triangle responses to approximate the second derivative of Gaussian function, which corresponds to the elements of Hessian matrix. Aside from approximation, ranking, voting and other learning based operations are integrated to improve the stability of interest point. ROLG (rank order Laplacian of Gaussian) [44] is based on weighted rank order filter and LOG. LoG filter can be expressed as the subtraction between weighted averages. In ROLG, weighted median responses are used to replace the weighted average and generate the interest point detection methods. Voting strategy is applied to grouping darker or brighter image pixels for interest point detection in [82]. Rank-SIFT [43] applies supervised learning as RankSVM to select stable interest SIFT points. The score to measure stability is modeled and RankSVM is used to solve the ranking function for interest point detection. In [45], three objective functions as stability, dispersion and information content for interest points are constructed. Genetic programming is used to solve the multiobjective interest point detection problem.

Classical interest points such as LoG, DoG, DoH are based on the partial differentiation on Gaussian scale spaces. In recent years, newly defined interest points based on non-linear Partial Differential Equations (PDE) come out. KAZE features [46] find local extremas by nonlinear diffusion filtering. Based on the consideration that Gaussian smoothing leads to image blurring, diffusion filtering is introduced to provide multi-scale image spaces meanwhile preserve natural image boundaries. Additive operator splitting (AOS) scheme is used to solve the nonlinear partial differential equations for diffusion function. Based on the derivation, Scharr filters are used to approximate the first and second order derivatives of diffusion function. The main drawback of KAZE feature is of high computational cost. An accelerated version of KAZE is further proposed in [47]. WADE [48] interest point detection framework is based on wave propagation. Wave equation is prior to highlight and isolate salient symmetries, therefore WADE is priori to detect interest point with symmetries. Since the theoretical foundations of the above interest point methods are Partial Differential Equations (PDEs), we refer them as PDE based interest point detection. Besides, edge Foci interest point [83] is defined as the point which is approximately equidistant to edge points with gradient orientations perpendicular to itself. The detection of Edge foci interest points is based on the aggregation of batched directional filtering. Other interest point of symmetry [84] quantifies the self-similarity of regions for interest point detection.

Inspired by the computational speed of decision trees based corner detector (e.g., FAST [32], FAST-ER [33], AGAST [34]), interest point detection based on binary comparison and decision tree classification comes out. ORB (oriented FAST and rotated BRIEF) [49] uses FAST corner detector at each scale of image pyramid. Harris cornerness measurement is borrowed to suppress non-maximal potential interest points. The feature description of ORB is a rotated version of BRIEF [78,79]. BRISK (binary robust invariant scalable keypoints) [50] applies AGAST corner detector [34] in scale spaces to locate potential interest points. FAST score is used as saliency measurement for non-maximal suppression and interest point fining. FREAK (Fast retina keypoint) [51] uses the same interest point detector as BRISK, with binary descriptors motivated from human retina. Binary features are quite favored from the engineering point of view, since they are less time consuming and storage saving.

3.3.2. *Interest region detection*

Interest region refers to the region segmented from neighboring area by exploiting the constancy of image properties. Typical procedure of interest region detection is illustrated in Fig. 10. The definition of pixel constancy can be on pixel intensity, zero gradient, etc. Regions which remain stable along a large threshold range are chosen as interest regions. Ellipses or parallelograms are used to fit the segmented interest regions. The mathematical representation of interest regions depends on the parameters of fitted ellipses or parallelograms. Different from interest point detection, interest region detection often do not need extra multi-scale pyramid construction.

MSER (Maximally stable extremal region) [52] obtains interest regions based on thresholding the pixel intensities. Extremal regions are defined as those in which all the pixel values are brighter or darker than those on the boundaries. Suppose $i = 1, \ldots, n-1, n, n+1, \ldots$ are the tested thresholds, nested extremal regions as $Q_1, \ldots, Q_{n-1}$, $Q_n, Q_{n+1}, \ldots$ are obtained. The stability of a region is defined as the function of threshold n.

$$\psi(n) = |Q_n| / |Q_{n+\Delta} \backslash Q_{n-\Delta}| \tag{9}$$

where Δ is a parameter to determine the thickness of boundary and $|\cdot|$ is to obtain the cardinality. MSER is defined as Q_{n^*} with the local maximas of ψ_{n^*}. It aims to find regions which remain stable over a large range of threshold changes. Other traditional interest regions include IBR (intensity extrema-based region) [2], EBR (edge-based region) [2], salient region based on probability distribution of pixel intensities [53].

In recent years, the structure factors such as shape convex and curvature computing are combined with the classical MSER to segment reliable regions in natural images. An extension of MSER [85] takes the shape factor into account. The classical stability criteria is amended and shape convex measurements are combined

Fig. 10. Interest region detection.

to make the detector prefer regions with irregular boundaries. PCBR (Principal curvature based region) [54] is based on MSER operating in watershed regions of principal curvature images. The principal curvature image is extracted from eigenvalues of Hessian matrices. Enhanced watershed segmentation is borrowed and used in cleaned binary principal curvature images. MSER is further applied to detecting and fitting interest region in watershed maps. In [86], a kind of affine-invariant interest blob detector is proposed based on Gaussian curvature analysis. Scale-invariant interest point detector is used to locate sparse candidate points in the first stage. In the second stage, the Gaussian function is to fit the shape and location parameters of each interest points and Gaussian curvature is generated. The interest region is further defined based on estimated Gaussian curvature. Besides, color information is involved. MSCR (maximally stable colour region) [57] is an extension of MSER with color information. The colorful distance is derived by the Poisson statistics of image pixels. Agglomerative clustering is applied to successively grouping neighboring pixels with similar colors.

There are natural connections between interest regions and boundaries. The latter represents the intersecting area of different regions. Rather than intensity information used in MSER [52], boundary information becomes more integrated into interest region detection recently. The medial feature detector [56] is a boundary based framework. Firstly, a weighted distance map is generated with image gradients. After that, a global weighted medial axis is computed in the weighted distance map. Regions are segmented by medial axis decomposition. Finally, a shape fragmentation factor is defined to select the interest regions. Scale and affine invariant fan feature [58] is based on the extracted edge fragments. It firstly applies Harris measurement [26] to selecting salient edge points as candidate interest points along boundaries. Edge fragments are associated with candidate interest points to determine the shape of subregions. Fan Laplacian of Gaussian (FLOG) is used for automatic scale selection and final interest region localization. BPLR (boundary preserving local region) [59] is based on the learning based Pb edge detector [87]. The distance transform (DT) is computed for each boundary segment. Candidate circular regions are generated by densely sampled with maximal DT values. The minimum spanning tree algorithm is used to group the neighboring circular regions and generate the densely placed boundary preserving local regions. Compared with interest point detection, more parameters like rotation angles, aspect ratio are obtained by interest region detection. Feature descriptors extracted with segmented regions can be normalized with more geometrical parameters therefore are mostly affine-invariant.

Saliency detection aims to locate the interest regions which simulate human's attention in images. Nowadays it is a hot and fast evolving topic in computer vision. Recent surveys and comparative studies on saliency detection can be found in [88–90]. Based on the consideration that saliency detection is currently independent from visual feature detection and there are several existing surveys on this topic, here we just present a brief introduction. Early saliency detection is mainly based on image contrast and difference computation. Local contrast based features are combined to find salient regions [91]. Later, global analysis which takes the spatial distribution of similar image regions into account booms out. Saliency detection is much related to segmentation techniques. Appearance similarities and spatial distributions are two important elements of current salient region detection methods. Graph models such as Conditional Random Field (CRF) are used to combine the two elements into saliency assignment [92]. Typical recent methods are on multiple low-level cue combination [93], hierarchical saliency detection [94] and aggregation [95].

3.3.3. Discussions

Blob detection methods vary a lot along with the defined interest properties. They can be briefly classified by the detection framework and output expression. Interest point detection is closely related to the construction of scale spaces. They find the local extremas in 3D scale space as interest points. Classical methods are based on second-order derivatives of Gaussian scale spaces, which are sensitive to noise and of high computation complexity. Recent progress can be coarsely divided into three ways. The first is the generalization of classical Gaussian based methods. Approximation of partial derivatives is used to lift up the detection speed and interpolation is used to lift up the localization accuracy. Ranking and voting are integrated into the existing Gaussian scale-space based detection framework to improve the stability. Machine learning techniques such as RankSVM and genetic programming are also used to improve the stability. The second way focuses on constructing new PDEs from non-linear scale spaces and solves for the local extremas in order to locate interest points. The non-linear scale-space proposed in KAZE [46] features can generate smoothed scale spaces meanwhile preserve the natural boundaries of regions. Other interest point detection methods such as WADE [48] aim to extract symmetry which naturally exists and is an important Gesalt factor. The derivation of PDE based interest point detection is complicated and the computational cost for derivatives computing is always a matter for concerning. With the development of corner detection based on learnt binary trees (e.g., FAST [32], AGAST [34]), another kind of methods combine template-based corner detection with pyramid construction to extract interest points [49,50]. The time cost is reduced sharply but the extremal measurement among scales is difficult to determine. Compared with PDE based detectors, the stability of newly emerged FAST and AGAST based detectors need to be improved.

Interest region detection aims to segment regions and extract shape parameters for normalization, in order to make the detected features affine invariant. A classical straightforward way is to extend interest point detectors with affine invariant techniques, such as Hessian-affine, Harris-affine [96]. But most of the recent methods are inspired by segmentation. Shape factors are more involved in recent interest region detection methods. The integration of shape factors adds shape restrictions for optimization and lifts up the stability. Besides, edge and contour detectors are more applied in the interest region segmentation framework. Canny edge detector is still the popular choice and the learning based gPb [87] detector is also integrated into interest region segmentation. The natural topological connections between contours/boundaries and interest regions have been investigated in the recent progress. Image processing techniques such as watershed algorithm and distance transform are also integrated. The interest regions detected with edge fragments and contours can tolerate illumination changes, but the detection framework also becomes much more sophisticated. Compared with interest point detection, the segmented regions can provide more geometrical parameters for stereo matching. Also interest region detection shows advantages in the extraction and representation of smooth regions. Blob detection is inspired by wide baseline matching and widely researched in 3D reconstruction and object recognition. How to choose visual points and regions which remain stable under different viewpoints and illumination conditions is still quite important. Besides, saliency detection whose goal is to segment regions of attention in natural images becomes a hot topic recently.

4. Evaluation and databases

The evaluation of visual feature detection is quite important. A convincing evaluation framework can promote the research significantly. Although visual feature detection can be intuitively evaluated through human observation, it is insufficient for large-scale databases. Three criteria need to be considered for empirical evaluation as *detection accuracy, localization error* and *computational efficiency. Computational efficiency* can be measured by detection time cost. *Localization error* can be computed from the error distances in 2-D spaces. The measurements of *detection accuracy* are various, with the key idea to find correspondences. The differences lie in the definition of correspondences. The first kind of correspondence is based on matching detected responses with human labeled ground truth. The second kind is to match detected responses in associated images. We mainly focus on the measurements used in recent literature. Typical databases for evaluating edge detection, corner detection and blob detection are presented in Table 3.

The evaluation of edge detection is realized by finding correspondences between detected responses and human labeled ground truth. The typical BSDS database provides several human labeled boundary maps for each image. The parametric curve named as precision-recall and attached real-value measurements (i.e., average precision, *F*-measure) are used for evaluation. With a fixed threshold, binary detected responses can be obtained by the edge detection methods. The detected responses are matched with human labeled boundaries. True positive N_{tp} is defined as the number of matched responses. False positive N_{fp} is defined the number of detected responses which cannot match the ground truth. Suppose there are N_p labeled edge points. Precision is defined as the ratio between true positives and the number of all detected responses. Recall is defined the ratio between true positives and human labeled ground truth. That is,

$$\text{Precision} = N_{tp}/(N_{tp}+N_{fp}), \text{Recall} = N_{tp}/N_p. \quad (10)$$

When there are several ground truth maps, the binary detected responses are matched with each human labeled boundary map. Recall is further defined as the average of recall rates obtained by matching each map. False positive is defined as the number of detected responses which cannot match any of human labeled edges and precision is further computed by Eq. (10). The detection methods which can achieve higher precision and recall are more highly ranked. Average precision (AP) is a real-value measurement which indicates the covering area of PR curves. The common-used *F*-measure is defined as

$$F = P \cdot R/(\alpha R+(1-\alpha)P) \quad (11)$$

$0<\alpha<1$ is the weight to balance the importance of precision and recall. Commonly $\alpha=0.5$. That is, $F=2P \cdot R/(R+P)$. The maximum *F*-measure is used as the real value for algorithm ranking. Besides, there are newly proposed measurements for edge detection [100,101].

By the widely acknowledged BSDS benchmark, edge and contour detection algorithms are evaluated. We list the maximum *F*-measure of representative methods in Table 4, with the results obtained from public evaluation[2] or existing literature. From the table we can find that learning based edge detection generally can achieve higher *F*-measure when compared with differentiation based methods. That is mainly because differentiation based edge detection do not discriminate texture edges. By classical differentiation based edge detection such as Roberts, Sobel, Prewitt, the achieved *F*-measure on BSDS300 dataset is around 0.48. Besides, the *F*-measure of Canny edge detection is 0.58. By learning based methods, the *F*-measure varies from 0.63 to 0.74. The learning based edge detection can be divided by the integrated cues (i.e., brightness, texture, and color). From the separate evaluation of Pb [15], the *F*-measure achieved by brightness gradient is 0.60, which is higher than that of texture gradient (i.e., 0.58) and color gradient (i.e., 0.57). The highest achieved *F*-measure with grayscale information is obtained by gPb [18] and SCG [24], as 0.68. The integration of color information would lift the *F*-measure by about 0.02–0.03, according to the empirical evaluation provided by BEL [21], gPb [18], and SCG [24]. The highest reported *F*-measure on BSDS300 dataset is 0.74 [19], which is based on tPb edge response prediction and sequential labeling based edge linking. Besides, the highest *F*-measure reported *F*-measure on BSDS500 dataset is 0.74 by SFE [25]. It is noteworthy that the ideal *F*-measure obtained by human annotation is 0.79/0.80, which is the ultimate goal for learning based boundary detection. Besides, the computational efficiency is also a concerned issue for existing empirical evaluation. In general, learning based boundary detection is much more computational expensive than differentiation based edge detection. Among all the learning based methods, only SFE [25] claims that it is real-time. BEL [21] and SFE [25] which are based on generic descriptors and feature selection classification algorithms (i.e., Boosting for BEL and random forest for SFE) have relatively low computational cost. But the claim is not suitable for SCG [24] which is based on the computation of sparse codes. It is the most computational expensive method [25]. The integration of global information and edge linking increases the *F*-measure but extra time cost also adds.

The evaluation of corner and blob detection is to find correspondences in pairwise images. Groups of images are captured for the same scenes under different conditions (e.g., scale, viewpoint, illumination, noise, JPEG compression, and image blur). The databases for evaluation are composed by those groups of images and the labeled imaging conditions. A simple measurement is the number of detected corners/blobs. It is used to measure the adaptability of methods. More features imply more sufficient information for matching and image interpretation. The most common-used measurement for empirical evaluation of corner/blob detection is repeatability. It aims to measure the ratio of correct correspondences in changing imaging conditions. Suppose I_a and I_b denote pairwise images and H is the estimated homography from I_a to I_b. P_a and P_b are detected keypoints/corners. R_a and R_b are detected interest regions. The correct correspondence is

[2] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/bench/html/algorithms.html

Table 3
The representative datasets for evaluation on feature detection.

	Name	Description
Edge	Heath dataset	28 color images are included in this dataset and each image is normalized to be 512 × 512. It includes manmade and natural images. Besides, ground truth of edges is provided. A detailed evaluation based on this dataset is provided in [97]. http://marathon.csee.usf.edu/edge/edge_detection.html
	Bowyer dataset	The dataset contains 50 object images and 10 aerial scene images. Each object image has a single object approximately centered in the image with surrounding natural background. All images are normalized to 512 × 512. Human specified ground truth is provided. A detail evaluation based on this dataset is provided in [98]. http://figment.csee.usf.edu/edge/roc/
	BSDS dataset	Berkeley segmentation dataset and benchmark (BSDS) is for evaluating edge detection. An early version of this dataset contains 300 images, with 200 training images and 100 test images. 200 extra test images are added and the dataset extends to 500 images in total. 5–10 human-marked boundaries with scores are provided for each image. A public benchmark is provided. http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/
Corner	Rosten dataset	It provides three sets of registered images, which are identified as *box, maze and junk*. *Box* has photographs taken of a test rig with strong perspective, scale changes as well as radial distortion. *Maze* has photographs taken of a prop used in an augmented reality, also with perspective and scale changes. *Junk* has photographs with many objects with significant relief [33]. http://www.edwardrosten.com/work/datasets.html
Blob	Oxford vision dataset	It contains eight groups of images. Each group has six images. Five different changes in image conditions (e.g: image blur, JPEG compression, viewpoint change, scale change and illumination change) are included. All the images are of medium size which is approximately 800 × 400 pixels. The perspective mappings are provided [7,2,96]. http://www.robots.ox.ac.uk/~vgg/research/affine
	Strecha dataset	It consists of image sequences from several scenes. They are used to evaluate the effects of perspective transformation of non-planar geometry [99]. http://www.cs.unc.edu/~jheinly/feature-evaluation/datasets.html
	Robot dataset	It contains 135,660 color images captured for 60 scenes in 119 positions. Each scene from each position is illuminated by 19 white LEDs. The images are with the resolution of 1200 × 1600. The images are captured with an industrial robot with mounted camera. The precise camera positioning is provided [3]. http://roboimagedata.dtu.dk

measured as

$$\textbf{Point-to-point correspondence}: \|P_a - HP_b\| < K. \tag{12}$$

$$\textbf{Region-to-region correspondence}: \frac{R_a \cap H^T R_b H}{R_a \cup H^T R_b H} > \varepsilon. \tag{13}$$

in which K is the pixel location error threshold and ε is the region overlap threshold. The repeatability measurement is defined as the ratio of corresponding pairs against the minimal total number of detected features. It can also be defined as the ratio of corresponding pairs and the summed number of detected features in different images. The method with higher repeatability is viewed as better, since it indicates that the method can stably detect features under changing conditions. Parametric curves can be obtained by continuously changing the scales, viewpoints, illuminations with computed repeatability as the dependent variables. Repeatability is widely used in the benchmark to evaluate corner and blob detection methods [3,7,103]. There are other measurements. For example, entropy is used to measure the spread in the spatial distributions of interest points in the images, based on the consideration that spread distribution helps reduce the feature confusion.

Although the repeatability is a well-acknowledged measurement for empirical evaluation, the experimental settings (e.g., pixel/overlap threshold, dataset, and computer configurations) vary for existing literature and comparative studies [7,3]. However, despite of the differences in experimental setting, there are several interesting results concluded from existing empirical evaluation. Oxford dataset is most widely used for evaluation. According to existing experimental results, the recent template based corner detection with decision tree classification such as FAST [32], FAST-ER [33], and AGAST [34] has brought significant computational efficiency. The corner detection can be accelerated significantly when compared with classical gradient based corner detection methods such as Harris [26]. FAST-ER [33] is slightly slower than FAST [32] due to the thicker template is applied. But the stability of FAST-ER corner detection is also increased by the thicker template. AGAST [34] is more effective and computational efficient than FAST [32]. Since the template based interest point detection such as ORB [49] and BRISK [50] are based on decision tree classification based corner detection, they share the advantage of low

Table 4
Typical edge/boundary detection methods and their achieved maximum *F*-measure on BSDS datasets.

Methods	Maximum *F*-measure on BSDS300 dataset	Maximum *F*-measure on BSDS500 dataset
Human	**0.79**	**0.80**
Roberts/Sobel/Prewitt	0.47/0.48/0.48	–/–/–
Canny	0.58	0.60
Pb (gray/color) [15]	0.63/0.65	–/–
MS-Pb (gray/color) [16]	0.66/0.68	–/–
gPb (gray/color) [18]	0.68/0.70	0.68/0.70
BEL (gray/color) [21]	0.64/0.66	–/–
SCG (gray/color) [24]	0.71/–	0.71/0.74
NMX [20]	0.66	–
tPb (SeqLabel) [19]	0.74	0.74
Sketch tokens [23]	–	0.73
SFE [25]	–	0.74

computational cost. However, there are several disadvantages of decision tree classification based corner detection. One is that they do not perform well in blurred images [49]. The repeatability declines in large viewpoint changes. The other is that the number of detected corners is not stable, according to the empirical evaluation provided in [3]. Besides, corner is scale relevant. Therefore the repeatability measurement score for corner detection such as classical Harris and recent FAST [32], AGAST [34] is not competitive under scale changes, which is expected.

Aside from the involvement of decision tree based corner detection, another progress in interest point detection is the integration of approximation techniques, such as SURF [40], CerSURE [41] and DART [42]. According to the experiments, the computational efficiency is lifted up meanwhile the repeatability of SURF [40], CerSURE [41], DART [42] is comparable of classical SIFT [39] in viewpoint, scale, illumination changes. But the repeatability of in-plane rotation changes is decreased, since the box and triangle filtering based approximation more or less break the isotropic property of Gaussian smoothing. In contrast, the introduction of non-linear PDEs and new features such as KAZE [46] and WADE [48] are computational expensive. But the repeatability under Gaussian blurred conditions is more stable. Interest region detection is integrated with segmentation techniques. Classical MSER [52] has displayed advantages [7], especially for structured scenes. In contrast, PDE based interest point detection is better performed in textured scenes. The difference is expected since the visual property of interest varies. In general, MSER [52] achieves higher repeatability scores and better performance under viewpoint and illumination changes, but is sensitive to image blur changes [7]. The integration of color information such as MSCR [57] increases the repeatability under viewpoint changes for texture scenes, but the computation burden also adds. On the other hand, the integration of structure factors (i.e., PCBR [54]) and boundary information (i.e., BPLR [59]) into interest region detection displays performance improvement in visual tasks such as object recognition [59] and image retrieval [56].

5. Summary and discussions

The objective of visual feature detection is to identify interest image structures (e.g., points, curves, and regions). Feature detection is an important part in computer vision systems. Effective and efficient features are required in booming computer vision applications, such as wide baseline matching, structure from motion, object recognition, and image retrieval. The research of feature detection can trace back to the beginning of computer vision and many classical methods are proposed. In this paper, we mainly focus on the recent progress in visual feature detection. We present and categorize the recent advances in detecting edges, contours, corners and blobs. In general, a part of recent proposals inherit the key ideas of classical methods. For example, LoG measurement for interest point detection has been integrated into several recent features. Apart from the extensions for classical methods, two trends need to be noted in recent progress. The first is the involvement of machine learning techniques in visual feature detection. The feature detection is modeled as a learning and inference problem. Human labeled features serve as the training samples for feature existence determination. The second trend is the exploitation of connections existing in different kinds of features. Boundary detection is more integrated into corner and interest region detection. Based on the categorization, we also discuss the pros and cons of different feature detection methods. We would like to provide a reference for interested researchers and identify the trend of feature detection.

Early edge detection finds pixels with abrupt changes by differential operators. Recent efforts on differentiation based edge detection are mainly on multi-resolution analysis, sub-pixel localization and hysteresis threshold determination. A significant progress on edge detection is on the emerging learning based methods. Edge detection is modeled as a learning and inference problem to discriminate boundaries from backgrounds. Multiple low-level visual cues (e.g., color, texture, gradient, and sparse codes) are fused into the learning and inference models for edge response prediction. Rather than serving as the ground truth for evaluation, human labeled boundaries also provide training samples for learning edge prediction models. The learning based methods can suppress internal edges which wildly occur in textured regions. Corner detection is much related to gradient orientations. Classical corner detection is based on the calculation of second order derivatives, which is quite time consuming. Although there are recent approximated detectors, the computational cost is still high. Recently machine learning has been combined with template based cornerness measurement. Fast corner detection is achieved by accelerated pixel comparisons inside templates. The natural connections between contours and corners promote the contour based corner/junction detection. Recent attempts include the integration of effective learning based boundary detection. There are mainly two motivations to employ learning. One is to improve detection accuracy and the other is to increase detection speed. Learning based boundary detection is motivated by improving detection accuracy. In contrast, the learning algorithms used in corner detection aims to increase the detection speed. A shared issue to introduce learning is that the learnt models for detection might be database-dependent.

Blob detection has always been an active topic, especially in wide baseline matching. Interest point detection is based on the construction of scale spaces. Traditional methods aim to find local extremas on Gaussian scale spaces. Approximations of LoG functions have been proposed to lift up the detection speed. Besides, machine learning techniques such as ranking, genetic programming are explored to improve the stability of detected responses. Other new definitions of interest point come out and partial derivative equations based theoretical analysis is fulfilled to identify them. Corner detection based on learning is combined with multi-scale analysis to detect interest points and produce binary features. An advantage of binary features is that they are quite time and storage saving, which are favored in large-scale web retrieval and mobile applications. Interest region detection is based on segmentation techniques. The classical MSER is based on pixel intensity constancy. There are several recent extensions of MSER, which combine structure factors and color information to generate more reliable segmented regions in natural images. Inspired by the connections between boundaries and interest regions, boundary information is more integrated in recent years. Salient region detection which simulates human attention has achieved much progress in recent years and it is now a hot topic. Local contrasts as well as other spatial information are combined in learning and inference model for saliency calculation.

Visual feature detection is much involved and inspired by various computer vision applications. Edges and contours are much related to object boundaries. They are required and applied in image interpretation, such as object recognition, image segmentation, visual tracking, action analysis. Compared with edges, corners and blobs are unique in local image regions, therefore are favored in wide baseline matching, stereo, SLAM. Blobs can also be extensively used to identify sparse image regions for compressed image representation, therefore are widely applied in object recognition and image retrieval. The evaluation of visual feature detection is distinct and dependent on applications. Straightforward human observation is difficult and subjective to evaluate numerous test images. Considering that, quantitative metrics are researched and proposed. The detection accuracy is the most concerned property and it relies on finding correspondences. The evaluation on edge and contour detection is based on measuring the correspondence between detected responses and human labeled ground truth. Precision–recall curves and attached F-measure are widely acknowledged measurements for evaluation. The detection accuracy of corner and blob detection is measured by finding correspondences in images captured under changing conditions. It is much inspired by the applications in stereo matching. Groups of images are captured under different images conditions for different scenes. The perspective mapping functions are recorded for the grouped images captured from the same scenes under different conditions. The repeatability is a common used measurement for corner and blob detection. Computational complexity needs to be considered for real-time applications. Besides, the storage requirement needs to be considered and estimated in large-scale image retrieval and mobile applications.

The future challenges of feature detection lie in four aspects. Firstly, we need to design more effective and efficient feature detection methods for booming computer vision applications.

Feature is the low-level representation bridging from image pixels to semantic meanings, and it is the basis for high-level image interpretation. How to devise and detect specified task-dependent features is an important issue. Besides, due to the wider video applications such as human action recognition [104], temporal analysis of visual features needs to be developed. Second, the relations among different kinds of features need to be further exploited. The definitions of visual features are topologically connected. The topological relations are important but far from sufficient exploitation in current methods. Although there are attempts to detect different types of visual features simultaneously, we need to exploit the universal framework to identify visual features of different types for comprehensive image parsing. Third, machine learning is more involved in visual feature detection than ever. Yet there are two issues need to be solved. One is to balance between detection accuracy and computation efficiency. The other is to make the learnt models for feature detection database-independent. There is still much work in exploiting the model and constructing generic databases. Finally, the evaluation for visual feature detection needs to be improved. A convincing and comprehensive evaluation framework can promote the research significantly. Visual feature detection is not the last processing step and different characteristics are valued for different tasks. The human labeled ground truth features might be ambiguous and unreliable. The evaluation of visual feature detection need include two parts. One is the universal evaluation to measure the correspondences between detected responses and ground truth. The other is the specified evaluation for various computer vision tasks.

References

[1] S. Gauglitz, T. Hollerer, M. Turk, Evaluation of interest point detectors and feature descriptors for visual tracking, Int. J. Comput. Vis. 94 (2011) 335–360.
[2] T. Tuytelaars, L.V. Gool, Matching widely separated views based on affine invariant regions, Int. J. Comput. Vis. 59 (1) (2004) 61–85.
[3] H. Aanas, A.L. Dahl, K.S. Pedersen, Interesting interest points a comparative study of interest point performance on a unique data set, Int. J. Comput. Vis. 97 (2012) 18–35.
[4] M. Basu, Gaussian-based edge-detection methods—a survey, IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev. 32 (3) (2002) 252–260.
[5] M.A. Oskoei, H. Hu, A Survey on Edge Detection Methods, Technical Report: CES-506, University of Essex.
[6] G. Papari, N. Petkov, Edge and line oriented contour detection: state of the art, Image Vis. Comput. 29 (2011) 79–103.
[7] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, L.V. Gool, A comparison of affine region detectors, Int. J. Comput. Vis. (2005) 43–72.
[8] D. Song, D. Tao, Biologically inspired feature manifold for scene classification, IEEE Trans. Image Proc. 19 (1) (2010) 174–184.
[9] Y. Huang, K. Huang, D. Tao, T. Tan, X. Li, Enhanced biologically inspired model for object recognition, IEEE Trans. Syst. Man Cybern. Part B: Cybern. 41 (6) (2011) 1668–1680.
[10] P. Perona, J. Malik, Detecting and localizing edges composed of steps, peaks and roofs, Proceedings of International Conference on Computer Vision (ICCV) (1990) 52–57.
[11] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern. Anal. Mach. Intell. 8 (6) (1986) 679–698.
[12] Q. Xu, S. Varadarajan, C. Chakrabarti, L.J. Karam, A distributed canny edge detector: algorithm and FPGA implementation, IEEE Trans. Image Proc. 23 (7) (2014) 2944–2960.
[13] W. McIlhagga, The canny edge detector revisited, Int. J. Comput. Vis. 91 (2011) 251–261.
[14] K. Yang, S. Gao, C. Li, Y. Li, Efficient color boundary detection with color-opponent mechanisms, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2810–2817.
[15] D. Martin, C. Fowlkes, J. Malik, Learning to detect natural image boundaries using local brightness, color, and texture cues, IEEE Trans. Pattern. Anal. Mach. Intell. 26 (5) (2004) 530–549.
[16] X. Ren, Multi-scale improves boundary detection in natural images, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2008, pp. 533–545.
[17] M. Maire, P. Arbelaez, C. Fowlkes, J. Malik, Using contours to detect and localize junctions in natural images, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2008, pp. 1–8.
[18] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. Pattern. Anal. Mach. Intell. 33 (5) (2011) 898–916.
[19] N. Payet, S. Todorovic, Sledge: sequential labeling of image edges for boundary detection, Int. J. Comput. Vis. 104 (1) (2013) 15–37.
[20] I. Kokkinos, Highly accurate boundary detection and grouping, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2520–2527.
[21] P. Dollar, Z. Tu, S. Belongie, Supervised learning of edges and object boundaries, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2006, pp. 1964–1971.
[22] J. Mairal, M. Leordeanu, F. Bach, M. Hebert, J. Ponce, Discriminative sparse image models for class-specific edge detection and image interpretation, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2008, pp. 43–56.
[23] J.J. Lim, C.L. Zitnick, P. Dollar, Sketch tokens: a learned mid-level representation for contour and object detection, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 3158–3165.
[24] X. Ren, L. Bo, Discriminatively trained sparse code gradients for contour detection, in: Proceedings of Conference in Advances in Neural Information Processing Systems (NIPS), 2012, p. 25.
[25] P. Dollar, C.L. Zitnick, Structured forests for fast edge detection, in: Proceedings of International Conference on Computer Vision (ICCV), 2013, pp. 1841–1848.
[26] C. Harris, M. Stephens, A combined corner and edge detector, in: Proceedings of Alvey Vision Conference, 1988, pp. 147–151.
[27] C. Tomasi, T. Kanade, Detection and Tracking of Point Features, Dept. Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep.,CMU-CS-91-132, April.
[28] J. Shi, C. Tomasi, Good features to track, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 1994, pp. 593–600.
[29] P. Mainali, Q. Yang, G. Lafruit, R. Lauwereins, L.V. Gool, Lococo: low complexity corner detector, in: Proceedings of IEEE International Conference on Acoustics, Speech Signal Process. (ICASSP), 2010, pp. 810–813.
[30] P. Mainali, Q. Yang, G. Lafruit, L.V. Gool, R. Lauwereins, Robust low complexity corner detector, IEEE Trans. Circuit Syst. Video Technol. 21 (4) (2011) 435–445.
[31] S.M. Smith, J.M. Brady, Susan – a new approach to low level image processing, Int. J. Comput. Vis. 23 (1) (1997) 45–78.
[32] E. Rosten, T. Drummond, Machine learning for high speed corner detection, Proceedings of European Conference on Pattern Recognition (ECCV) 1 (2006) 430–443.
[33] E. Rosten, R. Porter, T. Drummond, Faster and better: a machine learning approach to corner detection, IEEE Trans. Pattern. Anal. Mach. Intell. 32 (1) (2010) 105–119.
[34] E. Mair, G.D. Hager, D. Burschka, M. Suppa, G. Hirzinger, Adaptive and generic corner detection based on accelerated segment test, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2010, pp. 183–196.
[35] X. Zhang, H. Wang, A.W.B. Smith, X. Ling, B.C. Lovell, D. Yang, Corner detection based on gradient correlation matrices of planar curves, Pattern Recognit. 43 (4) (2010) 1207–1223.
[36] P.L. Shui, W.C. Zhang, Corner detection and classification using anisotropic directional derivative representations, IEEE Trans. Image Proc. 22 (8) (2013) 3204–3218.
[37] A. Willis, Y. Sui, An algebraic model for fast corner detection, in: Proceedings of International Conference on Computer Vision (ICCV), 2009, pp. 2296–2302.
[38] G.S. Xia, J. Delon, Y. Gousseau, Accurate junction detection and characterization in natural images, Int. J. Comput. Vis. 106 (1) (2013) 31–56.
[39] D. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[40] H. Bay, A. Ess, T. Tuytelaars, L.V. Gool, Surf: speeded up robust features, Comput. Vis. Image Underst. 110 (3) (2008) 346–359.
[41] M. Agrawal, K. Konolige, M.R. Blas, Censure: Center surround extremas for realtime feature detection and matching, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2008, pp. 102–125.
[42] D. Marimon, A. Bonnin, T. Adamek, R. Gimeno, Darts: Efficient scale-space extraction of daisy keypoints, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2416–2423.
[43] B. Li, R. Xiao, Z. Li, R. Cai, B.L. Lu, L. Zhang, Rank-sift: learning to rank repeatable local interest points, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 1737–1744.
[44] Z. Miao, X. Jiang, Interest point detection using rank order log filter, Pattern Recognit. 46 (2013) 2890–2901.
[45] G. Olague, L. Trujillo, Interest point detection through multiobjective genetic programming, Appl. Soft Comput. 12 (2012) 2566–2582.
[46] P.F. Alcantarilla, A. Bartoli, A.J. Davison, Kaze features, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2012, pp. 214–227.
[47] P.F. Alcantarilla, J. Nuevo, A. Bartoli, Fast explicit diffusion for accelerated features in nonlinear scale spaces, in: Proceedings of the British Machine Vision Conference (BMVC), 2013.
[48] S. Salti, A. Lanza, L.D. Stefano, Keypoints from symmetries by wave propagation, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2898–2905.

[49] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, Orb: an efficient alternative to sift or surf, in: Proceedings of International Conference on Computer Vision (ICCV), 2011, pp. 2564–2571.
[50] S. Leutenegger, M. Chli, R.Y. Siegwart, Brisk: Binary robust invariant scalable keypoints, in: Proceedings of International Conference on Computer Vision (ICCV), 2011, pp. 2548–2555.
[51] A. Alahi, R. Ortiz, P. Vandergheynst, Freak: fast retina keypoint, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 510–517.
[52] J. Matas, O. Chum, M. Urban, T. Pajdla, Robust widebaseline stereo from maximally stable extremal regions, Image Vis. Comput. 22 (10) (2004) 761–767.
[53] T. Kadir, A. Zisserman, M. Brady, An affine invariant salient region detector, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2004, pp. 345–357.
[54] H. Deng, W. Zhang, E. Mortensen, T. Dietterich, L. Shapiro, Principal curvature-based region detector for object recognition, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2007, pp. 1–8.
[55] S. Gu, Y. Zheng, C. Tomasi, Critical nets and beta-stable features for image matching, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2010, pp. 663–676.
[56] Y. Avrithis, K. Rapantzikos, The medial feature detector: stable regions from image boundaries, in: Proceedings of International Conference on Computer Vision (ICCV), 2011, pp. 1724–1731.
[57] P.E. Forssen, Maximally stable colour regions for recognition and matching, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2007, pp. 1–8.
[58] C. Cui, K.N. Ngan, Scale- and affine-invariant fan feature, IEEE Trans. Image Proc. 20 (6) (2011) 1627–1640.
[59] J. Kim, K. Grauman, Boundary preserving dense local regions, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 1553–1560.
[60] S. Yi, D. Labate, G.R. Easley, H. Krim, A shearlet approach to edge analysis and detection, IEEE Trans. Image Proc. 18 (5) (2009) 929–941.
[61] C. Lopez-Molina, B. De Baets, H. Bustince, J. Sanz, E. Barrenechea, Multiscale edge detection based on gaussian smoothing and edge tracking, Knowl.-Based Syst. 44 (2013) 101–111.
[62] J. Liang, S.Y. Yuen, An edge detection with automatic scale selection approach to improve coherent visual attention model, Pattern Recognit. Lett. 34 (2013) 1519–1524.
[63] D. Sen, S.K. Pal, Gradient histogram: thresholding in a region of interest for edge detection, Image Vis. Comput. 28 (2010) 677–695.
[64] R. Medina-Carnicer, A. Carmona-Poyato, R. Munoz-Salinas, F.J. Madrid-Cuevas, Determining hysteresis thresholds for edge detection by combining the advantages and disadvantages of thresholding methods, IEEE Trans. Image Proc. 19 (1) (2010) 165–173.
[65] T. Hermosilla, E. Bermejo, A. Balaguer, L.A. Ruiz, Non-linear fourth-order image interpolation for subpixel edge detection and localization, Image Vis. Comput. 26 (2008) 1240–1248.
[66] T. Bin, A. Lei, J. Cui, W. Kang, D.D. Liu, Subpixel edge location based on orthogonal Fourier–Mellin moments, Image Vis. Comput. 26 (4) (2008) 563–569.
[67] F. Da, H. Zhang, Sub-pixel edge detection based on an improved moment, Image Vis. Comput. 28 (2010) 1645–1658.
[68] P. Viola, M. Jones, Robust real-time object detection, Int. J. Comput. Vis. 57 (2) (2002) 137–154.
[69] Q. Zhu, G. Song, J. Shi, Untangling cycles for contour grouping, in: Proceedings of International Conference on Computer Vision (ICCV), 2007, pp. 1–8.
[70] Y. Ming, H. Li, H. Li, Winding number for region-boundary consistent salient contour extraction, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2818–2825.
[71] H. Wei, B. Lang, Q. Zuo, Contour detection model with multi-scale integration based on non-classical receptive field, Neurocomputing 103 (2013) 247–262.
[72] N. Widynski, M. Mignotte, A multiscale particle filter framework for contour detection, IEEE Trans. Pattern. Anal. Mach. Intell. http://doi.ieeecomputersociety.org/10.1109/TPAMI.2014.2307856, in press.
[73] X. Zhang, H. Wang, M. Hong, L. Xu, D. Yang, B.C. Lovell, Robust image corner detection based on scale evolution difference of planar curves, Pattern Recognit. Lett. 30 (1) (2009) 449–455.
[74] T.P. Nguyen, I. Debled-Rennesson, A discrete geometry approach for dominant point detection, Pattern Recognit. 44 (2011) 32–44.
[75] M. Awrangjeb, G. Lu, An improved curvature scale-space corner detector and a robust corner matching approach for transformed image identification, IEEE Trans. Image Proc. 17 (12) (2008) 2425–2441.
[76] R. Elias, R. Laganiere, Judoca: junction detection operator based on circumferential anchors, IEEE Trans. Image Proc. 21 (4) (2012) 2109–2118.
[77] M. Awrangjeb, G. Lu, C.S. Fraser, Performance comparisons of contour-based corner detectors, IEEE Trans. Image Proc. 21 (9) (2012) 4167–4179.
[78] M. Calonder, V. Lepetit, C. Strecha, P. Fua, Brief: binary robust independent elementary features, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2010, pp. 778–792.
[79] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, P. Fua, Brief: computing a local binary descriptor very fast, IEEE Trans. Pattern. Anal. Mach. Intell. 34 (7) (2012) 1281–1298.
[80] J. Heinly, E. Dunn, J.M. Frahm, Comparative evaluation of binary features, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2012, pp. 759–773.
[81] C. Ding, J. Choi, D. Tao, L.S. Davis, Multi-Directional Multi-Level Dual-Cross Patterns for Robust Face Recognition, Proceedings of CoRR, 2014.
[82] Z. Miao, X. Jiang, A vote of confidence based interest point detector, in: Proceedings of IEEE International Conference on Acoustic, Speech Signal Process (ICASSP), 2013, pp. 1459–1463.
[83] C.L. Zitnick, K. Ramnath, Edge foci interest points, in: Proceedings of International Conference on Computing Vision (ICCV), 2011, pp. 359–366.
[84] J. Maver, Self-similarity and points of interest, IEEE Trans. Pattern. Anal. Mach. Intell. 32 (7) (2010) 1211–1226.
[85] R. Kimmel, C. Zhang, A.M. Bronstein, M.M. Bronstein, Are MSER features really interesting? IEEE Trans. Pattern. Anal. Mach. Intell. 33 (11) (2011) 2316–2320.
[86] L. Ferraz, X. Binefa, A sparse curvature-based detector of affine invariant blobs, Comput. Vis. Image Underst. 116 (2012) 524–537.
[87] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, From contours to regions: an empirical evaluation, in: Proceedings of International Conference on Computing Vision and Pattern Recognition (CVPR), 2009, pp. 2294–2301.
[88] A. Toet, Computational versus psychophysical bottom-up image saliency: a comparative evaluation study, IEEE Trans. Pattern. Anal. Mach. Intell. 33 (11) (2011) 3131–2146.
[89] A. Borji, D.N. Sihite, L. Itti, Salient object detection: a benchmark, in: Proceedings of European Conference on Pattern Recognition (ECCV), 2012, pp. 414–429.
[90] N. Riche, M. Duvinage, M. Mancas, B. Gosselin, T. Dutoit, Saliency and human fixations: state-of-the-art and study of comparison metrics, in: Proceedings of International Conference on Computer Vision (ICCV), 2013, pp. 1153–1160.
[91] M.M. Cheng, G.X. Zhang, N.J.M.X. Huang, S.M. Hu, Global contrast based salient region detection, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 409–416.
[92] F. Perazzi, P. Krahenbuh, Y. Pritch, A. Hornung, Saliency filters: contrast based filtering for salient region detection, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 733–740.
[93] P. Jiang, H. Ling, J. Yu, J. Peng, Salient region detection by UFO: uniqueness, focusness and objectness, in: Proceedings of International Conference on Computer Vision (ICCV), 2013, pp. 1976–1983.
[94] Q. Yan, L. Xu, J. Shi, J. Jia, Hierarchical saliency detection, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 1155–1162.
[95] L. Mai, Y. Niu, F. Liu, Saliency aggregation: a data-driven approach, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 1131–1138.
[96] T. Tuytelaars, K. Mikolajczyk, Local invariant feature detectors: a survey, Comput. Gr. Vis. 3 (3) (2007) 177–280.
[97] M. Heath, S. Sarkar, T. Sanocki, K.W. Bowyer, A robust visual method for assessing the relative performance of edge-detection algorithms, IEEE Trans. Pattern. Anal. Mach. Intell. 19 (12) (1997) 1338–1359.
[98] K. Bowyer, C. Kranenburg, S. Dougherty, Edge detector evaluation using empirical roc curves, Comput. Vis. Image Underst. 84 (1) (2001) 77–103.
[99] C. Strecha, W. von Hansen, L. V. Gool, P. Fua, U. Thoennessen, On benchmarking camera calibration and multi-view stereo for high resolution imagery, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2008, pp. 1–8.
[100] C. Lopez-Molina, B. De Baets, H. Bustince, Quantitative error measures for edge detection, Pattern Recognit. 46 (2013) 1125–1139.
[101] X. Hou, A. Yuille, Boundary detection benchmarking: beyond f-measures, in: Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2123–2130.
[102] Y. Li, S. Wang, Q. Tian, X. Ding, Learning Cascaded Shared-Boost Classifiers for Part-Based Object Detection, IEEE Trans. Image Proc 23 (4) (2014) 1858–1871.
[103] E. Bostanci, N. Kanwal, A.F. Clark, Spatial statistics of image features for performance comparison, IEEE Trans. Image Proc. 23 (1) (2014) 153–162.
[104] Z. Zhang, D. Tao, Slow feature analysis for human action recognition, IEEE Trans. Pattern. Anal. Mach. Intell. 34 (3) (2012) 436–450.

Yali Li received the B.E. degree with Excellent Graduates Award from Nanjing University, China, in 2007 and the Ph.D degree from Tsinghua University, Beijing, China, 2013. Currently she is a research assistant in Department of Electronic Engineering, Tsinghua University. Her research interests include image processing, pattern recognition, computer vision, and video analysis etc.

Shengjin Wang received the B.E. degree from Tsinghua University, China, in 1985 and the Ph.D. degree from the Tokyo Institute of Technology, Tokyo, Japan, in 1997. From May 1997 to August 2003, he was a member of the research staff in the Internet System Research Laboratories, NEC Corporation, Japan. Since September 2003, he has been a Professor with the Department of Electronic Engineering, Tsinghua University, where he is currently also a Director of the Research Institute of Image and Graphics (RIIG). He has published more than 50 papers on image processing. He is the holder of ten patents. His current research interests include image processing, computer vision, video surveillance, and pattern recognition.

Qi Tian received the B.E. degree from Tsinghua University, China, in 1992, the M.S. degree from Drexel University in 1996 and the Ph.D. degree from the University of Illinois, Urbana-Champaign in 2002. He is currently a Professor in the Department of Computer Science at the University of Texas at San Antonio (UTSA). He took a 1-year faculty leave at Microsoft Research Asia (MSRA) during 2008–2009. His research interests include multimedia information retrieval and computer vision. He has published over 180 refereed journal and conference papers. He has received faculty research awards from Google, NEC Laboratories of America, FXPAL, Akiira Media Systems, and HP Labs. He received the Best Paper Awards in MMM 2013 and ICIMCS 2012, the Top 10% Paper Award in MMSP 2011, the Best Student Paper in ICASSP 2006, and the Best Paper Candidate in PCM 2007. He received 2010 ACM Service Award. He is the Guest Editors of IEEE Transactions on Multimedia, Journal of Computer Vision and Image Understanding, Pattern Recognition Letter, EURASIP Journal on Advances in Signal Processing, Journal of Visual Communication and Image Representation, and is in the Editorial Board of IEEE Transactions on Circuit and Systems for Video Technology (TCSVT), Multimedia Systems Journal, Journal of Multimedia(JMM) and Journal of Machine Visions and Applications (MVA).

Xiaoqing Ding graduated from the Department of Electronic Engineering, Tsinghua University, and won the Gold Medal for the best graduating student in 1962. Since then, she has been teaching at Tsinghua University. Now, she is a principal professor and Ph.D. supervisor. For many years, she has done research in image processing, pattern recognition, and computer vision and has received many superior achievements in various areas: character recognition, biometrics, and video surveillance. She has received many honors and has received the National Scientific and Technical Progress Award three times. She has published more than 300 papers and is the coauthor of four books. She has served as program committee member of many international conferences and editor of international journals. She is a fellow of the IEEE and fellow of the IAPR.