ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/48872330

Edge and line oriented contour detection: State of the art

Article *in* Image and Vision Computing · February 2011

DOI: 10.1016/j.imavis.2010.08.009 · Source: OAI

CITATIONS
226

READS
1,601

2 authors, including:

Giuseppe Papari
Altran
29 PUBLICATIONS **615** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Stress Free Service View project

All content following this page was uploaded by Giuseppe Papari on 11 June 2018.

The user has requested enhancement of the downloaded file.

Image and Vision Computing 29 (2011) 79–103

Contents lists available at ScienceDirect

Image and Vision Computing

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

Review article

Edge and line oriented contour detection: State of the art

Giuseppe Papari *, Nicolai Petkov

University of Groningen, Johan Bernoulli Institute of Mathematics and Computing Science, P.O. Box 407 9700 AK, Groningen, The Netherlands

ARTICLE INFO

Article history:
Received 12 March 2009
Received in revised form 21 July 2010
Accepted 18 August 2010
Available online xxxx

Keywords:
Contour detection
Preprocessing
Local pattern analysis
Contour salience
Gestalt grouping
Closure
Scale–space
Performance evaluation

ABSTRACT

We present an overview of various edge and line oriented approaches to contour detection that have been proposed in the last two decades. By edge and line oriented we mean methods that do not rely on segmentation. Distinction is made between edges and contours. Contour detectors are divided in local and global operators. The former are mainly based on differential analysis, statistical approaches, phase congruency, rank order filters, and combinations thereof. The latter include computation of contour saliency, perceptual grouping, relaxation labeling and active contours. Important aspects are covered, such as preprocessing aimed to suppress texture and noise, multiresolution techniques, connections between computational models and properties of the human visual system, and use of shape priors. An overview of procedures and metrics for quantitative performance evaluation is also presented. Our main conclusion is that contour detection has reached high degree of sophistication, taking into account multimodal contour definition (by luminance, color or texture changes), mechanisms for reducing the contour masking influence of noise and texture, perceptual grouping, multiscale aspects and high-level vision information.

© 2010 Elsevier B.V. All rights reserved.

Contents

* Corresponding author. Tel./fax: +31 0 50 363 7127.
E-mail addresses: g.papari@rug.nl (G. Papari), n.petkov@rug.nl (N. Petkov).

0262-8856/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2010.08.009

1. Introduction

Object contours play an important role in human vision. For instance, clinical evidence shows that damages in brain areas V1 and V2, which are responsible for the perception of contours, make patients completely unable to recognize objects [1]. An example of how a region boundary – often, region boundaries coincide with object contours – can influence human visual perception is depicted in Fig. 1. At the two ends of the image, the average luminance is the same; nevertheless, due to the luminance discontinuity in the middle, the left part of the image appears brighter than the right part. This shows that the presence of such a discontinuity influences the perception of the whole image and not only the immediate neighborhood of the boundary. Another example is shown in Fig. 2. The two object icons are identical in 95% of their pixels but the tiny differences in their contours lead to the perception of two objects that are semantically completely different. The importance of building up computational models for the perception of contours goes beyond the sole understanding of the human visual system. Contour detection algorithms are needed in many practical applications of computer vision, such as object recognition, robot vision, or medical image analysis.

Similar to other concepts used in computer vision, such as "texture" or "face", the notion of contour is the result of common human experience rather than a formal mathematical definition (we are not aware of any paper on face recognition in which a definition of the concept of face is given). Several authors tend to regard contours as the boundaries of interesting regions in an image. However, such a definition would exclude important situations in which contours do not arise from region boundaries. Two examples are given in Fig. 3. For the synthetic example on the left, every line in the image should be regarded as a contour, although none of the lines is a boundary between two regions of different colors or textures. In this example, in other words, the contour map coincides with the image itself. In the middle, we have a natural image and right to it a related contour map drawn by hand, both taken from the Berkeley dataset [2]. As we see, also in this case not all contours arise from region boundaries, such as the ropes of the boat. Another example is given in Fig. 4, which depicts four different ways in which contours are observed. In the first two images the contours are defined by local changes of luminance and texture, while in the third and fourth cases, global relations give rise to the perception of a contour. As we see, only in the first two cases contours arise from region boundaries. Therefore, we refrain from giving a formal definition of the term "contour". We rather assume that there would be a substantial concent among human observers who are required to draw contours in a given image. The quantitative analysis proposed in [2] shows how strong such an agreement is, showing that the concept of contour is used to refer to the same features by different observers.

Since human judgment is the only possible criterion that can be used in order to say if a given visual feature is a contour or not, we operationally define contours in a given image as *the set of lines that human observers would concent on to be the contours in that image* (one could give a similar operational definition of other concepts used in the image processing and visual pattern recognition literature, such as face). On the light of this, research in contour detection aims at understanding and modeling mathematically the features which people (consciously or consciously) use to recognize such line sets (such as contrast, good continuation, and closure).

In this paper, we propose a taxonomy of the existing contour detector algorithms which is inspired by the operational definition of contour mentioned above. Specifically, we classify contour detection algorithms according to the perceptually important features that are used in each approach. This results in two classes of algorithms: (i) *local methods*, in which the defining features are local differences of luminance, color and texture, and (ii) *global methods*, in which contours are identified based on good continuation and closure.

Contour detection is a difficult task due to various practical reasons, such as possible low signal-to-noise ratio (SNR) or presence of textures in an input image. Fig. 5 shows examples of images for which contours cannot be detected by simple low level vision models, and feedback from high-level vision cues, such as shape, is required too. This implies the necessity to interpret the input image by introducing prior knowledge (such as shape) about the contours to be detected. However, such integration of low- and high-level vision cues turns out to be a difficult task, which usually leads to very complex models and computationally demanding algorithms, which are effective only in special cases.

We make distinction between the following classes of contour detectors (i) region-oriented approaches [3,4], in which regions of constant or slowly varying texture or color are first identified; contours are then straightforwardly determined as closed region boundaries; (ii) edge and line oriented approaches [5], in which lines or boundaries defined by contrast in luminance, color or texture are detected; (iii) hybrid approaches [6], which aim at consistency between regions and region boundaries. Here, we are interested in the approaches of class (ii). This class includes a very large variety of techniques, and overview articles have been published for each specific class of methods. Examples are the analysis of the differential structure of images [7] (1994), scale space theory [8,9] (2001, 2003), statistical analysis, perceptual organization [10,11] (1993, 1999) and deformable models such as active contours [12] (1996). A general overview of the state of the art of edge detection has been proposed in

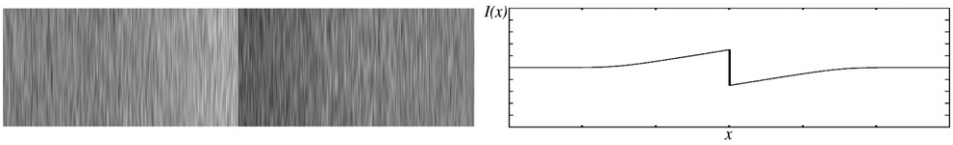

Fig. 1. Importance of boundaries in human vision: the image on the left has been obtained by adding synthetic texture to the 1D luminance profile shown on the right.

Fig. 2. Importance of contours in shape recognition: though the shown icons of a heart and an apple overlap in more than 95% of their pixels, we easily recognize them as distinct shapes due to the difference in their contours.

[6] (1998). It mainly focuses on local methods, while contextual and global techniques, such as edge grouping according to Gestalt laws or active contours, are not discussed in depth. Preprocessing aimed to reduce texture and noise in the input image, as well as quantitative performance evaluation, are also issues that have not received sufficient attention.

We acknowledge some work on region-based contour detection, i.e., image segmentation, which will not be treated in depth in this survey. Specifically, the most important techniques are based on graph theory and cuts [13,14], iterative mean shift [15], statistical methods [16,17], watershed [18,19], and multichannel analysis [20,21].

In this article, we give an overview of the main edge and line oriented approaches for contour detection that have been proposed in the last twenty years, including the progress made since the publication of the previous surveys referred to above. In Section 2 we discuss preprocessing techniques which aim at improving results of contour detection. We discuss local *edge* detectors based on differential and statistical analyses, phase congruency and morphological gradients in Section 3. In Section 4 we consider contextual and global *contour* detectors, which are based on the computation of contour saliency, edge grouping and active contours. Since the output of both local and global techniques depends on the resolution at which an image is analyzed, we review multiresolution contour detectors as well (Section 5). Finally, we describe quantitative performance evaluation procedures in Section 6 and draw conclusions in Section 8.

2. Preprocessing

This section presents contour preserving smoothers aimed to ease successive contour detection (Fig. 6). Unlike image restoration algorithms, contour preserving smoothers are meant to eliminate not only noise, but also texture, since the latter tends to disturb contour detection [22]. Local contour preserving smoothers are discussed in Section 2.1, while in Section 2.2 we consider global contour preserving smoothers based on variational methods and nonlinear diffusion.

2.1. Local adaptive filtering

The best known technique to reduce texture and noise is linear low-pass filtering. However, as edges and corners mainly have high frequency components, these features are attenuated too. In order to overcome this problem, several local nonlinear contour preserving smoothers have been designed. A general framework called *adaptive smoothing* [24] consists of computing a local weighted consists of computing a local weighted average of the gray levels on a neighborhood $\mathcal{N}(\mathbf{r})$ of each pixel $\mathbf{r}$, where the weights can depend in several ways on the local pattern configuration. The best known adaptive smoothing techniques are bilateral filtering [25], mean shift [15], value and criterion filter structure [26,27], and rank order filters (ROF), also known as Vector Order Statistics (VOS) [28].

Bilateral filtering. In bilateral filtering, the weights are given by a decreasing function of the so called range distance $d = |I(\boldsymbol{\rho}) - I(\mathbf{r})|$ between the gray level values on points $\mathbf{r}$ and $\boldsymbol{\rho} \in \mathcal{N}(\mathbf{r})$. In mean shift, the weights are derived by analyzing the local density distribution of the vector $\mathbf{z} = [\boldsymbol{\rho}, I(\boldsymbol{\rho})]^T$ and it can be shown that bilateral filtering is a special case of mean shift [29]. Both bilateral filtering and mean shift give good performance, especially if used iteratively; however, bilateral filtering is limited by the fact that the term $I(\boldsymbol{\rho}) - I(\mathbf{r})$ is very sensitive to noise [23].

Value and criterion filter structure. A different approach which does not suffer this drawback is the value and criterion filter structure. Specifically, N weighted local averages m_i and standard deviations s_i, $i = 1 \ldots N$ of the input image are computed on N possibly overlapping subregions of $\mathcal{N}(\mathbf{r})$. For each pixel, the output of the operator is given by the value of m_i that corresponds to the subregion with the minimum s_i. Well-known representatives of this class of operators are the Kuwahara and the Gauss–Kuwahara filters [27]. This mechanism allows to reduce noise and texture while preserving edges and corners and the framework includes several existing filters as special cases [30]. This class of operators is ill-posed, since it is not clear which value of m_i should be chosen in case the minimum value of s_i is reached by more than one region. A well posed value and criterion filter structure is proposed in [23].

Rank order filters. In ROF, the gray levels on points $\boldsymbol{\rho}_i \in \mathcal{N}(\mathbf{r})$, $i = 1, \ldots, N$ are ordered and the weights of local averaging depend on the rank positions in the ordering. For gray level images, the most common ordering criteria are the following:

- from the darkest to the brightest pixel;
- according to the quantity $|I(\mathbf{r}) - I(\boldsymbol{\rho})|$, $\boldsymbol{\rho} \in \mathcal{N}(\mathbf{r})$.

In the first case, several well-known filters can be obtained, such as median, weighted median, Lower–Upper–Middle (LUM) filters [31], and structural dilation and erosion. The second criterion gives rise to k-nearest neighbor denoising [32]. For vector-valued images, such

Fig. 3. Examples of contours which do not arise from region boundaries: (left) a synthetic image in which the contour map coincides with the image, (middle) a natural image from the Berkeley dataset [2], and (right) the related contour map, which illustrates that the concept of contour is broader than the concept of region boundary (see e.g. the ropes of the boat).

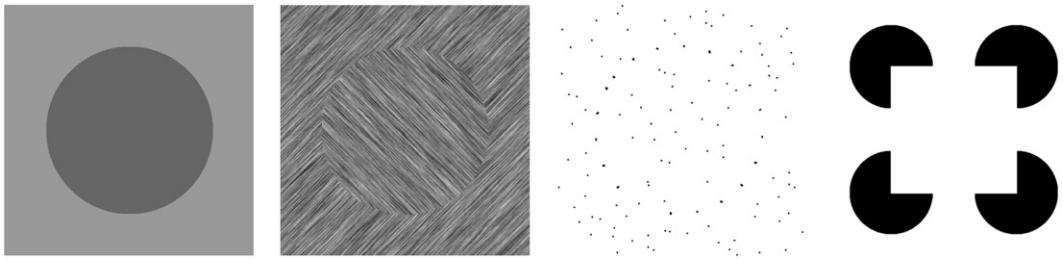

Fig. 4. Various contexts in which contours are perceived. From left to right: contours defined by luminance and texture changes, by the perceptual grouping or nearby points, and by illusory contour.

as color images, different VOS ordering criteria can be formulated [33,34]. The most common one, introduced in [35], is based on the computation of the aggregate distance $d_k = \sum_i \|\mathbf{I}(\boldsymbol{\rho}_k) - \mathbf{I}(\boldsymbol{\rho}_i)\|$ of each pixel $\boldsymbol{\rho}_k \in \mathcal{N}(\mathbf{r})$, where $\|\cdot\|$ is a norm defined on the concerned color space. d_k is high for outliers and minimum for the so called vector median. In general, ROF and VOS filters can effectively reduce both Gaussian and salt and pepper noise.

2.2. Functional minimization and nonlinear diffusion

General concepts. In this subsection we discuss contour preserving smoothers based on variational approaches and partial differential equations. We only cover the main aspects of the problem, referring to specialized literature for a more exhaustive treatment (see, e.g., [36] and the references therein). Global contour preserving smoothers can be designed by seeking the function $U(\mathbf{r})$, defined on a support Ω, that minimizes a functional of the form $J = J_1 + \lambda J_2$, with $J_1 = \int_\Omega [L(U, D^n U)\, d\Omega$, $J_2 = \int_\Omega \Lambda(I, U, D^n U)]\, d\Omega$, where $D^n U$ is the set of all partial derivatives of U up to the order n. The term J_1 is low for smooth functions $U(\mathbf{r})$, whereas J_2 is a distance between $I(\mathbf{r})$ and $U(\mathbf{r})$ (a common choice is $J_2 = \int_\Omega (I-U)^2 d\Omega$). The coefficient λ controls the tradeoff between noise rejection and data fitting. The minimization of J can be seen as the projection of the observed signal I into a given functional space, in which all functions have certain desired regularity properties. Examples are the well-known space of the functions with bounded variations, the Sobolev space, and the more recently introduced Meyer Space. We refer to [37] for a brief overview.

Tikohonov regularization. The functional $J_1 = \int_\Omega \phi(|\nabla U|) d\Omega$ has been exhaustively studied [38,39] and mathematical conditions on ϕ have been considered (see [39] for a list of the most common choices of ϕ). Important cases are $\phi(\xi) = \xi^2$, known as *second order Tikohonov regularization* [40,41], and $\phi(\xi) = |\xi|$ known as *total variation* [42]. Tikohonov regularization is also known as linear regularization since the corresponding functional can be minimized by linear filtering, but it is not robust to outliers due to the second power of $|\nabla U|$. Total variation is more robust to outliers, but it is optimal only for piecewise constant images. Thus, it may produce undesired staircase effects when applied to smooth images [43].

Mumford–Shah functional. A well-known extension of total variation is the so called *Mumford–Shah functional*: a piecewise constant image $U(\mathbf{r})$ is fit to $I(\mathbf{r})$ and a term related to the total length of the contours of the regions of constant gray level is added to J [44]. A survey of total variation methods, including several extensions of the Mumford–Shah functional, is provided in [36]. Other extensions concern vector-valued images [45,46], images defined on graphs [46], deblurring application [47,48], and decomposition of an image into three components (filtered image, texture, and noise) [37]. Numerical methods and behavioral analysis of variational approaches in image processing can be found in [41,49].

Functional minimization by nonlinear diffusion. The minimization problem $min_U J(U)$ can be solved by evolving the diffusion equation $\frac{\partial u}{\partial s} = \mathcal{F}(u)$ to a steady state, with the initial condition $u(\mathbf{r}, t) = I(\mathbf{r})$, where $\mathcal{F}$ is the Euler–Lagrange derivative (or first variation) [50]. This observation puts in connection different areas of image processing that have been considered independent of each other. Examples of diffusion equations derived by minimization of various functionals are proposed in [51,52]. On the other hand, not all diffusion equations in the form $\frac{\partial u}{\partial s} = W(u, D^n u)$, where W is a generic nonlinear function, can be derived from a variational problem. Therefore, the framework of nonlinear diffusion is more general than functional minimization.

Nonlinear diffusion has been introduced in image processing by [53] with the study of a diffusion equation in the form $\frac{\partial u}{\partial s} = \nabla \cdot [g(\|\nabla u\|)\nabla u]$, where g is a positive non-increasing function and the Gaussian and Lorentzian choices of g have been studied in detail. Due to the factor $g(\|\nabla u\|)$, which decreases with $\|\nabla u\|$, the diffusion is lower at points of high gradient magnitude, such as edges (Fig. 7). Despite its widespread use, this operator has two drawbacks:

- it is not robust to noise, especially concerning the term $g(\|\nabla u\|)$.
- The operator reinforces all edges for which $\|\nabla u\|$ is above a certain threshold T_g that depends on g, and blurs all edges for which $\|\nabla u\| < T_g$. Therefore, for edge detection purposes, it does not eliminate more texture or noise than simple gradient thresholding.

Fig. 5. Examples of images for which contour detection requires interpretation and feedback from high-level vision, taking into account global perceptual organization and knowledge of the world.

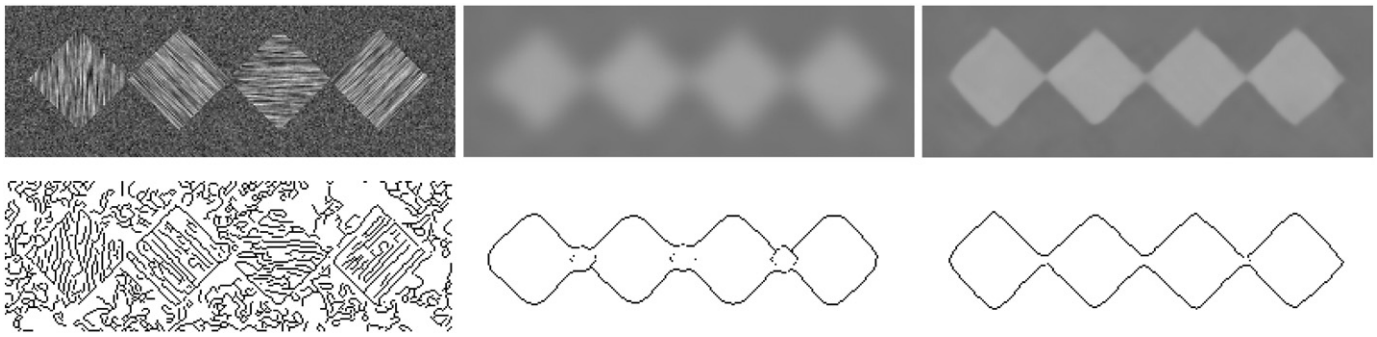

Fig. 6. Benefit of preprocessing on contour detection. First row, from left to right: input image, output of linear Gaussian filtering and output of the contour preserving smoothers proposed in [23] for the same value of σ. Second row, outputs of the Canny edge detector with the same values of the input parameters for the three images.

Concerning the first point, several regularized versions of the above diffusion equation have been proposed [54,55]. With respect to the second point, a multitude of different diffusion equations have been proposed [50,56,57]. Many of them are derived from constraints on the motion of the isophotes $u(\mathbf{r}, s) = constant$, as s increases. In particular, much attention has been paid to diffusion equations for which $W(u)$ only depends on the local curvature of the isophotes [50]. Other extensions of the nonlinear diffusion framework concern shock filters [58], vector-valued images [59,60], and a solution of the diffusion equation in terms of wavelet [61]. Connections between nonlinear diffusion and local contour preserving smoothers such as adaptive smoothing, bilateral filtering and mean shift are shown in [29] and the references therein.

This framework also offers the possibility to naturally combine image processing schemes which are useful for different complementary tasks: for instance, if M operators are expressed in terms of diffusion equations in the form $\frac{\partial u}{\partial s} = W_i(u, D^n u)$, a new operator can be obtained from the diffusion equation $\frac{\partial u}{\partial s} = \sum_i^M \lambda_i W_i(u, D^n u)$, with $\sum_i^M \lambda_i = 1$. An example is the combination of Gaussian blurring and morphological sharpening, which gives rise to a PDE formulation of the Gauss–Kuwahara filter [27].

2.3. Discussion

To summarize, preprocessing aimed at reducing texture and noise while preserving or enhancing object contours is an important aspect of contour detection. Classical local adaptive smoothing techniques, such as bilateral filtering, mean shift and value and criterion filter structure give good performance in the presence of natural texture and Gaussian noise, while VOS filters are more suitable for removing salt and pepper noise. Most of the aforementioned nonlinear filters can be incorporated in the framework of PDEs. It also offers the possibility to naturally combine image processing schemes which are useful for different complementary tasks. The PDEs formulation is very general, as it contains as special cases several local contour preserving smoothers, as well as operators based on the minimization of functionals. The main limitations of PDE approaches are computational complexity due to the iterative nature of the method and the need for an iterations stopping rule which usually is defined by heuristics.

3. Local pattern analysis

3.1. Differential operators

Continuous formulations. Discontinuities of the input luminance profile can be detected as points of high gradient magnitude. Therefore, much interest has been devoted to the differential structure of images. In this section, we use the gauge coordinates u and v [62], defined such that the lines $u = constant$ and $v = constant$ are orthogonal and parallel to the local gradient orientation, respectively. In these coordinates, the gradient magnitude is simply the first derivative $I_v \triangleq \frac{\partial I}{\partial v}$. Edges are identified as the local maxima of I_v in the v direction, therefore they satisfy the conditions $I_{vv} = 0$ and $I_{vvv} < 0$. I_{vv} is often replaced by the Laplacian $\nabla^2 I = I_{uu} + I_{vv}$ [63,64], which is linear, less computationally costly, and a good approximation of I_{vv} (in particular, the two operators coincide on zero-curvature points, for which $I_{uu} = 0$). A quantitative comparison of the edge detection accuracy of I_{vv} and $\nabla^2 I$ can be found in [65]. A biologically motivated approach based on the linear combination of the input image and its Laplacian has proved to give better edge accuracy than the sole Laplacian [66,67]. Zero-crossings of I_{vv} (or the Laplacian) which do not satisfy the condition $I_{vvv} < 0$ are local minima of the gradient magnitude and do not correspond to true edges. They are known in the literature as *phantom edges* and have been exhaustively studied [68].

Discrete formulations. As pointed out in [69], the computation of the derivatives of a digital image is an ill-posed problem. In order to regularize it, several approaches have been proposed in the literature, which are based on low-pass pre-filtering of the input. The earliest linear filtering approaches, such as the Sobel, Prewitt, Beaudet and Robert edge detectors [70], are based on measures of matching between the pixel values on $\mathcal{N}(\mathbf{r})$ and a given edge template. A common framework that includes these edge detectors as its special cases has been proposed in [71]. Canny proposed to optimize the template filter with respect to the following three criteria: good

Fig. 7. From left to right: synthetic input image and results of linear and nonlinear diffusion.

detection, good localization and low multiple responses [72,73]. He found that the optimal filter for step edge detection is very close to convolution of the input with the first derivatives of a Gaussian function. This is equivalent to computing the gradient of the input image that has first been smoothed by convolving it with the same Gaussian function (Fig. 8). The application of Canny criteria to other types of edges, such as ramp and roof edges, has been performed in [74]. Discrete version of these criteria have been formulated in [75,76].

Another approach to regularize the ill-posed problem of differentiating a function defined on a discrete support is to fit the local pattern around each pixel with a fitting function $f(x,y)$ and to compute analytically the derivatives of $f(x,y)$ [77,78]. The best known fitting function is polynomial, whose coefficients are obtained by minimizing the mean squared error. In this case, the coefficients can be computed by linear filtering and the derivatives of $f(x,y)$ in a given point are expressed as linear combinations of such coefficients [79,80]. In [81] it is shown that local function fitting is related to the local moments of an image, for which efficient algorithms are available. This fact is exploited in [81] for efficient edge detection, and in [82] a model based on the Zernike moments is proposed. In [83] cubic spline models are used for detecting roof edges, and in [84,85] a scale space based on regularized cubic spline models is proposed. If the fitting function $f(x,y)$ is not polynomial, or the fitting criterion is not the minimization of the mean squared error, the procedure is no longer linear. A recent survey of both linear and nonlinear fitting models, with applications to edge detection, can be found in [86].

The most important limitation of differential methods is that they do not distinguish between texture edges on one hand and region boundaries and object contours on the other hand. Finally, we note that the Canny method [73] is by far the most popular differential operator, which has almost become a synonym of edge detection.

3.2. Statistical approaches

The differential methods discussed in Section 3.1 are not robust to texture and are in general not able to detect boundaries defined by texture changes. In order to overcome this restriction, several statistical approaches, which analyze the local pattern around each pixel, have been proposed. The simplest technique consists in dividing $\mathcal{N}(\mathbf{r})$ in two equal parts $\mathcal{N}_\phi^+(\mathbf{r})$ and $\mathcal{N}_\phi^-(\mathbf{r})$ along a given orientation ϕ and using a two-sample statistical test of independence to measure the dissimilarity between the two halves. High values of the dissimilarity indicate the presence of a region boundary. This analysis is repeated for several directions and the one which gives rise to the maximum dissimilarity is regarded as the local contour direction. Several two-sample statistical tests have been deployed for this purpose, such as the χ^2 [87], the likelihood ratio [88], the Wilkoxon [89], the T [89], and the Kolmogorov–Smirnov test [89].

The effectiveness of these approaches for detecting texture transitions is illustrated by Fig. 9. In [90] these ideas have been extended to color images: the color distributions on the two half-neighborhoods are described in terms of compact color signatures by means of a vector quantization algorithm. The Earth Mover Distance [91] between color signatures is used to detect the presence of a contour (Fig. 10). More recently, in [2,92,93] this approach has been extended to texture. These approaches have good performance, but they are computationally demanding since the statistical analysis must be repeated for several orientations at each point. Moreover, half-circles are not appropriate in the presence of high-curvature contour points, or elongated objects which are narrower than $\mathcal{N}(\mathbf{r})$ (such as pales, wires, thin vessels, and so forth).

Other statistical approaches look at the distribution of the gradient on a neighborhood around each pixel. For instance, in [94,95], contours are detected by analyzing the covariance matrix Q of the gradient. Local contour strength and local contour orientation are derived from the eigenvalues and eigenvectors of Q (Fig. 11). Other algorithms [96,97] use only the gradient orientation inside $\mathcal{N}(\mathbf{r})$ and the presence of contours is characterized by a low angular dispersion. These differential statistical approaches are able to detect both contours and lines. They are characterized by two scale parameters: the size of $\mathcal{N}(\mathbf{r})$ and the radius of the neighborhood that is used for estimating the gradient. For contour detection applications, the two scale parameters should be approximatively equal. However, if the former is much larger than the latter, these approaches can be used for statistical analysis of directional texture [95]. Other local statistics that are deployed for contour detection are based on co-occurrence matrices [98] and the fraction of pixels whose color is sufficiently similar to the color of the central pixel [99].

In summary, algorithms based on local statistical analysis are more effective than differential methods: in fact, they detect edges determined by color and texture transitions but, at the same time, they are computationally more demanding.

3.3. Phase congruency and local energy

Phase congruency. The importance of phase in human perception of images has been demonstrated by the experiments of Oppenheim

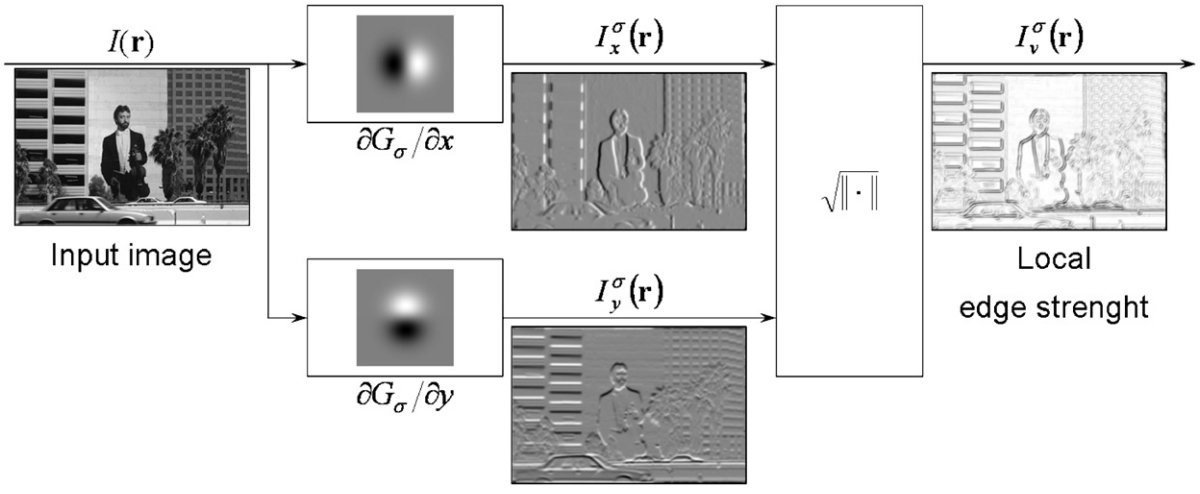

Fig. 8. The gradient of an input image I smoothed by convolution with a Gaussian function G_σ can be computed effectively by convolution of I with the x and y derivatives of G_σ. Note that the results depend on the scale parameter σ.

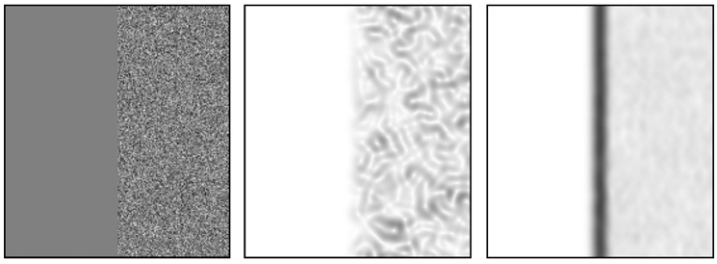

Fig. 9. From left to right: a synthetic input image of uniform average luminance, its Gaussian gradient magnitude computed as in [73], and local edge strength obtained by the statistical analysis proposed in [87]. Unlike differential methods, statistical approaches can detect transitions between differently textured areas of equal average luminance.

and Lim [100]. Further psychological evidence [101,102] indicates that the human visual system responds strongly to points in a natural image where the phase information is highly ordered. A computational model of this phenomenon has been proposed in [103], by introducing phase congruency of a 1D signal $x(t)$, defined as $PC(t) = \max_{\theta\in[0,2\pi]} \frac{\int_{-\infty}^{\infty} A(\omega)\cos[\omega t-\phi(\omega)-\theta]d\omega}{\int_{-\infty}^{\infty} A(\omega)d\omega}$, where $A(\omega)$ and $\phi(\omega)$ are the magnitude and the phase of the Fourier transform of $x(t)$. This quantity is always between 0 and 1, being 1 on those points for which all Fourier components are in phase (Fig. 12). For 2D signals, phase congruency can be computed for different directions by considering 1D projections of the image. It can be shown that such points correspond to salient visual events, such as step, peak and roof edges [104]. Therefore, edge extraction can be performed by analyzing phase congruency.

Local energy. In [103] it is shown that local maxima of $PC(t)$ correspond to local maxima of the magnitude of the analytic signal $\hat{x}(t) = x(t) + ix_{\mathcal{H}}(t)$, where $x_{\mathcal{H}}(t)$ is the Hilbert transform of $x(t)$. The computation of $x_{\mathcal{H}}(t)$ involves the entire frequency spectrum, whereas many computer vision applications require both spatial and frequency localization. Therefore, $x(t)$ and $x_{\mathcal{H}}(t)$ are replaced with their band pass versions. Thus, instead of the analytic signal, the quantity $L(t)\triangleq\sqrt{[x(t)\star f_e(t)]^2 + [x(t)\star f_o(t)]^2}$ called *local energy* is computed, where $f_e(t)$ and $f_o(t)$ are a symmetric and an anti-symmetric function, such that $f_o(t)$ is the Hilbert transform of $f_e(t)$. Several pairs of functions $f_e(t)$ and $f_o(t)$, which give rise to filters known in the literature as *quadrature pair filters*, have been studied, such as the Gabor, Log–Gabor, Gaussian derivatives, Difference of Gaussians, and Cauchy functions [105]. Other examples are robust quadrature filters [106] and optimal filters in the sense of Canny's criteria [75]. A recent comparative study of these filters has been presented in [105]. The best known quadrature filter is the Gabor energy filter, which gives the optimal compromise between spatial and frequency indetermination [107]. A recent survey of its properties and applications can be found in [108].

From 1D to 2D. The extension of these filters to the multidimensional case is not straightforward, as it is impossible to find a scalar function that is isotropic and antisymmetric. The most common way to overcome this problem is to implement anisotropic quadrature filters for different orientations and subsequently to combine the outputs with some algorithm (see [105] for a brief overview). However, these methods are typically based on heuristics and linearity is lost. More recently, an elegant generalization of the analytic signal has been proposed in [109], called the monogenic signal, which is based on the observation that *vector* functions can be both isotropic and antisymmetric.

Applications. Several edge detectors and feature extraction algorithms based on local energy and phase congruency have been proposed in the literature. In [104], a wavelet based implementation of phase congruency is proposed. In [110] it is shown that, unlike linear filtering, local energy analysis allows to identify not only step edges, but also roof edges and peak edges as well. In [111], the concept of constrained phase congruency is introduced, showing its ability to detect edges and other salient features such as Mach bands. In [112,113], local energy models are used for detecting a larger set of edge features, including junctions and line terminations. As discussed

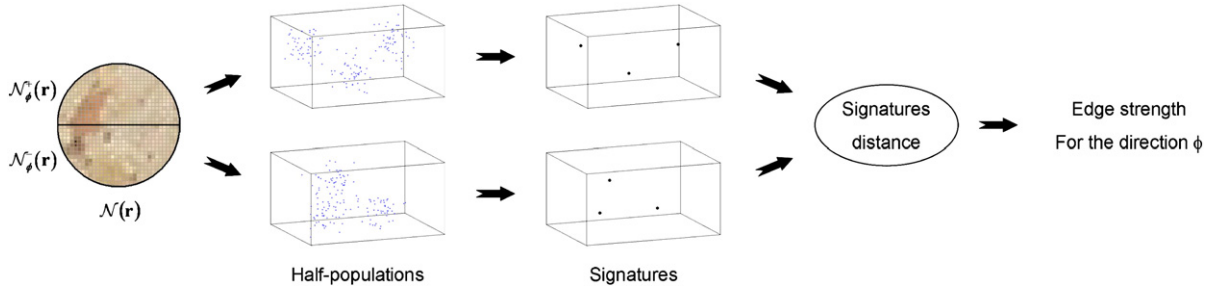

Fig. 10. Edge detection by statistical analysis of local color and texture patterns [2,91].

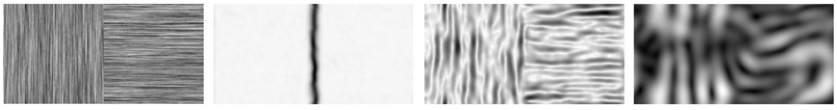

Fig. 11. From left to right: a synthetic input image and local edge strength obtained from gradient covariance, the statistical analysis of the local gray level distribution [89], and the gradient magnitude. Unlike differential methods and statistical analysis of the local luminance distribution, gradient covariance methods can detect transitions between textures of different orientations.

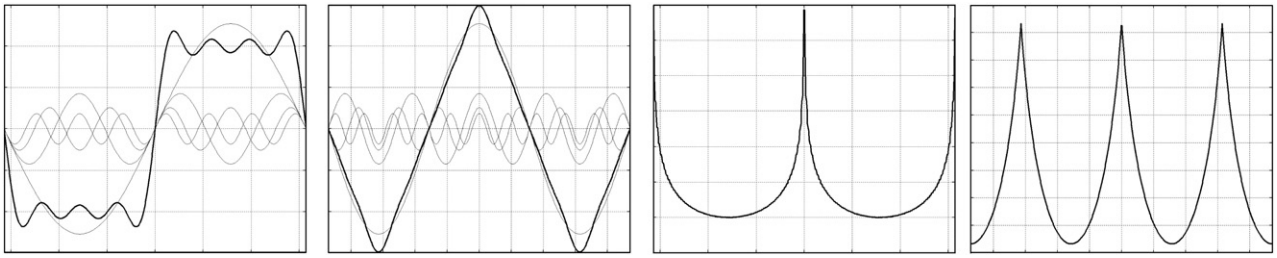

Fig. 12. From left to right: Fourier decompositions of a 1D step and roof edge, and the corresponding phase congruency plots. On the edge point, all the components are in phase.

in [114], the ability of local energy models of detecting roof, line and Mach band edges makes this operator idempotent. In other words, given an input image I, an edge detector $\mathcal{E}(I)$ that is based on local energy satisfies the condition $\mathcal{E}[\mathcal{E}(I)]=\mathcal{E}(I)$. Other development can be found in [115,116].

In conclusion, the success of local energy and phase congruency approaches is mainly due to (i) the fact that the perceptual importance of phase in images is taken into account, and (ii) their ability to detect different types of edges, such as step, ramp, roof, and line, thus making a unified framework. Moreover, phase congruency based edge detectors satisfy the idempotence condition [114]. However, for practical applications, local energy methods perform similarly to the faster and conceptually simpler differential methods [2].

3.4. VOS and morphological edge detectors

The general framework of VOS filters has been introduced in Section 2.1. In this section we show that with appropriate coefficients of the local linear combination, these filters react to discontinuities of the input luminance or color profile. The simplest example is the morphological gradient [117,118], defined as the difference between the maximum and the minimum value of I on $\mathcal{N}(\mathbf{r})$ or equivalently, the difference between the results of a dilation and an erosion: $\nabla_{morph}I(\mathbf{r}) = \max_{\boldsymbol{\rho}\in\mathcal{N}(\mathbf{r})} I(\boldsymbol{\rho}) - \min_{\rho\in\mathcal{N}(\mathbf{r})} I(\boldsymbol{\rho})$, which can be rewritten as $\nabla_{morph}I(\mathbf{r}) = \max_{\rho,r\in\mathcal{N}(\mathbf{r})} |I(\boldsymbol{\rho})-I(\mathbf{r})|$. The last equation can be straightforwardly extended to color images, by replacing the absolute value with a suitable norm in the color space (color morphological gradient [119]). This approach is fast, but not robust to outliers. In order to make it more robust to outliers, in [119] the s highest values of $|I(\boldsymbol{\rho})-I(\mathbf{r})|$ are excluded from the computation of the maximum.

Several more sophisticated filters, with a larger number of nonzero coefficients, have been proposed in [120]. These filters are robust both to Gaussian and salt and pepper noise, but at the expense of higher computational complexity. A performance comparison of these and other color edge detectors is presented in [121].

On one hand, these algorithms share some similarities with differential operators and other algorithms based on linear filtering reviewed above, because they compute linear combination of values in a neighborhood $\mathcal{N}(\mathbf{r})$; the difference is that coefficients are related to ranks rather than spatial positions. On the other hand, they are similar to local statistical analysis, by means of computing value ranks. In general, VOS edge detectors are less computationally expensive than statistical methods. The natural way in which VOS deals with vector images led to its main application to color images [120].

3.5. Combination of local features

In the previous subsections, different local features have been considered, which take into account complementary aspects of the edge detection problem. Considerable effort has been made in order to combine together the information carried out by different local features. The idea is to use the semantic information contained in available ground truths in order to train a classifier that works in the concerned feature space. The classifier returns a quantity that we call edge likelihood L (such as the posteriori probability or the Fisher discriminability [122]). Thresholding L is equivalent to classifying an edge pixel with the optimal decision boundary in the concerned feature space.

In [123] each pixel of the input image is associated to a two-dimensional feature vector, whose components are obtained by the gradient magnitude and a linear non-shift invariant template matching analysis. However, the decision boundaries are not derived from general principles. In [124], a Bayesian approach is followed in order to measure the performance improvement brought by combining the following local features: gradient magnitude, the output of a local energy analysis [110] and the Nitzberg edge strength [94]. A more exhaustive study is carried out in [2], where a larger set of local features is taken into account, including color and texture gradient. The dependence of the edge detection performance on the choice of the classifier is studied in [2,125].

3.6. Discussion

Most of the local edge features reviewed in this section are based on linear filtering, i.e. on the inner product of the local pattern $\mathcal{P}(\mathbf{r})$ surrounding each point $\mathbf{r}$ with a fixed template T. The template is designed in such a way that the output is expected to be high on several types of edges and low on flat or uniformly textured areas. However, $\langle T,\mathcal{P}(\mathbf{r})\rangle$ can be made arbitrarily high or low by simply multiplying $\mathcal{P}(\mathbf{r})$ by an appropriate factor. Thus, no matter what template T is used, a sufficiently high contrast noise will always produce higher response than a low-contrast edge. Due to these problems, some authors propose to normalize the inner product with respect to other local quantities, such as the variance of $\mathcal{P}(\mathbf{r})$ or other related quantities. However, these new features have several other limitations, such as the fact that they tend to over-react on areas where the contrast is very low. To solve these problems, a large variety of nonlinear operators for edge detection has been proposed in the literature, including the statistical approaches reviewed in Section 3.2. However, due to several limitations – such as high computational complexity, poor performance, or the necessity of a large number of ad hoc parameters – they are less popular than linear filters. We believe that this is due to the lack of a general theory which, like for the linear case, leads quite straightforwardly to a simple optimal solution.

A more serious limitation of the aforementioned local methods is that the decision of whether a pixel belongs to a contour or not is only based on a small neighborhood of each point. On the other hand, it is easy to produce images in which local patterns that are visually similar to an edge do not belong to object contours and vice-versa (see, e.g., Fig. 5). This fact has urged many authors to develop more sophisticated schemes in which global information is taken into account too, which will be reviewed in the next section.

4. Contextual and global methods

Considerable effort has been made in order to use contextual and global information in contour detection. These approaches can be

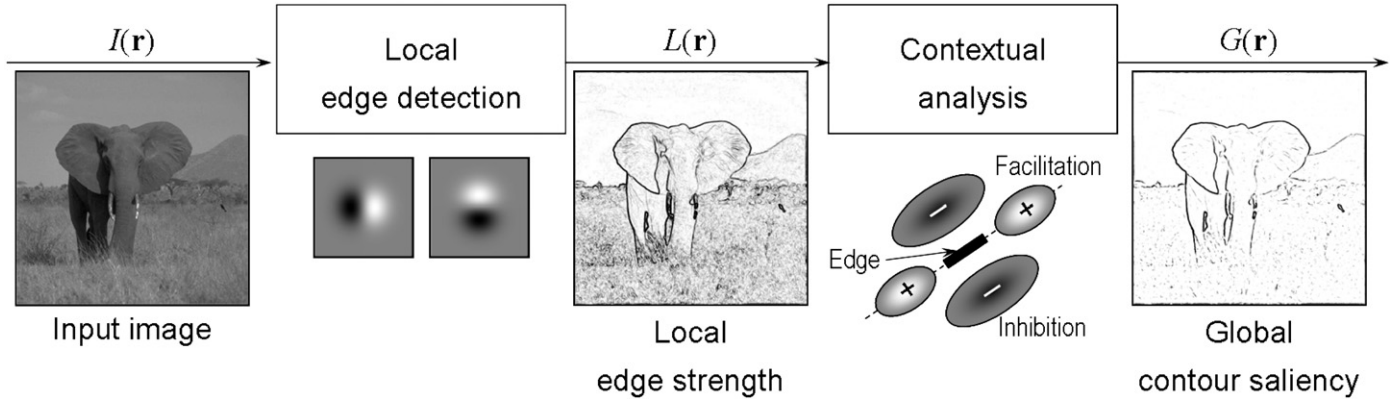

Fig. 13. Two-steps contour detection. Local contextual analysis improves the visibility of object contours by enhancing contours and reducing texture and noise.

divided in three classes: computation of the so called *contour saliency* (Section 4.1), grouping of pixels together in contours (Section 4.2) and active contours (Section 4.3).

4.1. Contour saliency

Various psychophysical and neurophysiological studies (see e.g. [126,127] and the references therein) show that the response of the human visual systems to an oriented stimulus is influenced by the presence of other similar stimuli in the surroundings. Two mechanisms have been identified:

- facilitation from stimuli which are collinear with the central one;
- surround suppression or inhibition by other stimuli.

These findings inspired several authors to perform contour detection in two steps (Fig. 13): first, local edge strength $L(\mathbf{r})$ is computed (e.g. as a gradient magnitude or response of Gabor filters); second, $L(\mathbf{r})$ is inhibited or enhanced depending on the surrounding context. The result is a more informative quantity $G(\mathbf{r})$ that is called contour saliency.

Surround suppression. A simple model of surround suppression has been proposed in [22,128]. It is based on the computation of an inhibition term $T(\mathbf{r})$ that is defined as the local average of $L(\mathbf{r})$ on a ring around each pixel. T is low on isolated edges and high on texture, thus subtracting it from the edge strength results in reduced response to texture edges and, therefore, in improved visibility of object contours and region boundaries, Fig. 14. This approach has been further developed in several ways: in [129], the annular inhibition neighborhood is split in two halves along the local edge direction in order to eliminate undesired self-inhibition of contours and decrease inhibition along region boundaries. This model has also been integrated in a multiresolution framework [130,131] and combined with a multi-thresholds approach [132]. Surround suppression can be applied as a post-processing step to the result of any of the local operators discussed in Section 3. Until now, it has been applied to differential (Canny) and local energy (Gabor) methods. The main benefit of surround suppression is the removal of texture edges and it should be used in applications in which such edges present a problem.

Facilitation. Concerning facilitation, one of the best known approaches to compute contour saliency is *tensor voting* [133]. A vector field $\mathbf{E}_{\mathbf{x}}(\mathbf{r})$, called *extension field*, is associated with each pixel $\mathbf{x}$ of the image (Fig. 15). Vector $\mathbf{E}_{\mathbf{x}}(\mathbf{r})$ is oriented along the curve that is the most likely to connect points $\mathbf{r}$ and $\mathbf{x}$ in the sense of good continuation, taking into account the edge orientations at $\mathbf{r}$ and $\mathbf{x}$. The magnitude of $\mathbf{E}$ is the value of such a likelihood and it is usually a decreasing function of $\mathbf{r}-\mathbf{x}$ and of the curvature of the arc $(\mathbf{r},\mathbf{x})$. In this way, each pixel of the image is characterized by the distribution of the values from all other pixels of the image (the so called *votes*) and a saliency map is obtained from statistics of this distribution. In [134] the covariance matrix K is considered and both saliency values and local edge orientation are obtained from the eigenvalues and eigenvectors of K. Further development of this technique can be found, for instance, in [135].

Recent work [136,137] shows that schemes similar to tensor voting can successfully model the facilitation and contour integration processes that are performed in the front-end part of the human visual system. This part of the brain is modeled as a 3D grid of neurons, which react to oriented stimuli. The first two coordinates of the position of the neuron in the grid are related to the spatial position of the edge and the third one to its orientation. Several psychophysical experiments show that the neural interaction, which give rise to both inhibition and facilitation phenomena, can be modeled by tensor voting like schemes. Special extension fields, called *association fields* in the psychophysical literature [138], are defined in order to take into account inhibition and facilitation. For an overview of the related biological background, we refer to [139,140]. Other recent approaches based on both inhibition and facilitation are presented in [141,142].

Relaxation labeling. A powerful probabilistic graph-based framework for the computation of saliency is *relaxation labeling*, introduced in [143] and successively developed in [144,145] with applications to contour detection. In its simplest formulation, relaxation labeling can be introduced as follows: let $S=\{s_1,\ldots,s_N\}$ be a set of nodes of a graph

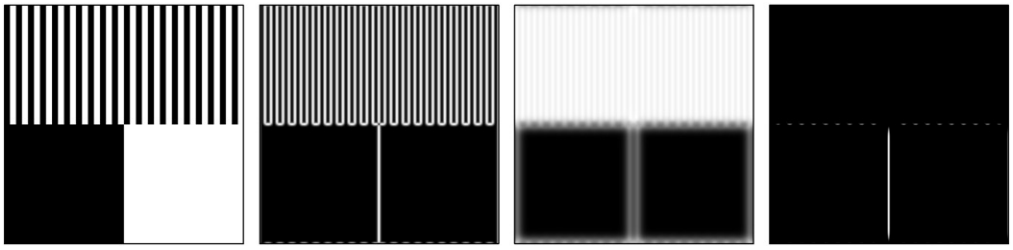

Fig. 14. From left to right: input image, gradient magnitude, inhibition term computed as local average of the gradient magnitude, and final edge strength. The inhibition term is high on texture edges, therefore texture edges are suppressed in the final result.

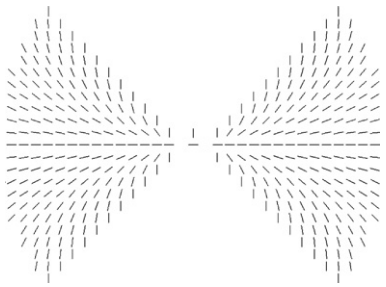

Fig. 15. Extension field as computed in [134].

G, and $L=\{l_1,\ldots,l_M\}$ be a set of labels; an *a priori* subset $L_k \subseteq L$ of possible labels is also associated with each node s_k. A *labeling* is an application $\mathcal{L}: S \mapsto L$ which assigns one label $\mathcal{L}(s_k) \in L_k$ to each object $s_k \in S$. Let us now introduce a *compatibility restriction* R on the class of all possible labelings $\mathcal{L}$. Specifically, a labeling $\mathcal{L}$ is admitted by $\mathcal{R}$ iff the pair $\{\mathcal{L}(s_i), \mathcal{L}(s_k)\}$ belongs to a given set $L_{i,k}$ $L_i \times L_k$, for every pair (s_i, s_k) of nodes which are joined by an edge in the graph $\mathcal{G}$. In general, a compatibility relationship reduces the number of possible labelings, as illustrated in Fig. 16.

The main limitation of such a binary formulation is that the compatibility relationships are often either too strict, so that no labeling is allowed, or too permissive, as in Fig. 16, where more than one labeling are admitted. Therefore, a fuzzy modification of such a framework is often preferred, in which a labeling $\mathbf{p}$ is a matrix of numbers $p_k(\alpha)$, with $k=1,\ldots,N$, and $\alpha=1,\ldots,M$, which represent the degree of confidence with which label l_α is assigned to object s_k; these numbers must satisfy the constrains $\sum_{\alpha=1}^{M} p_k(\alpha)=1, \forall k$ and $p_k(\alpha) \geq 0, \forall k, \alpha$. The numbers $p_k(\alpha)$ are often but not always interpreted as probabilities. Compatibility relationships are expressed in the form of a real-valued function $r_{i,k}(\alpha,\beta)$; it represents the degree of compatibility of labeling objects s_i with l_α and s_k with l_β. Compatibilities are often supposed to be given, though for some applications they are learned from input data [146]. With these quantities, it is possible to introduce the so called average consistency $a(\mathbf{p})$ associated to a labeling $\mathbf{p}$, defined as $a(\mathbf{p}) \sum_{i,k} \sum_{\alpha,\beta} r_{i,k}(\alpha,\beta) p_i(\alpha) p_k(\beta)$; it measures how much the labels assigned to each node agree with each other.

An interesting probabilistic interpretation of the framework described above, presented in detail in [147], links the theory of relaxation labeling to the theory of optimization of Markov random fields on graphs [148]. In particular, in [147] it is shown that when the quantities $p_k(\alpha)$ are interpreted as probabilities, and the $r_{i,k}(\alpha,\beta)$ are interpreted as potentials, then the average consistency $a(\mathbf{p})$ defined above is proportional to the energy of the Gibbs distribution related to the concerned Markov model. Thus, the main problem of relaxation labeling is to find the labeling $\mathbf{p}$ which maximizes $a(\mathbf{p})$, in the MRF formulation, it corresponds to find the configuration that corresponds to the maximum a posteriori probability. When an unambiguous labeling is required, the further constrain $p_k(\alpha) \in \{0,1\}$ is imposed to the minimization.

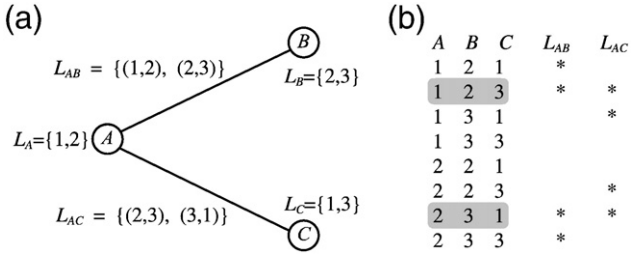

Fig. 16. Illustration of relaxation labeling with a simple example with three nodes $S=\{A,B,C\}$ and three labels $L=\{1,2,3\}$. (a) Graph representation, with the *a priori* sets L_A, L_B, L_C, and the compatibility relationships L_{AB}, L_{BC}. (b) List of all possible *a priori* labelings, where the asterisks on the fourth and fifth columns mark the combinations admitted by L_{AB} and L_{BC} respectively. The only labelings which are admitted by both compatibility relationships are (1,2,3), and (2,3,1), which are printed on a gray background.

Such a maximization problem is NP-hard, i.e. it cannot be solved in polynomial time [149]. The classical approach to solve relaxation labeling problems is to start with a reasonable initial labeling $\mathbf{p}_0$ and to use some iterative gradient-descent-like scheme to maximize $a(\mathbf{p})$ [150,151]. However, such a simple approach is generally slow due to the large number of iterations required to reach an optimum. Much effort has been made in order to develop more efficient algorithms and to study theoretical conditions under which convergence to the optimum is achieved with a finite number of steps. The most important methods are graph diffusion [152], belief propagation [153], convex combination of trees [154,155], max-sum diffusion, and the augmenting directed acyclic graph algorithm [149]. Most of these methods try to minimize of an upper bound of $a(\mathbf{p})$, which turns out to give better performance in terms of speed, with respect to the direct maximization of $a(\mathbf{p})$. However, their main drawback is that it is not theoretically guaranteed that the point at which they converge is the desired optimum.

In the context of contour detection, relaxation labeling is applied as follows: First a graph is constructed whose nodes are the pixels of the input image and whose arcs link neighboring pixels (according to a given notion of neighborhood). To each pixel, a label must be assigned (which can be binary, real or vector-valued), indicating how confidently a given point can be regarded as part of an object contour. The compatibility relationships $r_{i,k}(\alpha,\beta)$ convey contextual information; specifically, $r_{i,k}(\alpha,\beta)$ is high for pairs of collinear edges and low for randomly placed edges. Therefore, the functions $p_k(\alpha)$ which maximize $a(\mathbf{p})$ can be interpreted as a global saliency of a contour. In practice, the labels are initialized by using local edge information, and iteratively updated in order to reinforce the underlying object contours and suppress undesired responses. At each iteration, each label interacts with the ones in its immediate surroundings, and after a sufficient number of iterations the value of each label is determined by the values of all other labels. In this way, global contextual information is taken into account. Recent improvements concern the introduction of curvature [156] and texture [151] information in this framework.

To summarize, the two frameworks of tensor voting and relaxation labeling deliver similar algorithms, as both of them boil down into iterative schemes in which, at each iteration, an initial local edge strength is augmented in the presence of long chains of collinear edge pixels. One aspect in which relaxation labeling is certainly superior to tensor voting is that the former faces the problem of contour detection as an optimization problem and relies on a much stronger theoretical background. Consequently, while tensor voting may converge to some local minima far from the optimum, relaxation labeling can benefit of annealing procedures that are available for Markov random fields optimization, which help in getting closer to the global minima.

4.2. Grouping pixels into contours according to Gestalt principles

Edge pixels can be grouped together into contours, according to the Gestalt laws of proximity, good continuation and, more recently, closure [157] (Fig. 17), and symmetry [158]. Graph theory is a natural mathematical framework for this purpose: nodes of a graph $\mathcal{G}$ correspond to edge pixels and interconnections between nodes model potential links between contiguous edges. A contour $\mathcal{C}$ is represented by a subgraph of $\mathcal{G}$ and it is detected by minimizing a cost function C computed over $\mathcal{G}$. In this section, we will use the term *contour fragment* to indicate the connected components of such a subgraph. The cost function C is usually defined as the sum over $\mathcal{C}$ of *node prominence* terms, which are decreasing functions of the local edge strength, and *interconnections affinity* terms, which take into

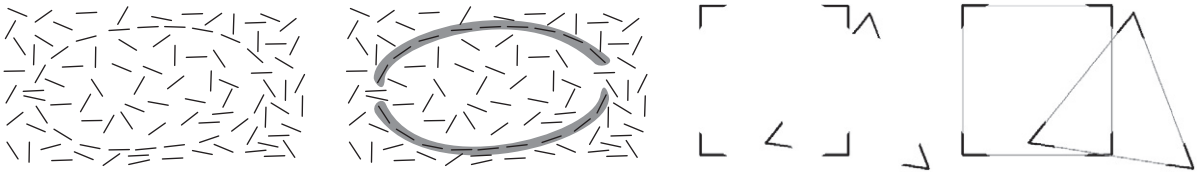

Fig. 17. Grouping according to the Gestalt laws of (left) good continuation and (right) closure.

account collinearity and co-circularity between pairs of adjacent edges [159–161].

The open issues regarding this approach are two-fold:

- Though the mathematical problem of partitioning a graph by minimizing a cost function has been formally solved, it is computationally intractable. Therefore, a large variety of suboptimal minimization algorithms have been proposed, including methods based on the computation of the minimum spanning tree [162] and the Delaunay graph [162,163].
- Since Gestalt principles only provide qualitative statements, their translation into a cost function is arbitrary.

Various cost functions have been proposed in the literature, including geometrical [160] and probabilistic [164] models. More recently, also corroborated by psychophysical evidence [165,166], contour closure is taken into account as well in defining such cost functions [162,164,167,168]. Contour closure approaches tend to take into account more global information and are able to detect illusory contours [169].

Closure. In general, graph-based grouping algorithms which take closure into account identify the subgraph $\mathcal{G}'$ of $\mathcal{G}$ which minimize a certain cost function, with the mathematical constraint that $\mathcal{G}'$ is a cycle. In some more advanced techniques, additional constraints are imposed on $\mathcal{G}'$, such as convexity [170] or the rotation index to prevent self-crossing cycles [171]. For instance, in [171] a sparsely connected graph $\mathcal{G}$ is first computed from the edge map, by including arcs between contiguous edges which are likely to belong to the same contour in the sense of the Gestalt laws of proximity and good continuation. Then, the shortest non-self-crossing cycles are detected and regards as object contours (thus, the cost function is simply the number of pixel of a contour). The method is not very fast, as its computational complexity is $O(n^2 log n)$, where n is the number of edge pixels. A more sophisticated approach based on Markov models has been proposed in [164]. Specifically, a transition matrix $\mathbf{P}$ is first computed, where $P_{i,j}$ is the conditioned probability that a path that contains the edge pixel i will also contain the edge pixel j. These probabilities are computed according to the Gestalt laws of proximity and good continuation by means of the theory presented in [172]. It is shown that the eigenvalues of $\mathbf{P}$ can be regarded as a saliency measure, and that cycles can be easily detected from the eigenvectors of $\mathbf{P}$ as its strongly connected components.[1] This method detects contours more accurately than the one presented in [171] and it is faster, as its computational complexity is $O(n^2)$.

A further increase in terms of both speed and accuracy can be achieved by the algorithm called *ratio contour* proposed in [174]. Rather than building a probabilistic model of the edge distribution, a simple cost function is minimized, based on the normalized length of the gaps between contour fragment and on the total curvature of the detected boundary; this cost function tends to favor smooth boundaries with short gaps. Both the length of the gaps and the curvature term are normalized with respect to the number of pixels in a given connected component of $\mathcal{G}'$, in order to prevent bias toward small objects. A parameter λ controls the relative importance of the curvature term with respect to the gaps length term. Contours are detected as follows: similarly to [164,171], a graph $\mathcal{G}$ is first constructed by linking edge pixels according to the Gestalt laws of proximity and good continuation. Then, the cycles which minimize the aforementioned cost function are detected in $\mathcal{G}$ and regarded as object contours. This method is faster than the other two referred above, with a computational complexity equal to $O(n^{7/4})$, and detects contours more accurately. However, its higher performances are due to a fine tuning of the parameter λ for which there is no theoretical justification.

Symmetry. In more refined schemes, symmetry is also taken into account by seeking for groups of edge pixels which exhibit some forms of global or local symmetry around a possibly curved symmetry axis. In order to define the local symmetry axis of a shape $S \in R^2$, the concept of *skeleton* is introduced, which is commonly defined as the *medial axis* m of S. Specifically, m is defined as the loci of centers of circles which are tangent to the boundary of S in at least two points, without touching the inner part of S (see [175,176] for an overview of the subject). By definition, points of the medial axis are equidistant from the shape boundary, thus detecting local symmetries as illustrated in Fig. 18 (left). However, skeletonization algorithms are well known to be extremely unstable with respect to perturbation of the input shape S, thus giving rise to many additional spikes departing from the desired skeleton, as shown in Fig. 18. Much research is going on in order to provide a well posed definition of skeleton [177–179].

A graph-based approach to group edge pixels according to the above mentioned definition of symmetry has been presented in [158]. A new cost function is introduced as the ratio between two terms: the numerator measures the degree of symmetry of seeked symmetric region R, while the denominator is equal to the area of R. In order to guarantee the symmetry of the detected structures, a new grouping token is introduced, defined as a symmetric trapezoid constructed from previously detected edge segments. Thus, the grouping algorithm identifies a closed boundary as the sequence of trapezoids that minimizes the aforementioned cost function. This method is able to identify the boundary of symmetric objects in polynomial time, and it is not biases toward smaller objects due to the denominator of the cost function. However, it is still quite slow, since its computation time is $O(n^{5.5})$, where n indicates the number of line segments. A faster method has been presented in [180], whose computational complexity is $O(n^2)$. However, it is biased toward small objects and it is not optimal in the sense of an elegant and easily interpretable cost function. More elaborated approaches for symmetric object detection have been proposed in [181,182] which differently than the methods referred above, use additional texture information to identify objects which are symmetric in the 3D world, but which do not result symmetric in the image due to the perspective skew.

In contrast to many of the methods considered in the previous subsection, which deliver only a binary map of edge pixels that are unrelated with each other, the grouping methods referred here produce groups of pixels, with the members of one group belonging to the same contour. In this way they provide information that is more useful for object recognition. However, the main disadvantage of the aforementioned techniques is their computational complexity. Much research has been carried out in order to provide fast algorithms. Examples are morphological approaches, which basically tend to fill the gaps between contour segments by *directional dilation* [183,184]

[1] A strongly connected component $\mathcal{C}$ of a directed graph $\mathcal{G} = (V, A)$ is a subgraph of $\mathcal{G}' = (V', A')$ such that whenever node i is reachable from node j in $\mathcal{G}'$, then node j is reachable from node j too, with $i,j \in V'$ [173].

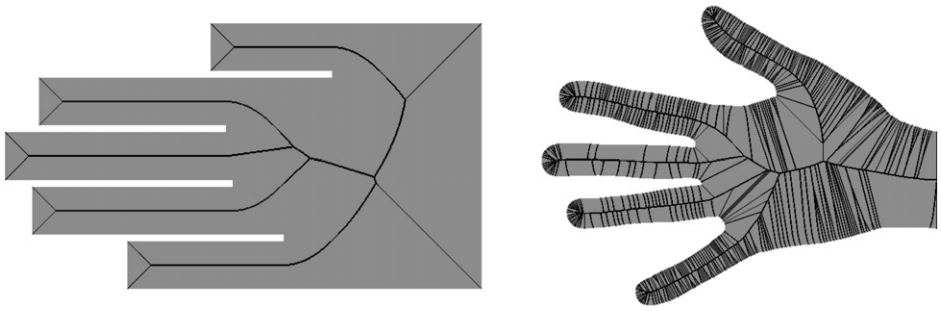

Fig. 18. A (left) synthetic and (right) real shape of a hand, with the corresponding medial axis.

and clustering algorithms based on some distance function between edges, which takes into account both their proximity and their collinearity and co-circularity [185,186]. For an overview of edge grouping algorithms and a recent discussion of the properties that a grouping algorithm should have we refer to [10,187].

4.3. Active contours

General concepts. Active contours, also known in the literature as *snakes* or *deformable models*, have been introduced in [188] as a variational approach to contour localization and interactive contour detection.

The idea is to evolve a curve C, drawn by the user around an object $\mathcal{O}$, by the minimization of an energy functional E (Fig. 19). Typically, the energy E is given by the sum of two terms: internal energy E_{int} that measures the smoothness of C, and external energy E_{ext} that takes into account how close C is to the actual contour of $\mathcal{O}$. In the standard formulation, we have $E_{int} = \oint_C(\alpha\|\dot{\mathbf{r}}\| + \beta\|\ddot{\mathbf{r}}\|)dl$ and $E_{ext} = \oint_C g(\|\nabla I\|)dl$, where α and β are input parameters and g is a decreasing function such that $g(\xi)\to 0$ when $\xi\to\infty$. Usually, the form $g(\xi) = (1+\xi^p)^{-1}$ is chosen, with p a positive parameter. The two terms of E_{int} are related to the length and the total curvature of C respectively. On the other hand, E_{ext} is low when C is located on maxima of the gradient magnitude. For $\beta = 0$, it is possible to prove that the functional referred above can be rewritten as $L_g(C) = \int_C \sqrt{\sum_{i,k} g_{i,k}(\mathbf{r})dr^i dr^k}$, where $r^i(s)$, $i = 1,2$ are the components of the vector $\mathbf{r}(s)$ and the tensor $g_{i,k}(\mathbf{r})$ is a function of the input image $I(\mathbf{r})$ (see [189] for details). $L_g(C)$ is interpreted as the geodesic length of the curve C in a Riemannian space[2] whose metric is determined by the input image. This important result links together snakes theory and geometric models, which are by their nature topologically independent [189].

Active contour algorithms can be categorized into two types, depending on how the curve $\mathcal{C}$ is represented. In *parametric snakes*, a curve is expressed by its parametric equation $\mathbf{r} = \mathbf{r}(s)$, with $s\in[0,1]$, and $\mathbf{r}(0) = \mathbf{r}(1)$. The variable s represents the ratio between the arc length of the portion of curve between points $\mathbf{r}(0)$ and $\mathbf{r}(s)$ and the length of the whole curve. In *level set snakes*, C is represented by a function $\Psi(\mathbf{r})$ defined such that $\Psi(\mathbf{r}) = 0$ on C [190]. The main difference between these two formulations is in dealing with topology changes: with parametric snakes, the curve C preserves its topology during its evolution, unless complicate split and merge procedures are integrated in the algorithm. Level set snakes are more suitable for topology changes, which is appealing in situations in which the number of the objects that must be detected is not known in advance. However, this makes the method less robust to noise because snakes can oversplit. Therefore, topology preserving level set snake models have been proposed as well [191,192]. For a recent survey of level set snakes, we refer to [193].

With respect to the local edge detectors discussed in Section 3, snake models have several advantages, such as the possibility to impose constraints of continuity, smoothness and closure of the detected contours. Moreover, these techniques are able of detect illusory contours [188], an impossible task for local edge detectors. On the other hand, the minimization of E has several shortcomings, such as strong dependence on the initial conditions, slow convergence, and little ability to detect low-contrast contours. In order to overcome these disadvantages, a large variety of modifications of the functional E have been proposed in the literature. Examples are Fourier snakes [194], finite elements snakes [195], dual snakes [196], fuzzy tracing [197,198], multiresolution approaches [199], and models inspired by electrostatic principles [200]. In [201], an external energy that depends on the edge direction has been proposed. Active contour models for color images have been presented in [202]. Recently, most of these techniques have been unified in a framework called *United snakes* [203]. For an overview, we refer to [204,205].

Diffusion equation associated to active contours. An important development concerns the equivalence between the minimization of E and the solution of a diffusion equation in the form $\frac{\partial \mathbf{r}}{\partial s} = \frac{\partial}{\partial s}\nabla E_{int} + \mathbf{F}_{ext}$, where the *driving force* $\mathbf{F}_{ext} = -\nabla E_{ext}$ attracts a snake to an object contour. This formulation becomes more general than minimizing E as soon as the external force is not constrained to be the gradient of some energy function. A relevant example of this class of models is the so called gradient vector flow [206,207]. In [206], the external force is defined as the final state of another nonlinear diffusion process by taking $-\nabla E_{ext}$ as initial condition. The main advantage of GVF with respect to standard snake models is a better ability to move snakes into boundary concavities.

Region-based active contours. A limitation of most of the aforementioned approaches is that E_{ext} is computed by means of the gradient magnitude. The disadvantages of using this quantity are twofold:

- $|\nabla I|$ takes low values on low-contrast contours, thus making the snake collapse [208].
- $|\nabla I|$ can be strong in the presence of noise, thus making the snakes be attracted by undesired local minima [209].

These problems could be overcome by introducing balloon forces which prevent the collapse [195], or by replacing the gradient magnitude with a more sophisticated contour saliency [209]. However, a much more effective approach is to compute E as a sum of integrals over the *regions* inside C, rather than over their boundaries. Region-based snakes, [210–212], are more robust to image degradations and less sensitive to initialization, because more global statistics are involved with respect to contour-based snakes. Moreover, region-based snakes can be more naturally integrated with texture models [213].

[2] A Riemannian space $\mathcal{R}$ is a metric space in which the infinitesimal distance between points $\mathbf{r} = (x_1,\ldots,x_n)$ and $\mathbf{r} + d\mathbf{r} = (x_1 + dx_1,\ldots,x_n + dx_n)$ is given by $ds \triangleq \sqrt{\sum_{i,k} g_{i,k}(\mathbf{r})dx_i dx_k}$. The tensor $g_{i,k}(\mathbf{r})$ completely defines the metric of $\mathcal{R}$. For $g_{i,k}(\mathbf{r}) = \delta_{i,k}$, $\mathcal{R}$ reduces to the ordinary Euclidean space.

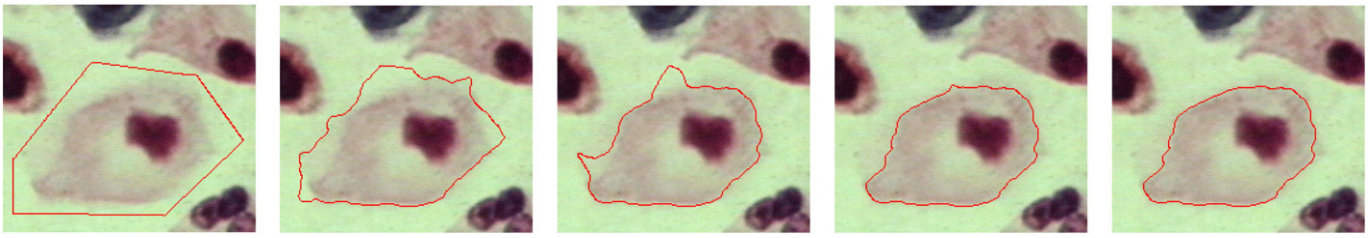

Fig. 19. Contour determination by snakes. From left to right: hand-drawn initial curve and results after $n = 4$, 8, 12, 16 iterations, respectively.

By their nature, region-based snakes have a close connection with the Mumford–Shah functional [44]. In [211], the input image is assumed to be a noisy version of piece-wise constant luminance profile $U(\mathbf{r})$, which only takes the two values c_1 and c_2 over two not necessarily connected regions Ω_{in} and Ω_{out}. The boundary C of Ω_{in} is detected by the minimization of a functional in the form $E = E_{int} + E_{ext}$: E_{int} is a linear combination of the length of C and the area of Ω_{in}; E_{ext} is the weighted variance of the fitting error, expressed as $\lambda_1 \iint_{\Omega_{in}} |I(x,y) - c_1|^2 dxdy + \lambda_2 \iint_{\Omega_{out}} |I(x,y) - c_2|^2 dxdy$, with weights λ_1 and λ_2. The weights λ_1 and λ_2 represent the costs which are paid when the input image $I(x,y)$ deviates from the constant values c_1 and c_2 in Ω_{in} and Ω_{out} respectively. We observe that the driving forces, rather than seeking for the points with maximum $|\nabla I|$, tend to move the snake toward the configuration that globally maximizes the homogeneity of Ω_{in} and Ω_{out}. These ideas have been extended to multilevel [214] and color [215] images as well. In general, region-based snakes have better segmentation performance than contour-based snakes and are less dependent on the initialization. However, they are computationally more demanding because of the computation of double integrals.

Shape based active contours. Finally, we briefly discuss the introduction of shape analysis in the context of active contours. The idea is to add to E a term E_{shape} that privileges shape similarity between C and a set of training objects [216,217]. We assume that the training shapes have been aligned to each other with respect to scaling, rotation and translation in order to maximize their overlapping [217]. The term E_{shape} is computed by the following two steps: first, vectorization of all shapes. For active contours applications, the most popular approach consists in computing vectors whose components are the values of the so called *signed distance function* [218] of the corresponding shape.[3] With respect to other vectorization algorithms [219], this one is numerically stable, can successfully deal with high-curvature points and topological changes [217], and has a natural implementation in the framework of level set. The second step is the estimation of a probability distribution p of the training shapes in the shape space. E_{shape} is then computed as $-logp(\mathbf{v}_C)$, where $\mathbf{v}_C$ is the vector associated to the shape of the snake C. This term is minimal when the evolving snake C takes the most frequent shape in the training database.

Shape-based active contours models are frequently used in combination with the Mumford–Shah functional [220,221]. In this respect, the assumption of Gaussianity for p has the advantage of convexifying the Mumford–Shah functional. On the other hand, such an assumption is too simplistic, and more sophisticated distributions have been proposed as well [220].

As long as the shape of the object to be detected is sufficiently similar to the observed training shapes, the addition of the term E_{shape} brings significant improvement, especially in detecting complex objects. However, its main limitations are the high dimensionality of the shape space. Typically this is solved by linear principal component analysis, but the signed distance function is not closed with respect to linear operations [217]. Therefore the linear PCA is not the optimal solution and this remains an open issue in the field of shape-based active contours.

[3] The signed distance function of S is defined as the product between the distance transform of the contour of S, and a term which is equal to -1 inside S and to 1 outside S.

4.4. Discussion

The contextual and global contour detectors reviewed in this section can be classified into two categories: (i) computing a contour saliency by taking into account some contextual information and then extracting edges by thresholding the saliency, and (ii) directly identifying meaningful structures.

Concerning the former, the most important areas of research are tensor voting and relaxation labeling. Both of them can be regarded as mathematical models of the inhibitory and facilitatory phenomena that happen in the human visual system. In particular, the vector and tensor fields involved in tensor voting schemes have a biological counterpart called association fields, which plays a central role in human vision. As for relaxation labeling, the nodes and the edges in the graph $\mathcal{G}$ can be interpreted as neurons and synaptic interconnections respectively, while the evolution of the labeling to a compatible state effectively models the firing mechanism that happens in the neurons of the human visual system [143]. With this respect, it should be noted that in the original formulation, only the facilitatory mechanism was taken into account, while inhibition was included later [222].

As to the latter, the identification of meaningful structures directly from the input image is generally formulated as an optimization problem. We classify them into two categories of algorithm depending on whether they rely on a continuous or discrete formulation. In the first case, we have the variational approaches reviewed in Section 4.3, in which a curve evolves according to the Euler–Lagrange diffusion equations associated to the concerned functional. The second class, discussed in Section 4.2, comprises techniques for edge grouping according to Gestalt principles, where the optimization is performed in a graph-based framework by means of dynamic programming techniques. Variational active contours are certainly more popular than the Gestalt grouping approaches for several reasons:

- computer vision problems can be more easily and elegantly be formulated in a continuous fashion with respect to a discrete one;
- differential and variational calculus provides more powerful tools than graph theory does;
- the active contours framework is top-down, while Gestalt grouping is bottom-up;
- the active contours framework has been successfully deployed in fields related to contour detection, for which Gestalt principles are inapplicable (such as e.g. object detection from 3D models).

On the other hand, algorithms based on edge grouping are intrinsically unsupervised, while most of the existing active contour techniques require an initialization from the user. While much research has been made to achieve insensitivity with respect to the initialization (see e.g. [200,223]), it is still unclear how robust those

Fig. 20. Different types of blurring in natural images. (left) Focal blurring, (center) penumbral blurring, and (right) shading. See [226] for a discussion.

methods are, especially for general purpose contour detectors in the presence of texture and noise.

Both in the discrete and the continuous case, contour detection based on the optimization of a given cost (objective) function(al) is very powerful, as it effectively uses global information, and it is very versatile, as the cost function can be adapted for different applications. In particular, it is possible to modify the cost function in order to include some prior information about the detected objects, such as shape or symmetry. On the other hand, the main drawback of this approach is that as soon as the cost function is non-convex, a local minimum might be reached instead of the global one. This problem becomes particularly serious for high dimensional searching spaces and for very complex cost functions, for which the number of undesired local minima may increase very rapidly.

5. Multiresolution methods

Many of the operators discussed in the previous sections have a scale or resolution parameter, which is related to the size of the neighborhood $\mathcal{N}(\mathbf{r})$ that is considered for detecting local features. In this section, we discuss multiresolution analysis. Its importance in contour detection is two-fold:

- natural images contain edges with different levels of blurring (Fig. 20);
- there is psychophysical evidence that the human visual system analyzes separately different portions of the frequency domain [224,225]. Since edges are operationally defined based on what human observers perceive, the multiresolution aspect cannot be neglected.

Subsection 5.1 overviews general facts about multiresolution analysis, while specific multiresolution contour detectors are discussed in subsection 5.2.

5.1. Multiresolution analysis

Scale–space. A natural mathematical framework for multiresolution analysis is the wavelet theory [227–229]: in the linear case, the input image $I(\mathbf{r})$ is convolved with a family of scaled versions $\psi_s(\mathbf{r})$ of a fuction $\psi(\mathbf{r})$ that is called *mother wavelet*; s is the resolution or scale parameter. The outputs of these convolutions give rise to a 3D object $I(\mathbf{r},s) = I(\mathbf{r}) \star \psi_s(\mathbf{r})$ that is called *linear scale space* (Fig. 21). The best known mother wavelet $\psi_s(\mathbf{r})$ is the 2D Gaussian function, due to several mathematical properties (such as isotropy, separability, entropy minimization, no zero-crossing creation [230], solution of the linear diffusion equation $\frac{\partial I}{\partial s} = \nabla^2 I$, monotonicity of local maxima [231], semi-group property, and sub-optimality for noise rejection [73]).

Non-Gaussian linear scale–spaces. Until recently, it was generally believed that the Gaussian function was the only one for which such mathematical properties hold, and several axiomatic formulations of the scale space theory have been proposed (see [232,233] for an overview). However, as pointed out in [234,235], non-Gaussian scale spaces compatible with the above mentioned axiomatic formulations are possible as well, as soon as the smoothing kernel is no longer required to have a finite variance. Relevant examples are the monogenic scale space [236,237], the Bessel scale–space [238], the relativistic scale–spaces [239], the Gabor scale space,[4] and the α-scale space [240,241] (Fig. 22). The α-scale space satisfies the linear diffusion equation $\frac{\partial I}{\partial s} = -\left(-\nabla^2\right)^{\alpha} I$. For $\alpha = 1$ and $\alpha = 1/2$, this equation reduces to the Gaussian and the Poisson cases respectively. For other values of α, the wavelets exhibit a more complicated pattern of excitatory and inhibitory regions, similar to the ones present in the classical center-surround receptive fields of lateral geniculate nucleus cells and retinal ganglion cells of the mammals visual systems [234,242]. This fact makes the α-scale space attractive for modeling of the front-end visual system of the brain.

Nonlinear scale–spaces. For high values of the scale parameter s, $I(\mathbf{r},s)$ is robust to noise in the input image $I(\mathbf{r})$ because of high blurring. This fact makes scale–space theory attractive for regularization and inverse problems [243,244] (see also [9,245] for a survey). On the other hand, high noise rejection is paid in terms of edge smearing and the problem cannot be overcome in the linear framework. Therefore, several nonlinear scale spaces have been introduced. Important examples are affine-invariant scale spaces [57], morphological scale spaces [8,246,247], variational scale–spaces [245] and Lee-theoretic scale spaces [248], which are derived from nonlinear diffusion equations, and scale spaces derived from nonlinear wavelets [249–251]. For a survey and a discussion of these scale spaces, we refer to [252–254].

5.2. Contour detection in the scale space

The behavior of edges in the scale space has been exhaustively studied [255,256]. It is illustrated in Fig. 23 for a noisy 1D signal $I(x)$, for which the Gaussian scale space $I(x,s)$ is computed and the edges are detected as the zero-crossings of $\frac{\partial^2}{\partial x^2} I(x,s)$. As the scale parameter increases, the zero-crossings of the Laplacian that are due to texture and noise disappear. On the other hand, the coarser the scale, the worse is the edge localization capability of the zero-crossings of the Laplacian.

[4] In the 1D case, the Gabor scale space is obtained by convolving the input signal $I(x)$ with the complex function $g_{\sigma,\lambda}(x) \propto e^{-\frac{x^2}{2\sigma^2} + \frac{2\pi x}{\lambda}}$. Strictly speaking, this is scale space in the axiomatic sense if and only if λ is constant and the scale parameter σ varies between 0 and ∞. However, it should be noted in many multiresolution applications with Gabor filters, the parameter λ is made proportional to σ, which violates the semigroup property of scale spaces. Similar considerations hold for the multi-dimensional case.

Fig. 21. Example of the Gaussian scale–space: an input image and several outputs of the convolution with Gaussian functions G_σ, with $\sigma = 2, 4, 8, 16$, respectively. The variance of the Gaussian function fulfills the role of the scale parameter s.

Edge focusing. A common way to combine the advantages of both coarse and fine scales is edge focusing (EF) [257–260]. It consists in detecting edges at some coarse scale s_{max}, at which the influence of noise is negligible, and tracking them down through the $(\mathbf{r}, s)$ space to a fine scale s_{min}, at which edges are well localized. Fig. 24 illustrates EF for a 2D example. This idea has been developed in several ways: In [261], a probabilistic model has been proposed, in which causality from coarse to fine scales is taken into account by a Markov process. In [131,262], EF is not performed pixel by pixel, but connectivity information about the object contours is taken into account. In [263–265] snakes based on EF are described. In [129,130] EF is combined with biologically motivated texture suppression schemes and in [266] EF is applied to color images. In [267,268], it is shown that EF can be used to measure global contrast and the degree of diffuseness of a contour.

EF can successfully achieve high noise rejection and good edge localization for a wide range of input images. However, it suffers several limitations:

- The postulate that highly salient contours survive to very coarse scales is not necessarily true. For instance, edges that are close to each other may quickly disappear as the scale parameter increases, even though they are highly salient in the input image (see, for instance, the bars of the golf cart in Fig. 21).
- For 2D signals, the complex topological structure of the singularities in the scale space [68] makes the method computationally unattractive [226].
- The output of EF crucially depends on the values of s_{min} and s_{max}, which are not derived by general principles. Moreover, for good results, these two parameters need to take different values for each pixel of the input images (especially s_{max}).

Adaptive scale parameter. Concerning the latter point, some effort has been made in order to develop edge detectors with a *position-dependent* scale parameter $s = s(\mathbf{r})$. The key idea is that edges are obtained by the intersections of two surfaces in the $s = s(\mathbf{r})$ space: the first one is traced by the edge curves detected at different scales by a scale dependent operator; the second one is simply the position-dependent choice of the scale $s(\mathbf{r})$. These two surfaces intersect in a 3D curve in the $(\mathbf{r}, s)$ space, whose projection on the xy plane defines the detected contours. Thus, the scale selection mechanism is reduced to the definition of $s(\mathbf{r})$.

A simple approach to define $s(\mathbf{r})$ is to compute, for each pixel, the so called *minimum reliable scale*, which is the minimum scale that guarantees a SNR above an acceptable threshold SNR_{min}. However, such a definition only shifts the problem to a different domain: since the amount of noise inside a local neighborhood $\mathcal{N}(\mathbf{r})$ of each pixel must be measured, a new scale parameter that defines the size of $\mathcal{N}(\mathbf{r})$ needs to be introduced. Moreover, the choice of SNR_{min} is heuristic. A different solution is proposed in [269]: $s(\mathbf{r})$ is determined by a generalization of non-maxima suppression to the scale space. However, such a procedure is based on ad hoc normalization of the input luminance profile. In general, contour detectors based on a position-dependent scale parameter outperform EF for images that contain edges with different levels of blurring (Fig. 25). However, the choice of $s(\mathbf{r})$ is based either on heuristics [269,270], or on some additional information about the input image, which may not always be available (such as, for instance, range data [271], or models of the sensors used for acquiring the input image [226]).

In summary, multiresolution contour detection can be classified into edge focusing and position-dependent blurring. Edge focusing can achieve both high noise rejection and good edge localization. It can be seen as a top down global method, which is more robust to noise than bottom up global methods, such as saliency computation or grouping. On the other hand, its main drawback is that the complex topological structure of the singularities of the scale space makes the method computationally unattractive. Position-dependent blurring is in general superior to edge focusing for images that contain different degrees of blurring. However, these methods can be used in combination with edge focusing, as soon as edges are tracked down until a position-dependent minimum scale.

6. Performance evaluation

A rigorous comparison of different contour detectors and a study of the influence of their input parameters require quantitative performance evaluation. Despite its importance, quantitative performance evaluation is absent in most of the published work on contour detection [272,273]. Since all the proposed approaches suffer different shortcomings, there is no general agreement about how performance evaluation should be carried out. Moreover, the quality of a contour detector may depend on the specific task for which it is deployed [274]. This section reviews the most frequently used procedures for quantitative performance evaluation.

6.1. General methodology

A first issue concerns the general methodology of performing quantitative performance evaluation and, in particular, whether ground truths (*GT*), i.e. contour specification, should be used or not. In [275,276], the quality of the detected contours is evaluated by measuring various properties of the detected contours such as coherence, continuity, smoothness and good continuation. This approach has the advantage to be objective, but consistency between the detected contours and the input image is not taken into account. In particular, edge detectors that over-blur the input image may score high with these performance indicators, though human observers would judge the related outputs as poor [273]. Conversely, agreement

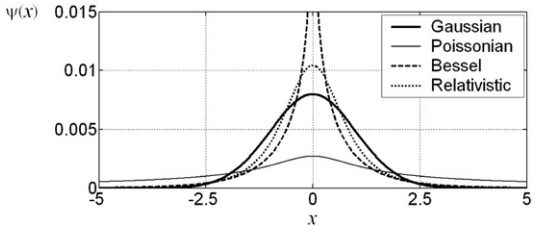

Fig. 22. Mother wavelets for the 1D Gaussian, Poissonian, Bessel and relativistic scale–spaces, for the value $s = 1$ of the scale parameter.

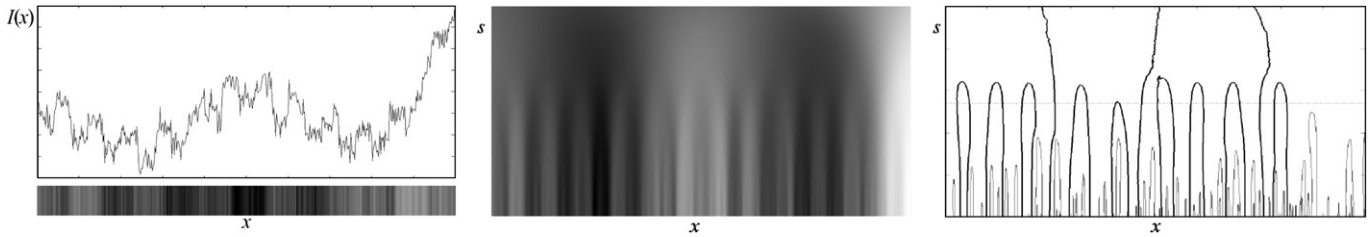

Fig. 23. Behavior of edges in the scale space for a 1D example. (left) A noisy luminance profile $I(x)$, (center) its Gaussian scale–space $I(x,s)$, shown as an intensity map in the (x,s) space, and (right) the positions (x) of zero-crossings of the Laplacian at different scales (s).

with human observers in the process of performance evaluation is achieved by performance indicators based on subjective evaluation of the quality of the detected contours [273]. This is basically done by performing psychophysical experiments in which human observer must judge the quality of the outputs of the contour detectors under examination. However, these approaches are expensive and cannot be performed automatically.

Due to these difficulties, most authors compare the set of detected contour pixels *DC* with a ground truth set of pixels *GT* that specifies the desired output for a given input image. In order to have objective *GT*s, in early work this approach was applied only to synthetic images [277,278]. Automatic estimation of such *GT*s has been proposed in [279]. However, high scores with synthetic inputs do not necessarily imply good performance with natural images [280]. Therefore, more recently *GT* based quantitative performance evaluation has been carried out for natural images as well [2,281]. For this purpose, several datasets of natural images with associated *GT* have been produced (Fig 26). Examples are the Berkeley (300 images) [2], South Florida (50 images), Sowerby (100 images) [124], Corel (5000 images) [282] and RuG (40 images) [22] datasets. In order to overcome the subjectivity of a ground truth, some datasets offer different ground truths for the same image, associated with different human observers [2]; in this way the performance of a given algorithm for a given input image can be averaged over different observers. However, since in most cases the number of observers is hardly larger than four or five, this type of quantitative performance evaluation is still quite subjective. Moreover, it is not obvious that a dataset with only a few hundreds of image is sufficiently representative.

To summarize, *GT* based performance evaluation is certainly the most popular approach, as it seems a good compromise between completely objective and completely subjective quantitative performance evaluation approaches. While for task-dependent contour detection this is probably the best choice, as reasonably objective ground truths can be generated, for general purpose applications the most reliable approach is still the direct evaluation from human observers. In this respect, despite of its aforementioned drawbacks, a comparison with ground truth if often deployed as a suboptimal solution due to its reduced cost in terms of human effort.

6.2. Performance indicators

We now move to the problem of quantifying the (dis)similarity between *DC* and *GT*. A widely used similarity measure between detected contours and a ground truth is the Pratt's figure of merit, defined as $FoM = \frac{1}{\max(card\{DC\}, card\{GT\})} \sum_{\mathbf{x}\in DC} \frac{1}{1+\left[\frac{d_{GT}(\mathbf{x})}{d_0}\right]^2}$, with d_{GT} being the distance transform of *GT* and d_0 a scale parameter. *FoM* takes values in (0,1], being equal to 1 iff *DC* coincides with *GT*, and is non-symmetric with respect to *DC* and *GT*. The scale parameter d_0 controls the sensitivity of *FoM* to differences between *GT* and *DC*: for small values of d_0, *FoM* is close to 1 only if *DC* is very similar to *GT*, while for large values of d_0 larger differences between *GT* and *DC* can be tolerated.

FoM provides an overall evaluation of the quality of a given contour map, by taking into account both the amount of false positive (i.e. presence of undesired responses in the contour map), false negative (i.e. missing contours), and shifting or deformation of a correctly detected contour from its *GT* position. In this approach,

Fig. 24. Example of EF for a 2D image. First row: input image and outputs of the Canny edge detector for $\sigma = 8,4,2,1$, respectively. Second row: edges tracked from scale $\sigma = 8$ down to the scales $\sigma = 4,2,1$, respectively.

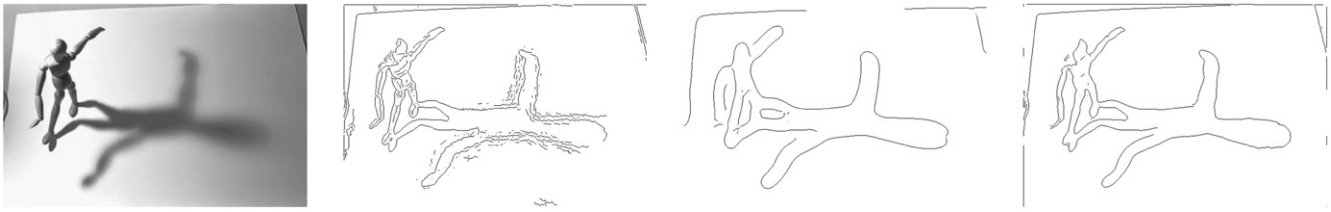

Fig. 25. Multiresolution contour detection. From left to right: input image; outputs of the Canny edge detector for $\sigma = 1$, which exhibits multiple responses on blurred edges, and for $\sigma = 4$, which smooths sharp edges; output of the multiresolution method proposed in [269], which combines the advantages of both.

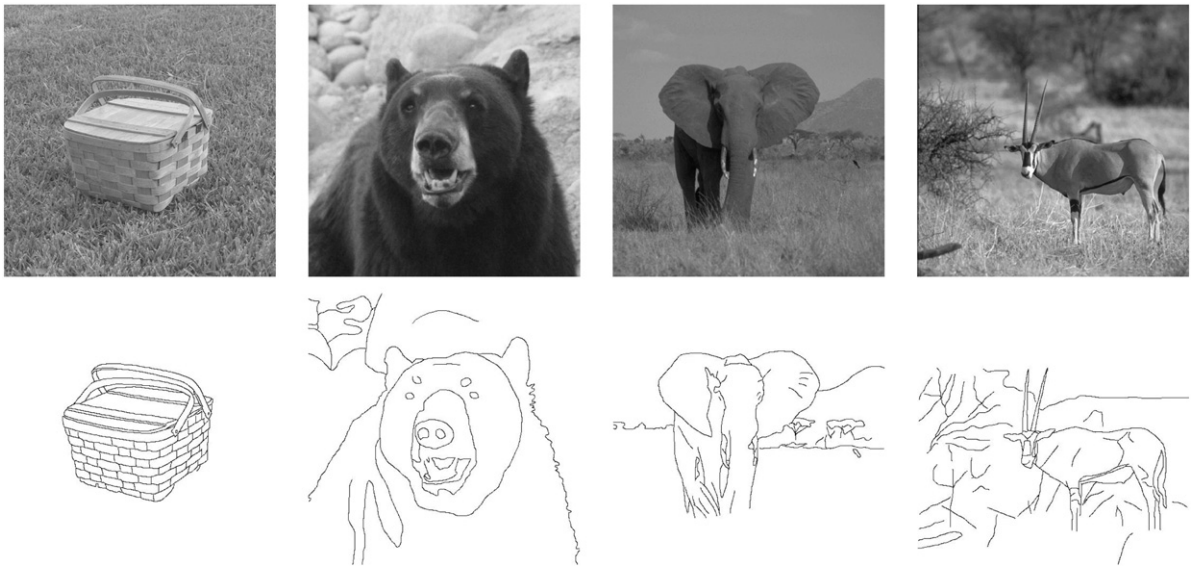

Fig. 26. First row: natural images from the RuG dataset [22]. Second row: the corresponding *GT*s.

algorithms for contour detection are compared in terms of the statistical distributions of *FoM* for test datasets of input images (Fig. 27). Other global metrics are the *F*-measure introduced in [2] and the ρ-coefficient introduced in [22], which unlike the Pratt's FoM are symmetric with respect to *DC* and *GT*, the Haursdoff distance [283], the Odet discrepancy [284], and the Chernoff Entropy [124].

The main drawback of the global figures of merit referred to above is that no information is provided about the origin of the dissimilarity: false positives, false negatives, or contour shifting. To face this problem, a well-known approach is to use *recall* $R = \frac{card\{GT \cap DC\}}{card\{DC\}}$ and *precision* $P = \frac{card\{GT \cap DC\}}{card\{DC\}}$ as performance indicators. They measure the fraction of correctly detected contour pixels ($card\{GT \cap DC\}$) with respect to the number of all contour pixels detected by a given operator ($card\{DC\}$) or the number of *GT* contour pixels ($card\{GT\}$), respectively. However, due to a possible displacement error δ of the hand-drawn contours in the ground truth with respect to their exact position in the input image, the set $GT \cap DC$ can be empty even in total absence of false positive and false negative. Therefore, slightly different quantities are computed in practice, by replacing the set $GT \cap DC$ by the set of those points of *DC* whose distance to a point from *GT* is less than a fixed threshold δ_0.

Recall and precision are usually deployed to study the influence of the input parameters on the performance of a given contour detector. Specifically, when an input parameter p is varied, the functions $R(p)$ and $P(p)$ are plotted versus each other in a ROC-like diagram. A typical situation is depicted in Fig. 28 for the Canny edge detector and a given input image, where the input parameter p is the higher threshold t_H. Since the *R*–*P* diagram can vary across different images, it is customary to plot the average functions $R_{avg}(p) \triangleq \frac{1}{N} \sum_{i=1}^{N} R_i(p)$ and $P_{avg}(p) \triangleq \frac{1}{N} \sum_{i=1}^{N} P_i(p)$ computed across several images of a dataset, where $R_i(p)$ and $P_i(p)$ are recall and precision related to the i-th image respectively. Usually, high values of precision go together with low values of recall as a consequence of many false negatives; inversely, high values of recall are achieved at the expense of many false positives and, consequently, low values of precision. *R*–*P* diagrams share some similarity to ROC diagrams, as both are used to quantify the compromise between good detection (low false negative) and good noise rejection (low false positive). However, precision and recall have the advantage to be scale-invariant, since the normalizing denominator increases linearly with the linear size of the image instead of quadratically as in the ROC [2].

A simple differential analysis of the functions $R(p)$ and $P(p)$ allows to determine an optimal value p_{opt} of the parameter p. In the following, we assume that precision and recall are equally important, i.e., a small loss in precision must be balanced by an equal gain in recall, and vice-versa.[5] It is easy to see that this is reached at the point on the *R*–*P* diagram whose tangent has an angular coefficient equal to -1.

The performance indicators discussed so far depend on a scale parameter – d_0 for *FoM* and δ_0 for *P* and *R* – that is here generically denoted by σ. While such a dependence is often neglected in the literature, it strongly influences the results of quantitative performance evaluation, thus making impossible to compare the values of the performance indicator obtained by different authors. We have studied such a dependence and the result is that these performance indicators grow proportionally to $log\sigma$. Specifically, a set $\mathcal{S}$ of $N = 160$ algorithmic results has been generated by applying four different

[5] When this is not the case, it is sufficient to replace *P* and *R* with new quantities $P' = \lambda P$ and $R' = \mu R$, where the coefficients λ and μ are chosen so that, for that specific task, P' and R' are equally important.

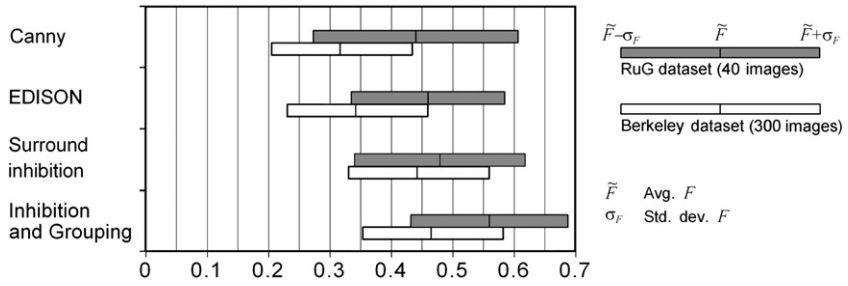

Fig. 27. Statistical distribution of the values of *FoM* computed over the RuG [22] and the Berkeley [2] datasets, for four different contour detectors: the Canny edge detector [73], the local method EDISON which combines different local features [123], a contextual method which deploys surround inhibition only [128], and a more sophisticated global scheme which exploits both inhibition and Gestalt edge grouping [184]. The increase of sophistication corresponds to a performance improvement in the results.

contour detectors [73,123,128,184] for each one of the 40 images of the RuG dataset [128]. For each algorithmic result, the similarity measures *FoM* and *F* discussed above have been computed for different values of $\sigma = 1,\ldots,10$, and let $f(\sigma)$ values corresponding to the i-th algorithmic result (for conciseness, here the same symbol f is used to indicate both the values of the metrics *FoM* and *F*). We have then computed the normalized differences $r_i(\sigma) = \frac{f_i(\sigma)-f_i(1)}{f_i(1)}$ and averaged them across all N algorithmic results, thus obtaining the function $r_{avg}(\sigma) \triangleq \frac{1}{N}\sum_{i=1}^{N} r_i(\sigma)$. Such a function is plotted in Fig. 29 for both *FoM* and *F*. As we see, they are well approximated by a logarithmic function, especially for what concerns *FoM*. This result allows to relate different measurements performed with different values of the scale parameter (Fig. 29).

In conclusion, the methods proposed in the literature to compare an algorithmic result with a ground truth can be divided in schemes based on (i) a single performance measure, and (ii) and multiple performance indicators such as precision and recall. The former are usually deployed to compare different algorithms in terms of the statistical distributions, evaluated across different images, of the figure of merit, which has been optimized with respect to the input parameters; it should be noted that the Pratt's FoM and the Haursdoff distance take into account the shifting and deformation error between the detected contour and the ground truths, thus penalizing algorithms which oversmooth contours. As to multiple indicators, they are usually plotted against each other for different values of some input parameter to illustrate best compromise between different goals. Besides these considerations, there is neither theoretical basis nor empirical evidence to prefer a specific performance indicator.

7. Computational complexity

In this section, we discuss some aspects related to the computational complexity of the contour detection algorithms discussed above. Let us first consider the local edge detectors reviewed in Section 3. The majority of them are based on linear filtering, such as differential methods (Section 3.1), some of the statistical approaches described in Section 3.2, and the computation of local energy and phase congruency (Section 3.3); therefore the computational complexity is related to the number of required convolutions. In this respect, differential methods and algorithms based on the monogenic signal require at maximum three convolutions, thus faster than schemes based on the computation of local energy by means of Gabor filters, for which the number of required convolutions is proportional to the number of considered orientations (usually at least six).

7.1. Complexity of linear local methods

In general, the computation time of a convolution depends on its specific implementation: working directly in the space domain would require $O(N\sigma^2)$ for non-separable kernels and $O(N\sigma)$ for separable ones (i.e. kernels $w(x,y)$ which can be expressed as the product of a function of the sole x with one of the sole y, $w(x,y) = w_1(x)w_2(y)$), where N is the number of pixels in the input image and σ is the scale parameter.

When scale invariance is required, thus making $\sigma \propto \sqrt{N}$, the computation time becomes $O(N^2)$ for the non-separable case and $O(N^{3/2})$ for the separable one. On the other hand, the fast Fourier transform allows to compute convolution in $O(N\log N)$ independently on σ, which is slightly less than $O(N^{3/2})$. It is widely claimed that working in the Fourier domain is faster than in the space domain. However, while it is certainly true in the limit $N \leftarrow \infty$, this is not necessarily true when the smoothing kernel is separable. Specifically, for finite N, $O(N\log N)$ can be very similar to $O(N^{3/2})$, and the most important role is played by the proportionality factor which relates the computation time to the asymptotical complexity. Such a factor is constant for the Fourier domain, while, for the spatial domain, it is proportional to the ratio $r = \sigma / \sqrt{N}$ between the scale parameter and the linear size of the input image. Therefore, at least for small values of r, the spatial domain implementation is faster than the frequency domain one.[6] A significant example, in which the spatial domain offers more efficient implementations than the Fourier one even for large values of r, is given by the computation of the Gaussian scale space of an image. Specifically, due to the semigroup property of the scale space, the convolution between the input image and a Gaussian kernel with a large σ can be reduced to the cascade of several convolutions with a small σ. Since the intermediate scales are also needed to compute the scale space, the overall computational complexity is lower in the spatial domain than in the frequency domain. It must also be pointed out that for some filters, such as the optimal filter in the sense of the Canny criteria, fast IIR implementations are available whose computational complexity is $O(N)$ independently on σ.

7.2. Complexity of nonlinear local methods

The computational complexity of the nonlinear edge detectors reviewed in Sections 3.2 and 3.4 is higher than the aforementioned linear methods. Statistical approaches (Section 3.2) require, for each pixel and for each orientation, the computation of a statistical two-samples test on a population of roughly σ^2 pixels, thus making the overall computational complexity $O(N\sigma^2 N_\phi)$, where N_ϕ is the number of orientations. As to algorithms based on VOS (Section 3.4), it is generally claimed that they are very expensive from a computational point of view. However, this is true only if a straightforward implementation is deployed. Specifically, ordering pixels in a neighborhood of linear size σ requires $O(\sigma^2 \log \sigma)$ operations, which would make the overall computation time as high as $O(N\,\sigma^2 \log \sigma)$. However, the neighborhoods $\mathcal{N}(i,k)$ and $\mathcal{N}(i,k+1)$ of two adjacent pixels (i,k) and $(i,k+1)$ strongly overlap and only differ of two one-

[6] For instance, for a 512×512 image, working in the spatial domain is faster than in the frequency domain as long $\sigma \geq 6$.

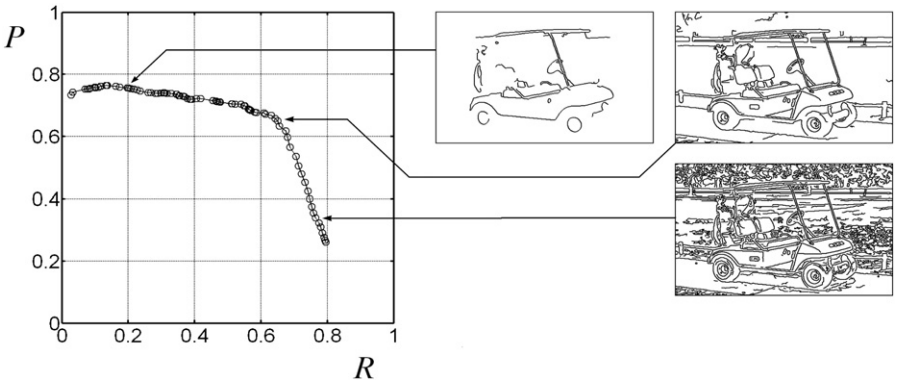

Fig. 28. *R–P* plot related to the Canny edge detector for different values of the higher threshold t_H.

pixel-wide strips of pixels with linear size $O(\sigma)$; therefore, ordering $\mathcal{N}(i,k+1)$, when $\mathcal{N}(i,k)$ has already been ordered, reduces to replacing $O(\sigma)$ pixels from an ordered set of size $O(\sigma^2)$. With binary trees, this can be done in $O(\sigma log]\sigma)$ operations, thus reducing the overall computation time reduces to $O(N\sigma log\sigma)$, which is only slightly higher with respect to a convolution in the spatial domain with a separable kernel.

7.3. Complexity of non-local methods

We finally move to the contextual and global methods discussed in Section 4. Since most of them are iterative, their computational complexity-higher with respect to local methods, and it is mainly determined by the number N_{iter} of iterations necessary to reach convergence. For algorithms based on diffusion processes, such as active contours or tensor voting, N_{iter} strongly depends on the stopping rule and its parameters and changes sensitively for different images. In general, for these methods is difficult even to make upper bounds to N_{iter}. On the other hand, for graph-based methods, such as the Gestalt grouping approaches described in Section 4.2 we can state that N_{iter} grows polynomially with the size of the input image. Examples are the methods for detecting closed structures proposed in [164,171,174], whose computational complexities are, respectively, $O(n^2 logn)$, $O(n^2)$ and $O(n^{7/4})$.

8. Summary, discussion, and conclusions

In Table 1, we give a taxonomy of edge and line oriented contour detection algorithms. Our main distinction is between local and global methods. The former, mainly based on differential, statistical and local Fourier analysis, aim at detecting local transitions of luminance, color and texture in an image. Most of them are fast, conceptually simple and intuitive, and easy to implement. However, they fail in capturing mid-level and high-level visual cues, such as good continuation, contour closure, shape, and symmetry. These drawbacks are overcome by contextual and global techniques, which go far beyond the simple analysis of a local neighborhood of each pixel. We classify them into algorithms that compute contour saliency, based on surround suppression and facilitation, algorithms which group edge pixels into long chains of collinear responses according to the Gestalt laws of good continuation, closure and symmetry, and active contours. Global methods give much higher performance than local methods, but they are also computationally more demanding. Most of the local and global approaches can be implemented in both a single resolution and a multiresolution fashion. We classify the strategies to combine the outputs at different scales into (i) edge focusing and (ii) the use of local scale parameter. The former gives an excellent tradeoff between good localization and good noise rejection, while the latter is more indicated to detect contours with different blurring. Probably, the best way to achieve the advantages of both techniques is to track down edges and contours through the scale–space until a position-dependent scale $\sigma(\mathbf{r})$ instead of a fixed one.

8.1. Dependence on the input parameters

A critical aspect of all aforementioned approaches for contour detection is that several input parameters need to be specified by the user. Their choice profoundly influences the results of contour detection and there are no universal values which lead to good results for every input images. Therefore, much research has been carried out to reduce the influence of the input parameters. The proposed strategies can be classified into two groups:

- Multivalued schemes, in which the concerned algorithm is applied several times for N different values $p_1,\ldots,p_N$ in a range $[p_{min},p_{max}]$ of the concerned parameter p, and supplementary techniques are used to combine the results. Examples are edge focusing [257], hysteresis thresholding [73], or the multithreshold approach proposed in [184]. These approaches actually increase the number of input parameters to be specified, since the user must provide the values of p_{min}, p_{max} and N instead of the sole value p. However, multivalued algorithms are generally less sensitive to the values of p_{min}, p_{max} and N with respect to the single-valued case and it easier to find a range $[p_{min},p_{max}]$ which produces good results for a large number of input images. Moreover, better performance is achieved since the advantage of using both large and small values of p are exploited.
- Schemes which include further analysis of the input image aimed to compute the optimal value of the concerned parameter p for that specific image. In some cases, the value of p is computed locally for each pixel of the input image, rather than globally for the whole image (see, e.g., [269,288]). While these approaches can bring some improvement in the results, their main problem, from a theoretical point of view, is that the sub-algorithms deployed to optimize a certain parameter p need the introduction from the user to other input parameters. Thus, they just move the problem of parameter selection from a domain to another one.

We comment on two parameters: (i) the size s of the window that is deployed to extract local edge features (scale parameter), and (ii) the value of the threshold which the edge strength is compared to.

The value of the scale parameter is determined, for single resolution methods, by a tradeoff between noise rejection, achieved with large values of s, and good localization, for which small values of s give better result. Some of the multiresolution techniques reviewed in Section 5 attempt to automatically determine the scale parameter from the input image. For instance, in [269], a local scale parameter is obtained from a generalization of the non-maxima suppression performed in the scale space. However, a new parameter γ, which

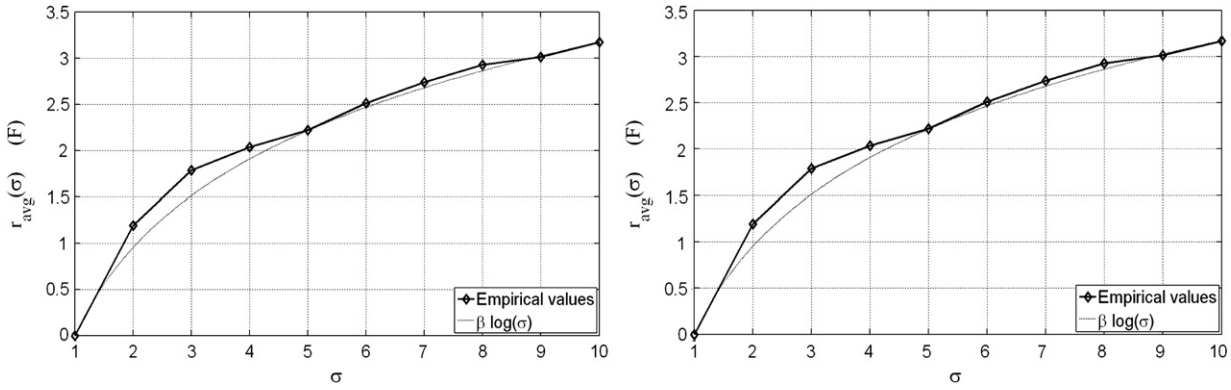

Fig. 29. Plots of the function $r_{avg}(\sigma)$ for the *FoM* and *F* metrics, which fit very well with a logarithmic model.

relates the edge strength at different scales, needs to be introduced, whose choice is not derived from general principles. Other multiscale approaches, based on edge focusing, are characterized by two scale parameters, s_{min} and s_{max}, which are the minimum and the maximum values of the size of the smoothing window. While their choice is mainly heuristic, in general edge focusing schemes are less affected by the values of s_{min} and s_{max} with respect to single resolution methods.

The second input parameter we are going to discuss is the value T of the threshold. While it is relatively easy to tune its value to a single image, it is virtually impossible to find a universal value of T which produces acceptable results for every input image. Therefore, much research has been made to develop algorithms which automatically determine the threshold from the input image. The simplest approach is to analyze the histogram H of the edge feature f that must be thresholded and look for bimodality, thus placing the threshold T at the local minima between the two modes. However, in most cases the concerned histogram is far from multimodal. Other authors have proposed to determine the optimal threshold in a Bayesian framework. Specifically, the set of all pixels of the input images is divided into the classes A and B of edge and non-edge responses. Then, let $p_A(f)$ and $p_B(f)$ be the conditioned probability distributions of f for classes A and B respectively, and P_A and P_B the related prior probabilities, the optimal threshold is given by the value f_{opt} for which the equality $P_A p_A(f_{opt}) = P_B p_B(f_{opt})$ holds. In this process, the most difficult task determining of $p_A(f)$ and $p_B(f)$. Some authors estimate $p_A(f)$ and $p_B(f)$ from the histogram H of f by means of some analytic assumptions on $p_A(f)$ and $p_B(f)$ (the most used analytical forms for $p_1(f)$ and $p_2(f)$ are the Rayleigh, the χ^2, and the Gamma distribution [289]). However, $p_A(f)$ and $p_B(f)$ can hardly be modeled with simple analytical forms, thus weakening the optimality of the resulting value of the threshold. In other work, $p_A(f)$ and $p_B(f)$ are estimated from hand-drawn ground truths [2,124], thus allowing for non-parametric models. While this brings some improvement in the performance, there are two important drawbacks: (i) the resulting threshold is somewhat subjective, as different observers will produce different ground truths, and (ii) it is assumed that the image dataset for which ground truth are available are sufficiently representative, which is not necessarily the case. Due to these limitations, other authors refrain from perform a statistical analysis of f and, rather, determine f_{opt} by optimizing some morphological properties of the resulting contour map, such as the number or the average length of the connected components in the resulting binary map [290].

In general, it can be concluded that the influence of the input parameters is still an open problem in contour detection and that most of the recent sophistication bring little improvement with respect to standard techniques, when they are constrained to use the same values of the input parameters for a large range of input images.

8.2. Interdipendence of computational steps

Another important limitation of most of the existing algorithms for contour defection, which leaves space for further research, is that the steps which they consist of are optimized separately, rather than considering the entire process as a whole (though there are some exceptions, such as some methods based on Markov random fields). An important example is given by the algorithms to extract local edge features, which are not designed in terms of subsequent steps such as computing a contour saliency from local responses, or grouping edges according to Gestalt principles. In particular, much research has been made to distinguish from long chains of collinear edges and randomly scattered oriented stimuli [136,137,291]. Biological evidence suggests that the human visual system deploys similar mechanisms [139,140]. However, the local edge direction extracted with most of the existing local techniques, shows high correlation even when the input image is just random noise.[7] This implies that edges originated by both object contours and noise or texture tend to be in good continuation with each other, thus weakening the effectiveness of the aforementioned post-processing technique. As a conclusion, we think that a more elaborate optimization of contour detection schemes, in which the interdependence of the different sub-algorithms is taken into account, will result in a considerably higher effectiveness and efficiency.

8.3. Importance of shape information

A major progress achieved recently is the introduction of prior shape information about the contours to be detected, so that only contours of certain given shapes will be detected. The main advantages of these approaches are their ability to detect objects in the presence of strong occlusion and, at the same time, to reject undesired contours, such as those ones due to sharp shadows or to uninteresting objects. Local and even global methods which do not exploit shape information are certainly inadequate for this task. Moreover, shape analysis allows to reach stronger immunity to image corruption, such as noise and blurring. Shape based contour detection is usually formulated as an optimization problem, where the cost function to be minimized includes a term that takes into account the dissimilarity between the shape of the detected contour and the desired shape. Such a term is expressed as a likelihood ratio, which is learned from shape datasets by means of density estimation techniques.

[7] More precisely, let $\phi(\mathbf{r})$ be the orthogonal direction to the Gaussian gradient of a white random noise $n(\mathbf{r})$. Then, the probability distribution $p(\phi(\mathbf{r}+\mathbf{h}(\mathbf{r}))-\phi(\mathbf{r}))$, with $\mathbf{h}(\mathbf{r})$ oriented along the local edge direction, shows a pronounced peak, even for $|\mathbf{h}| > 2\sigma$.

Table 1
Proposed taxonomy of the main contour detectors presented in the literature. For each class, the main features are indicated in parenthesis and the most representative articles are mentioned.

Local methods	Differential (simple, fast)	Basic concepts [62] Zero-crossings, Marr–Hildreth, [64,65,68] Optimal edge detection, Canny, [73–76] Local fitting [81–83,85]
	Statistical (texture boundaries)	Distribution of gray-levels [87–89] Distribution of color and texture attributes [2,90] Distribution of the gradient vector field [95,97]
	Phase congruency (idempotence)	Quadrature filters, Gabor, [103,104,107,108] Monogenic signal [109] Applications [104,110,113,114]
	VOS and morphological (color, salt and pepper noise)	[118–120]
	Features combination (optimization)	[2,123–125]
Contextual and global methods	Saliency (low-contrast contours)	Surr. suppr. and facilitation [22,129,137,285] Tensor voting [133,135] Relaxation labelling [151,156,286]
	Gestalt grouping (from edge pixels to contours)	Good continuation [10,159,160,287]. Closure [164,171,174] Symmetry [158,180,181] Directional dilation [183,184]
	Active contours (user interaction)	Key concepts [12,188,189,193] Gradient vector flow [206,207] Region-based energy [210,211,213] Shape priors [216,219,221]
Multiresolution methods	Edge behavior in the sale space	[68,255,256]
	Edge focusing (good localization and noise rejection)	[129,257,260,261]
	Local scale control (different types of blurring)	[269,271]

Despite of the conceptual simplicity of this approach, several technical aspects need to be taken into account: first, the extremely high dimensionality of the shape space on which the cost function is defined makes the optimization difficult to carry out. The easiest way to face this problem is by PCA or ICA, but the assumption of linearity is often too simplistic (see also the discussion in [217]). Also, the assumption that the entire shape should follow the prior model is often too restricted and there are important situations in which this is not the case (such as object occlusion), moreover, if the similarity between the object of interest in the input image and the prior shape model occur only in some part of the image, these approaches will not converge to an optimum. An effective way to solve this problem is to associate a local confidence map to the prior model, and to constrain the detected contours only on high confidence regions [292]. In general, the introduction of shape analysis in contour detection brings a substantial improvement in the performance, especially for algorithms devoted to solve a specific problem. However, its success is nowadays limited to situations in which the object of interest is sufficiently similar in shape to the observed training examples used to build the prior model.

Our main conclusion is that contour detection has reached high degree of maturity and sophistication, taking into account multimodal contour definition (by luminance, color or texture changes), mechanisms for reducing the contour masking influence of noise and texture, perceptual grouping, multiscale aspects and, in some cases, high-level vision information such as shape (Fig. 30). As a result, considerable progress has been achieved through the years regarding the robustness of algorithms to the choice of parameter values. We think that research in and development of feedback schemes from higher vision operations has high potential for achieving further substantial progress.

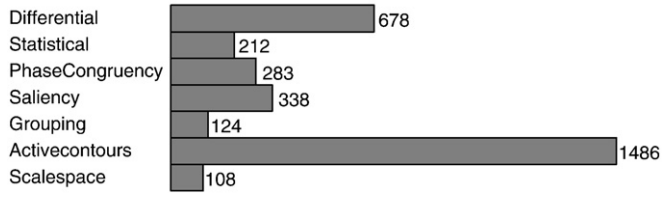

Fig. 30. Total number of citations in the years 2004–2006 of the 5 most cited articles on contour detection from each sub-area according to the ISI Web of Science.

References

[1] S. Zeki, A Vision of the Brain, Blackwell Scientific Publications, Oxford, 1993.
[2] D.R. Martin, C. Fowlkes, J. Malik, Learning to detect natural image boundaries using local brightness, color, and texture cues, IEEE T-PAMI 26 (5) (2004) 530–549.
[3] N.R. Pal, S.K. Pal, A review on image segmentation techniques, PR 26 (9) (1993) 1277–1294.
[4] H.D. Cheng, X.H. Jiang, Y. Sun, J. Wang, Color image segmentation: advances and prospects, PR 34 (12) (2001) 2259–2281.
[5] X. Cufi, X. Munoz, J. Freixenet, J. Marti, A review of image segmentation techniques integrating region and boundary information, Adv. Imaging Electron. Phys. 120 (2002) 1–39.
[6] D. Ziou, S. Tabbone, Edge detection techniques—an overview, Int. J. Pattern Recognit. Image Anal. 8 (4) (1998) 537–559.
[7] M. Petrou, The differentiating filter approach to edge detection, Adv. electron. electron phys. 88 (1994) 297–345.
[8] P.T. Jackway, Morphological scale–spaces, Adv. Imag. Elec. Phys. 119 (2001) 123–189.
[9] O. Scherzer, Scale–space methods and regularization for denoising and inverse problems, Adv. Imag. Elec. Imag. 128 (2003) 445–530.
[10] S. Sarkar, K.L. Boyer, Perceptual organization in computer vision: a review and a proposal for a classificatory structure, IEEE SMC 23 (2) (1993) 382–399.
[11] K.L. Boyer, S. Sarkar, Guest editors' introduction. Perceptual organization in computer vision: status, challenges, and potential, CVIU 76 (1) (1999) 1–6.
[12] T. McInerney, D. Terzopoulos, Deformable models in medical image analysis: a survey, Med. Image Anal. 1 (2) (1996) 91–108.
[13] S. Sarkar, P. Soundararajan, Supervised learning of large perceptual organization: graph spectral partitioning and learning automata, IEEE Trans. Pattern Anal. Mach. Intell. 22 (5) (2000) 504–525.
[14] S. Wang, J.M. Siskind, Image segmentation with ratio cut, IEEE Trans. Pattern Anal. Mach. Intell. 25 (6) (2003) 675–690.
[15] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE T-PAMI 24 (5) (2002) 603–619.
[16] S.C. Zhu, A. Yuille, Region competition: unifying snakes, region growing, and Bayes/mdl for multiband image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 18 (9) (1996) 884–900.

[17] A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review, ACM comput. sur. (CSUR) 31 (3) (1999) 264–323.
[18] R. Adams, L. Bischof, Seeded region growing, IEEE Trans. Pattern Anal. Mach. Intell. 16 (6) (1994) 641–647.
[19] K. Haris, S.N. Efstratiadis, N. Maglaveras, A.K. Katsaggelos, Hybrid image segmentation using watersheds and fast region merging, IEEE Trans. Image Process. 7 (12) (1998) 1684–1699.
[20] A.K. Jain, F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, Pattern Recognit. 24 (12) (1991) 1167–1186.
[21] J. Liu, Y.H. Yang, Multiresolution color image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 16 (7) (1994) 689–700.
[22] C. Grigorescu, N. Petkov, M.A. Westenberg, Contour detection based on nonclassical receptive field inhibition, IEEE TIP 12 (7) (2003) 729–739.
[23] G. Papari, N. Petkov, P. Campisi, Artistic edge and corner enhancing smoothing, IEEE TIP 29 (10) (2007) 2449–2462.
[24] P. Saint-Marc, J.S. Chen, G. Medioni, Adaptive smoothing: a general tool for early vision, IEEE T-PAMI 13 (6) (1991) 514–529.
[25] C. Tomasi, R. Manduchi, Bilateral filtering for gray and color images, Proceedings of the Sixth International Conference on Computer Vision, Narosa Publishing House, Bombay, 1998, p. 839.
[26] Mark A. Schulze, John A. Pearce, A morphology-based filter structure for edge-enhancing smoothing, ICIP, vol. 2, 1994, pp. 530–534.
[27] R. van den Boomgaard, Decomposition of the Kuwahara–Nagao operator in terms of linear smoothing and morphological sharpening, Proc. of the 6th International Symposium on Mathematical Morphology, 2002, pp. 283–292.
[28] I. Pitas, A.N. Venetsanopoulos, Order statistics in digital image processing, Proc. IEEE 80 (12) (December 1992) 1893–1921.
[29] D. Barash, D. Comaniciu, A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift, IVC 22 (1) (2004) 73–81.
[30] P. Bakker, L.J. van Vliet, P.W. Verbeek, Edge preserving orientation adaptive filtering, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos, 1999, pp. 535–540, (IEEE).
[31] R.C. Hardie, C. Boncelet, Lum filters: a class of rank-order-based filters for smoothing and sharpening, IEEE TSP 41 (3) (1993) 1061–1076.
[32] V.I. Ponomaryov, F.J. Galleros-Funes, A. Rosales-Silva, Real-time color imaging based on rm-filters for impulsive noise reduction, J. Imag. Sci. Tech. 49 (3) (2005) 205–219.
[33] V. Barnett, The ordering of multivariate data, J. Royal Stat. Soc. A 139 (3) (1976) 318–355.
[34] K.N. Plataniotis, A.N. Venetsanopoulos, Color Image Processing and Applications, Springer, 2000.
[35] J.T. Astola, P. Haavisto, Y. Neuvo, Vector median filters, Proc. IEEE 78 (1990) 678–689.
[36] A. Brook, R. Kimmel, N.A. Sochen, Variational restoration and edge detection for color images, J. Math. Imag. Vis. 18 (3) (2003) 247–268.
[37] J.F. Aujol, A. Chambolle, Dual norms and image decomposition models, IJCV 63 (1) (2005) 85–104.
[38] D. Geman, G. Reynolds, Constrained restoration and the recovery of discontinuities, IEEE T-PAMI 14 (3) (1992) 367–383.
[39] S. Teboul, L. Blanc-Feraud, G. Aubert, M. Barlaud, Variational approach for edge-preserving regularization using coupled pdes, IEEE TIP 7 (3) (1998) 387–397.
[40] P.C. Hansen, Analysis of discrete ill-posed problems by means of the l-curve, SIAM Rev. 34 (4) (1992) 561–580.
[41] C.R. Vogel, M.E. Oman, Fast, robust total variation-based reconstruction of noisy, blurred images, IEEE TIP 7 (6) (1998) 813–824.
[42] L.I. Rudin, S.J. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D 60 (1992) 259–268.
[43] A. Chambolle, P.L. Lions, Image recovery via total variation minimization and related problems, Num. Math. 76 (2) (1997) 167–188.
[44] D. Mumford, J. Shah, Optimal approximations by piecewise smooth functions and variational problems, Comm. Pure Appl. Math. XLII (5) (1988) 577–685.
[45] P. Blomgren, T.F. Chan, Color tv: total variation methods for restoration of vector-valued images, IEEE TIP 7 (3) (1998) 304–309.
[46] T.F. Chan, S. Osher, J. Shen, The digital tv filter and nonlinear denoising, IEEE TIP 10 (2) (2001) 231–241.
[47] A. Marquina, S. Osher, Explicit algorithms for a new time dependent model based on level set motion for nonlinear deblurring and noise removal, SIAM J. Sci. Comp. 22 (2) (March 2000) 387–405.
[48] S. Farsiu, M.D. Robinson, M. Elad, P. Milanfar, Fast and robust multiframe super resolution, IEEE TIP 13 (10) (2004) 1327–1344.
[49] David C. Dobson, R.V. Curtis, Convergence of an iterative method for total variation denoising, SIAM J. Num. Anal. 34 (5) (1997) 1779–1791 (October).
[50] V. Caselles, J.M. Morel, G. Sapiro, A. Tannenbaum, Introduction to the special issue on partial differential equations and geometry-driven diffusion in image processing and analysis, IEEE TIP 7 (3) (1998) 269–273.
[51] M.J. Black, G. Sapiro, D.H. Marimont, D. Heeger, Robust anisotropic diffusion, IEEE TIP 7 (3) (1998) 421–432.
[52] J. Weickert, Theoretical foundations of anisotropic diffusion in image processing, IVC 11 (1996) 221–236.
[53] P. Perona, J. Malik, Scale–space and edge detection using anisotropic diffusion, IEEE T-PAMI, PAMI 12 (7) (1990) 629–639.
[54] F. Catté, P.L. Lions, J.M. Morel, T. Coll, Image selective smoothing and edge detection by nonlinear diffusion, SIAM J. Num. Anal. 29 (1) (February 1992) 182–193.
[55] F. Torkamani-Azar, K.E. Tait, Image recovery using the anisotropic diffusion equation, IEEE TIP 5 (11) (1996) 1573–1578.
[56] M. Nitzberg, T. Shiota, Nonlinear image filtering with edge and corner enhancement, IEEE T-PAMI 14 (8) (1992) 826–833.
[57] G. Sapiro, A. Tannenbaum, Affine invariant scale–space, IJCV 11 (1) (1993) 25–44.
[58] L. Alvarez, L. Mazorra, Signal and image restoration using shock filters and anisotropic diffusion, SIAM J Num. Anal. 31 (2) (April 1994) 590–605.
[59] R.T. Whitaker, Geometry-limited diffusion in the characterization of geometric patches in images, CVGIP 57 (1) (January 1993) 111–120.
[60] G. Sapiro, D.L. Ringach, Anisotropic diffusion of multivalued images with applications to color filtering, IEEE TIP 5 (11) (1996) 1582–1586.
[61] F.L. Fontaine, S. Basu, Wavelet-based solution to anisotropic diffusion equation for edge detection, Int. J. Imaging Sys. Tech. 9 (5) (1999) 356–368.
[62] B.M. ter Haar Romeny, Front-end Vision and Multi-scale Image Analysis, Kluwer Academic Publishers, 2002.
[63] D. Marr, E. Hildreth, Theory of edge detection, Proc. Royal Soc. London B 207 (1167) (1980) 187–217.
[64] X. Wang, Laplacian operator-based edge detectors, PAMI, IEEE Trans. 29 (5) (2007) 886–890.
[65] J. Vriendt, Accuracy of the zero crossings of the second directional derivative as an edge detector, Multidim. Sys. Sign. Proc. 4 (3) (1993) 227–251.
[66] L.M. Kennedy, M. Basu, Image enhancement using a human visual system model, PR 30 (12) (1997) 2001–2014.
[67] L.M. Kennedy, M. Basu, A Gaussian derivative operator for authentic edge detection and accurate edge localization, IJPRAI 13 (3) (1999) 367–380.
[68] J.J. Clark, Singularity theory and phantom edges in scale space, IEEE T-PAMI 10 (5) (September 1988) 720–727.
[69] M. Bertero, T.A. Poggio, V. Torre, Ill-posed problems in early vision, Proc. IEEE 76 (8) (1988) 869–889.
[70] K. Anil, Jain, Fundamentals of Digital Image Processing. Prentice-Hall, Englewood Cliffs, 1989.
[71] L. Ganesan, P. Bhattacharyya, Edge detection in untextured and textured images —a common computational framework, IEEE SMC 27 (5) (1997) 823–834.
[72] J.F. Canny, A variational approach to edge detection, in: M.A. Genesereth (Ed.), Proceedings of the National Conference on Artificial Intelligence, AAAI Press, Washington, D.C, 1983, pp. 54–58, (August).
[73] J.F. Canny, A computational approach to edge detection, IEEE T-PAMI 8 (6) (1986) 679–698.
[74] M. Petrou, J. Kittler, Optimal edge detectors for ramp edges, IEEE T-PAMI 13 (5) (1991) 483–491.
[75] S. Sarkar, K.L. Boyer, On optimal infinite impulse response edge detection filters, IEEE T-PAMI 13 (11) (1991) 1154–1171.
[76] D.R. Demigny, T. Kamlé, A discrete expression of Canny's criteria for step edge detector performances evaluation, IEEE T-PAMI 19 (11) (1997) 1199–1211.
[77] M. Unser, A. Aldroubi, M. Eden, B-spline signal processing: part I — theory, IEEE TSP 41 (2) (1993) 821–833.
[78] M. Unser, A. Aldroubi, M. Eden, B-spline signal processing: part II — efficient design and applications, IEEE TSP 41 (2) (1993) 834–848.
[79] G. Chen, Y.H.H. Yang, Edge-detection by regularized cubic B-spline fitting, IEEE T-SMC 25 (4) (1995) 636–643.
[80] Y.P. Wang, Image representations using multiscale differential operators, IEEE TIP 8 (12) (1999) 1757–1771.
[81] B. LI, J. Shen, Two-dimensional local moment, surface fitting and their fast computation, PR 27 (6) (1994) 785–790.
[82] S. Ghosal, R. Mehrotra, Detection of composite edges, IEEE TIP 3 (1) (1994) 14–25.
[83] K.H. Liang, T. Tjahjadi, Y.H. Yang, Roof edge detection using regularized cubic b-spline fitting, PR 30 (5) (1997) 719–728.
[84] K.H. Liang, T. Tjahjadi, Y.H. Yang, Bounded diffusion for multiscale edge detection using regularized cubic b-spline fitting, IEEE SMC 29 (2) (1999) 291–297.
[85] Y.P. Wang, S.L. Lee, Scale–space derived from B-splines, IEEE T-PAMI 20 (10) (1998) 1040–1055.
[86] P. Qiu, Jump surface estimation, edge detection, and image restoration, J. Am. Stat. Ass. 102 (478) (2007) 745–756.
[87] J.S. Huang, D.H. Tseng, Statistical theory of edge detection, CGIP 43 (1988) 337–346.
[88] E. Chuang, D. Sher, χ^2 test for feature detection, PR 26 (11) (1993) 1671–1681.
[89] D.H. Lim, S.J. Jang, Comparison of two-sample tests for edge detection in noisy images, Stat 51 (1) (2002) 21–30.
[90] M.A. Ruzon, C. Tomasi, Edge, junction, and corner detection using color distributions, IEEE T-PAMI 23 (11) (2001) 1281–1295.
[91] Y. Rubner, C. Tomasi, L.J. Guibas, The earth mover's distance as a metric for image retrieval, IJCV 40 (2) (2000) 99–121.
[92] J. Malik, S. Belongie, T.K. Leung, J. Shi, Contour and texture analysis for image segmentation, IJCV 43 (1) (2001) 7–27.
[93] N. Chaji, H. Ghassemian, Texture-gradient-based contour detection, JASP (2006) (2006).
[94] Mark Nitzberg, David Mumford, Takahiro Shiota, *Filtering, Segmentation and Depth*, volume 662 of *Lecture Notes in Computer Science*, Springer, 1993.
[95] S. Ando, Image field categorization and edge/corner detection from gradient covariance, IEEE T-PAMI 22 (2) (2000) 179–190.
[96] P.H. Gregson, Using angular dispersion of gradient direction for detecting edge ribbons, IEEE T-PAMI 15 (7) (1993) 682–696.
[97] J.B. Martens, Local orientation analysis in images by means of the hermite transform, IEEE TIP 6 (8) (1997) 1103–1116.
[98] D.J. Park, Kwon M. Nam, R.H. Park, Edge detection in noisy images based on the co-occurrence matrix, PR 27 (6) (1994) 765–775.
[99] S.M. Smith, J.M. Brady, Susan: a new approach to low-level image-processing, IJCV 23 (1995) 45–78.
[100] A.V. Oppenheim, J.S. Lim, The importance of phase in signals, Proc. IEEE 69 (5) (1981) 529–541.
[101] M.C. Morrone, D. Burr, J. Ross, R. Owens, Mach bands are phase dependent, Nat. (1986) 250–253.

[102] M. Morrone, D. Burr, Feature detection in human vision: a phase dependent energy model, Proc. Royal Soc. London Bull. (1988) 221–245.
[103] M. Morrone, R. Owens, Feature detection from local energy, PRL 6 (1987) 303–313.
[104] P. Kovesi, Image features from phase congruency, Videre: J. Comp. Vis. Res. 1 (3) (1999) (26 pages).
[105] D. Boukerroui, J.A. Noble, M. Brady, On the choice of band-pass quadrature filters, J. Math. Imag. Vis. 21 (1) (2004) 53–80.
[106] J.L. Marroquin, J.E. Figueroa, M. Servin, Robust quadrature filters, J. Opt. Soc. Am. A 14 (4) (1997) 779–791.
[107] J.G. Daugman, Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters, J. Opt. Soc. Am. A 2 (1985) 1160–1169.
[108] J.K. Kamarainen, V. Kyrki, H. Kalviainen, Invariance properties of Gabor filter-based features—overview and applications, IEEE TIP 15 (5) (2006) 1088–1099.
[109] M. Felsberg, G. Sommer, The monogenic signal, IEEE TSP 49 (12) (2001) 3136–3144.
[110] P. Perona, J. Malik, Detecting and localizing edges composed of steps, peaks and roofs, International Conference on Computer Vision, 1990, pp. 52–57.
[111] D. Reisfeld, The constrained phase congruency feature detector: simultaneous localization, classification and scale determination, PRL 17 (11) (1996) 1161–1169.
[112] B. Robbins, R. Owens, 2d feature detection via local energy, IVC 15 (5) (1997) 353–368.
[113] S. Venkatesh, R. Owens, On the classification of image features, PRL 11 (5) (1990) 339–349.
[114] C. Ronse, On idempotence and related requirements in edge detection, IEEE T-PAMI 15 (5) (1993) 484–491.
[115] M. Kisworo, S. Venkatesh, G. West, Modeling edges at subpixel accuracy using the local energy approach, IEEE T-PAMI 16 (4) (1994) 405–410.
[116] P.L. Rosin, Edges: saliency measures and automatic thresholding, Mach. Vis. Appl. 9 (4) (1997) 139–159.
[117] J. Lee, R. Haralick, L. Shapiro, Morphologic edge detection, IEEE JRA 3 (2) (1987) 142–156.
[118] J.F. Rivest, P. Soille, S. Beucher, Morphological gradients, J. Elec. Imag. 2 (4) (1993) 326–336.
[119] A.N. Evans, X. Liu, A morphological gradient approach to color edge detection, IEEE TIP 15 (6) (2006) 1454–1463.
[120] P.E. Trahanias, A.N. Venetsanopoulos, Vector order statistics operators as color edge detectors, IEEE SMC 26 (1) (1996) 135–143.
[121] S.Y. Zhu, K.N. Plataniotis, A.N. Venetsanopoulos, Comprehensive analysis of edge detection in color image processing, Opt. Eng. 38 (1999) 612.
[122] Richard Duda, Peter Hart, David Stork, Pattern Classification, John Wiley and Sons, 2001.
[123] Peter Meer, Bogdan Georgescu, Edge detection with embedded confidence, IEEE T-PAMI 23 (12) (2001) 1351–1365.
[124] S. Konishi, A.L. Yuille, J.M. Coughlan, S.C. Zhu, Statistical edge detection: learning and evaluating edge cues, IEEE T-PAMI 25 (2003) 57–74.
[125] D.R. Martin, C. Fowlkes, J. Malik, Learning to detect natural image boundaries using brightness and texture, in: Suzanna Becker, Sebastian Thrun, Klaus Obermaye (Eds.), NIPS, MIT Press, 2002, pp. 1255–1262.
[126] N. Petkov, M.A. Westenberg, Suppression of contour perception by band-limited noise and its relation to non-classical receptive field inhibition, Biol. Cyb. 88 (2003) 236–246.
[127] T.S. Meese, R.J. Summers, D.J. Holmes, S.A. Wallis, Contextual modulation involves suppression and facilitation from the center and the surround, J. Vis. 7 (4) (2007) 7.
[128] C. Grigorescu, N. Petkov, M.A. Westenberg, Contour and boundary detection improved by surround suppression of texture edges, IVC 22 (8) (2004) 609–622.
[129] G. Papari, P. Campisi, N. Petkov, A. Neri, A biologically motivated multiresolution approach to contour detection, JASP (2007) (28 pages).
[130] G. Papari, P. Campisi, N. Petkov, A. Neri, A multiscale approach to contour detection by texture suppression, Proc. of Electronic Imaging, Image proc.: Alg. and Syst, , 2006, (60640D-1–60640D-12).
[131] G. Papari, P. Campisi, N. Petkov, Contour detection by multiresolution surround inhibition, Proc. ICIP, 2006, pp. 749–752.
[132] G. Papari, P. Campisi, N. Petkov, Multilevel surround inhibition: a biologically motivated contour detector, Proc. of Electronic Imaging, Image proc.: Alg. and Syst., , 2007, (pages 649702-1–6497-11).
[133] Mi-Suen Lee, Gérard Medioni, Grouping … into regions, curves, and junctions, CVIU 76 (1) (October 1999) 54–69.
[134] G. Guy, G. Medioni, Inferring global perceptual contours from local features, IJCV 20 (1) (1996) 113–133.
[135] W.S. Tong, C.K. Tang, Robust estimation of adaptive tensors of curvature by tensor voting, IEEE T-PAMI 27 (3) (2005) 434–449.
[136] M. Ursino, G.E. La Cara, A model of contextual interactions and contour detection in primary visual cortex, Neural Netw. 17 (5–6) (2004) 719–735.
[137] T.N. Mundhenk, L. Itti, Computational modeling and exploration of contour integration for visual saliency, Biol. Cyb. 93 (3) (2005) 188–212.
[138] D.J. Field, A. Hayes, R.F. Hess, Contour integration by the human visual system: evidence for a local "association field", Vis. Res. 33 (2) (1993) 173–193.
[139] S.C. Yen, L.H. Finkel, Extraction of perceptually salient contours by striate cortical networks, Vis. Res. 38 (5) (1998) 719–741.
[140] Zhaoping Li, A neural model of contour integration in the primary visual cortex, Neur. Comp. 10 (4) (1998) 903–940.
[141] Q. Tang, N. Sang, T. Zhang, Extraction of salient contours from cluttered scenes, PR 40 (11) (2007) 3100–3109.
[142] N. Sang, Q. Tang, T. Zhang, Contour detection based on inhibition of primary visual cortex, JIMW 47 (1) (2007).
[143] A. Rosenfeld, R.A. Hummel, S.W. Zucker, Scene labeling by relaxation operations, IEEE T-SMC 6 (6) (1976) 420–433.
[144] J.F. Boyce, J. Feng, E.R. Haddow, Relaxation labelling and the entropy of neighbourhood information, PRL 6 (4) (1987) 225–234.
[145] P. Parent, S.W. Zucker, Trace inference, curvature consistency, and curve detection, IEEE T-PAMI 11 (8) (1989) 823–839.
[146] M. Pelillo, M. Refice, Learning compatibility coefficients for relaxation labeling processes, IEEE Trans. Pattern Anal. Mach. Intell. 16 (9) (1994) 933–945.
[147] S.Z. Li, H. Wang, M. Petrou, Relaxation labeling of Markov random fields, Proceedings of IEEE International Conference on Pattern Recognition, vol. 1, 1994, pp. 488–492.
[148] S. Geman, D. Geman, Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images, IEEE Trans. Pattern Anal. Mach. Intell. 6 (6) (1984) 721–741.
[149] T. Werner, A linear programming approach to max-sum problem: a review, IEEE Trans. Pattern Anal. Mach. Intell. 29 (7) (2007) 1165–1179.
[150] J. Kittler, J. Illingworth, Relaxation labelling algorithms—a review, Image Vis. Comput. 3 (4) (1986) 206–216.
[151] O. Ben-Shahar, S.W. Zucker, The perceptual organization of texture flow: a contextual inference approach, IEEE T-PAMI 25 (4) (2003) 401–417.
[152] H. Wang, E.R. Hancock, Probabilistic relaxation labelling using the Fokker–Planck equation, Pattern Recognit. 41 (11) (2008) 3393–3411.
[153] J.S. Yedidia, W.T. Freeman, Y. Weiss, Constructing free-energy approximations and generalized belief propagation algorithms, IEEE Trans. Inf. Theory 51 (7) (2005) 2282–2312.
[154] M.J. Wainwright, T.S. Jaakkola, A.S. Willsky, A new class of upper bounds on the log partition function, IEEE Trans. Inf. Theory 51 (7) (2005) 2313–2335.
[155] M.J. Wainwright, T.S. Jaakkola, A.S. Willsky, MAP estimation via agreement on trees: message-passing and linear programming, IEEE Trans. Inf. Theory 51 (11) (2005) 3697–3717.
[156] L. Matalas, R. Benjamin, R. Kitney, An edge detection technique using the facet model and parameterized relaxation labeling, IEEE T-PAMI 19 (4) (1997) 328–341.
[157] J.H. Elder, R.M. Goldberg, Ecological statistics of Gestalt laws for the perceptual organization of contours, J. Vis. 2 (4) (2002) 324–353.
[158] J.S. Stahl, S. Wang, Globally optimal grouping for symmetric boundaries and region information, IEEE Trans. Pattern Anal. Mach. Intell. 30 (3) (2008) 395–411.
[159] A. Amir, M. Lindenbaum, A generic grouping algorithm and its quantitative analysis, IEEE T-PAMI 20 (2) (1998) 168–185.
[160] L.R. Williams, K.K. Thornber, A comparison of measures for detecting natural shapes in cluttered backgrounds, IJCV 34 (2) (1999) 81–96.
[161] A. Sanfeliu, R. Alquézar, J. Andrade, J. Climent, F. Serratosa, J. Vergés, Graph-based representations and techniques for image processing and image analysis, PR 35 (3) (2002) 639–650.
[162] A.D. Sappa, Unsupervised contour closure algorithm for range image edge-based segmentation, IEEE TIP 15 (2) (2006) 377–384.
[163] X. Ren, C. Fowlkes, J. Malik, Scale-invariant contour completion using conditional random fields, ICCV, IEEE Computer Society, Washington, DC, USA, 2005, pp. 1214–1221.
[164] S. Mahamud, L.R. Williams, K.K. Thornber, K. Xu, Segmentation of multiple salient closed contours from real images, IEEE T-PAMI 25 (4) (2003) 433–444.
[165] I. Kovacs, B. Julesz, A closed curve is much more than an incomplete one: effect of closure in figure-ground segmentation, PNAS 90 (16) (1993) 7495–7497.
[166] J. Elder, S. Zucker, A measure of closure, Vis. Res. 34 (24) (1994) 3361–3370.
[167] S. Casadei, S. Mitter, Beyond the uniqueness assumption: ambiguity representation and redundancy elimination in the computation of a covering sample of salient contour cycles, CVIU 76 (1) (1999) 19–35.
[168] E. Saund, Finding perceptually closed paths in sketches and drawings, IEEE T-PAMI 25 (4) (2003) 475–491.
[169] E. Saund, Perceptual organization of occluding contours of opaque surfaces, CVIU 76 (1) (1999) 70–82.
[170] D.W. Jacobs, Robust and efficient detection of salient convex groups, IEEE Trans. Pattern Anal. Mach. Intell. 18 (1) (1996) 23–37.
[171] J.H. Elder, S.W. Zucker, Computing contour closure, Lect. Notes Comput. Sci. 1064 (1996) 399–412.
[172] K.K. Thornber, L.R. Williams, Analytic solution of stochastic completion fields, Biol. Cybern. 75 (2) (1996) 141–151.
[173] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, 2001.
[174] S. Wang, T. Kubota, J.M. Siskind, J. Wang, Salient closed boundary extraction with ratio contour, IEEE T-PAMI 27 (4) (2005) 546–561.
[175] L. Lam, S.W. Lee, C.Y. Suen, Thinning methodologies—a comprehensive survey, IEEE Trans. Pattern Anal. Mach. Intell. 14 (9) (1992) 869–885.
[176] H.I. Choi, S.W. Choi, H.P. Moon, Mathematical theory of medial axis transform, Pac. J. Math. 181 (1) (1997) 57–88.
[177] H. Shroff, J. Ben-Arie, Finding shape axes using magnetic fields, IEEE Trans. Image Process. 8 (10) (1999) 1388–1394.
[178] K. Siddiqi, S. Bouix, A. Tannenbaum, S.W. Zucker, Hamilton–Jacobi skeletons, Int. J. Comput. Vision 48 (3) (2002) 215–231.
[179] X. Bai, L.J. Latecki, W.Y. Liu, Skeleton pruning by contour partitioning with discrete curve evolution, IEEE Trans. Pattern Anal. Mach. Intell. 29 (3) (2007) 449–462.
[180] R. Mohan, R. Nevatia, Perceptual organization for scene segmentation and description, IEEE Trans. Pattern Anal. Mach. Intell. 14 (6) (1992) 616–635.
[181] T. Tuytelaars, A. Turina, L. Van Gool, Noncombinatorial detection of regular repetitions under perspective skew, IEEE Trans. Pattern Anal. Mach. Intell. 25 (4) (2003) 418–432.

[182] H. Cornelius, G. Loy, Detecting bilateral symmetry in perspective, Workshop on Perceptual Organization in Computer Vision, 2006, pp. 191–198.
[183] X. Jiang, An adaptive contour closure algorithm and its experimental evaluation, IEEE T-PAMI 22 (11) (2000) 1252–1265.
[184] G. Papari, N. Petkov, Adaptive pseudo dilation for Gestalt edge grouping and contour detection, IEEE Trans. Image Process. 17 (10) (2008) 1950–1962.
[185] P.F.M. Nacken, A metric for line segments, IEEE T-PAMI 15 (12) (1993) 1312–1318.
[186] A.M. Martı´nez, P. Mittrapiyanuruk, A.C. Kak, On combining graph-partitioning with non-parametric clustering for image segmentation, CVIU 95 (1) (2004) 72–85.
[187] E. Engbers, A.W.M. Smeulders, Design considerations for generic grouping in vision, IEEE T-PAMI 25 (4) (2003) 445–457.
[188] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, IJCV 1 (4) (1987) 321–331.
[189] V. Caselles, R. Kimmel, G. Sapiro, Geodesic active contours, IJCV 22 (1) (1997) 61–79.
[190] S. Osher, J. Sethian, Fronts propagating with curvature-dependent speed— algorithms based on Hamilton–Jacobi formulations, J. Comp. Phys. 79 (1) (1988) 12–49.
[191] T.O.C. View, A topology preserving level set method for geometric deformable models, IEEE T-PAMI 25 (6) (2003) 755–768.
[192] G. Sundaramoorthi, A. Yezzi, Global regularizing flows with topology preservation for active contours and polygons, IEEE TIP 16 (3) (2007) 803–812.
[193] S. Osher, R.P. Fedkiw, Level set methods— an overview and some recent results, J. Comp. Phys. 169 (2) (2001) 463–502.
[194] L.H. Staib, J.S. Duncan, Boundary finding with parametrically deformable models, IEEE T-PAMI 14 (11) (1992) 1061–1075.
[195] L.D. Cohen, I. Cohen, Finite-element methods for active contour models and balloons for 2-d and 3-d images, IEEE T-PAMI 15 (11) (1993) 1131–1147.
[196] S.R. Gunn, M.S. Nixon, A robust snake implementation; a dual active contour, IEEE T-PAMI 19 (1) (1997) 63–68.
[197] H. Yan, Fuzzy curve-tracing algorithm, IEEE SMC 31 (5) (2001) 768–780.
[198] H. Yan, Convergence condition and efficient implementation of the fuzzy curve-tracing (fct) algorithm, IEEE SMC 34 (1) (2004) 210–221.
[199] D. Geiger, A. Gupta, L.A. Costa, J. Vlontzos, Dynamic programming for detecting, tracking, and matching deformable contours, IEEE T-PAMI 17 (3) (1995) 294–302.
[200] A.C. Jalba, M.H.F. Wilkinson, J. Roerdink, Cpm: a deformable model for shape recovery and segmentation based on charged particles, IEEE T-PAMI 26 (10) (2004) 1320–1335.
[201] H.W. Park, T. Schoepflin, Y. Kim, Active contour model with gradient directional information: directional snake, IEEE TRANS. Circ. Sys. Video Tech. 11 (2) (2001) 252–256.
[202] A. Dumitras, A.N. Venetsanopoulos, Angular map-driven snakes with application to object shape description in color images, IEEE TIP 10 (12) (2001) 1851–1859.
[203] J. Liang, T. McInerney, D. Terzopoulos, United snakes. Med. Im. Anal. 10 (2) (2006) 215–233.
[204] A. Blake, M. Isard, Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion, Springer-Verlag, New York, 1998 (Inc. Secaucus, NJ, USA).
[205] J.S. Suri, K. Liu, S. Singh, S.N. Laxminarayan, X. Zeng, L. Reden, Shape recovery algorithms using level sets in 2-d/3-d medical imagery: a state-of-the-art review, IEEE ITB 6 (1) (2002) 8–28.
[206] C. Xu, J.L. Prince, Snakes, shapes, and gradient vector flow, IEEE TIP 7 (3) (1998) 359–369.
[207] N. Paragios, O. Mellina-Gottardo, V. Ramesh, Gradient vector flow fast geometric active contours, IEEE T-PAMI 26 (3) (2004) 402–407.
[208] M.L. Sakalli, K.M.H. Yan, A faster converging snake algorithm to locate object boundaries, IEEE TIP 15 (5) (2006) 1182–1191.
[209] X.M. Pardo, D. Cabello, Biomedical active segmentation guided by edge saliency, PRL 21 (6–7) (2000) 559–572.
[210] R. Ronfard, Region-based strategies for active contour models, IJCV 13 (2) (1994) 229–251.
[211] T.F. Chan, L.A. Vese, Active contours without edges, IEEE TIP 10 (2) (2001) 266–277.
[212] H. Li, A.J. Yezzi, Local or global minima: flexible dual-front active contours, IEEE T-PAMI 29 (1) (2007) 1–14.
[213] N. Paragios, R. Deriche, Geodesic active regions and level set methods for supervised texture segmentation, IJCV 46 (3) (2002) 223–247.
[214] L.A. Vese, T.F. Chan, A multiphase level set framework for image segmentation using the Mumford and Shah model, IJCV 50 (3) (2002) 271–293.
[215] T. Chan, B. Sandberg, L. Vese, Active contours without edges for vector-valued images, J. Vis. Comm. Image Repr. 11 (2) (2000) 130–131.
[216] M.E. Leventon, W.E.L. Grimson, O.D. Faugeras, Statistical shape influence in geodesic active contours, CVPR, 2000, pp. 316–323.
[217] A. Tsai, A. Yezzi Jr, W. Wells, C. Tempany, D. Tucker, A. Fan, W.E. Grimson, A. Willsky, A shape-based approach to the segmentation of medical imagery using level sets, IEEE TMI 22 (2) (2003) 137–154.
[218] J.A. Sethian, Level Set Methods, Cambridge University Press, 1996.
[219] D. Cremers, F. Tischhäuser, J. Weickert, C. Schnörr, Diffusion snakes: introducing statistical shape knowledge into the Mumford–Shah functional, IJCV 50 (3) (2002) 295–313.
[220] D. Cremers, T. Kohlberger, C. Schnörr, Shape statistics in kernel space for variational image segmentation, PR 36 (9) (2003) 1929–1943.
[221] X. Bresson, P. Vandergheynst, J.P. Thiran, A variational model for object segmentation using boundary information and shape prior driven by the Mumford–Shah functional, IJCV 68 (2) (2006) 145–162.
[222] A. Kostin, J. Kittler, W. Christmas, Object recognition by symmetrised graph matching using relaxation labelling with an inhibitory mechanism, Pattern Recognit. Lett. 26 (3) (2005) 381–393.
[223] F.M. Ansia, M.G. Penedo, C. Marino, A. Mosquera, J. Lopez, Automatic 3D shape reconstruction of bones using active net based segmentation, International Conference on Pattern Recognition, 2000, pp. 486–489.
[224] J.J. Koenderink, The structure of images, Biol. Cyb. 50 (1984) 363–370.
[225] S.G. Mallat, Multifrequency channel decomposition of images and wavelet models, IEEE Trans. Acoust. Speech, Signal Proces. 37 (12) (1989) 2091–2110.
[226] J.H. Elder, S.W. Zucker, Local scale control for edge detection and blur estimation, IEEE T-PAMI 20 (7) (1998) 699–716.
[227] S.G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE T-PAMI 11 (7) (1989) 674–693.
[228] I. Daubechies, The wavelet transform, time–frequency localization and signal analysis, IEEE TIT 36 (5) (1990) 961–1005.
[229] D.L. Donoho, I.M. Johnstone, Ideal spatial adaptation via wavelet shrinkage, Biom. 81 (s 425) (1992) 425–455.
[230] A.L. Yuille, T.A. Poggio, Scaling theorems for zero crossings, IEEE T-PAMI 8 (1) (1986) 15–25.
[231] J. Babaud, A.P. Witkin, M. Baudin, R.O. Duda, Uniqueness of the Gaussian kernel for scale–space filtering, IEEE T-PAMI 8 (1) (1986) 26–33.
[232] L. Alvarez, F. Guichard, P.L. Lions, J.M. Morel, Axioms and fundamental equations of image processing, Arch. Rat. Mech. Anal. 123 (3) (1993) 199–257.
[233] J. Weickert, S. Ishikawa, A. Imiya, Linear scale–space has first been proposed in Japan, J. Math. Imag. Vis. 10 (3) (1999) 237–252.
[234] E.J. Pauwels, L.J. van Gool, P. Fiddelaers, T. Moons, An extended class of scale-invariant and recursive scale space filters, IEEE T-PAMI 17 (7) (1995) 691–701.
[235] R. Duits, L. Florack, J. de Graaf, B. ter Haar Romeny, On the axioms of scale space theory, J. Math. Imag. Vis. 20 (3) (2004) 267–298.
[236] M. Felsberg, G. Sommer, The monogenic scale–space: a unifying approach to phase-based image processing in scale–space, J. Math. Imag. Vis. 21 (1) (2004) 5–26.
[237] M. Felsberg, R. Duits, L. Florack, The monogenic scale space on a rectangular domain and its features, IJCV 64 (2) (2005) 187–201.
[238] Bernhard Burgeth, Stephan Didas, Joachim Weickert, The Bessel scale–space, *DSSCV*, volume 3753 of *Lecture Notes in Computer Science*, Springer, 2005, pp. 84–95.
[239] B. Burgeth, S. Didas, J. Weickert, Relativistic scale–spaces, Scale Space, 2005, pp. 1–12.
[240] R. Duits, M. Felsberg, L.M.J. Florack, B. Platel, Alpha scale spaces on a bounded domain, scale space, 2003, pp. 494–510.
[241] Remco Duits, Frans Kanters, Luc Florack, M. Bart, ter Haar Romeny, A comparison of the deep structure of alpha-scale spaces, *DSSCV*, volume 3753 of *Lecture Notes in Computer Science*, Springer, 2005, pp. 234–248.
[242] D. Marr, Vision, Freeman, San Francisco, 1982.
[243] G. Steidl, J. Weickert, T. Brox, P. Mrázek, M. Welk, On the equivalence of soft wavelet shrinkage, total variation diffusion, total variation regularization, and SIDEs, SIAM J. Num. Anal. 42 (2) (April 2004) 686–713.
[244] J. Xu, S. Osher, Iterative regularization and nonlinear inverse scale space applied to wavelet-based denoising, IEEE TIP 16 (2) (2007) 534–544.
[245] M. Nielsen, L. Florack, R. Deriche, Regularization, scale–space, and edge detection filters, J. Math. Imag. Vis. 7 (4) (1997) 291–307.
[246] R. van den Boomgaard, A. Smeulders, The morphological structure of images: the differential equations of morphological scale–space, IEEE T-PAMI 16 (11) (1994) 1101–1113.
[247] P.T. Jackway, M. Deriche, Scale–space properties of the multiscale morphological dilation–erosion, IEEE T-PAMI 18 (1) (1996) 38–51.
[248] A.H. Salden, J. Weickert, B.M.t.H Romenij, Bluman and Kumei's nonlinear scale–space theory, Technical Report, ERCIM-02/99-R067, 1999.
[249] A. Chambolle, R.A. DeVore, N. Lee, B.J. Lucier, Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage, IEEE TIP 7 (3) (1998) 319.
[250] B. Vidakovic, Nonlinear wavelet shrinkage with Bayes rules and Bayes factors, J. Amer. Stat. Ass. 93 (441) (1998).
[251] A. Chambolle, B.J. Lucier, Interpreting translation-invariant wavelet shrinkage as a new image smoothing scale space, IEEE TIP 10 (7) (2001) 993–1000.
[252] B.M. ter Haar Romeny, Geometry-Driven Diffusion in Computer Vision, Kluwer, September, 1994.
[253] Joachim Weickert, A review of nonlinear diffusion filtering, in: B. ter Haar Romney, L. Florack, J. Koenderink, M. Viergever (Eds.), *Scale–Space Theory in Computer Vision*, 1252, Springer-Verlag, 1997, pp. 3–28.
[254] A.H. Salden, B.M.T.H. Romeny, M.A. Viergever, A dynamic scale–space paradigm, J. Math. Imag. Vis. 15 (3) (2001) 127–168.
[255] Y. Lu, R.C. Jain, Behavior of edges in scale space, IEEE T-PAMI 11 (4) (April 1989) 337–356.
[256] L. Florack, B. Ter Haar Romeny, M. Viergever, J. Koenderink, The Gaussian scale–space paradigm and the multiscale local jet, IJCV 18 (1) (1996) 61–75.
[257] F. Bergholm, Edge focusing, IEEE T-PAMI, PAMI 9 (6) (1987) 726–741 (November).
[258] F. Ulupinar, G.G. Medioni, Refining edges detected by a LoG operator, CVGIU 51 (1990) 275–298.
[259] T. Lindeberg, Detecting salient blob-like image structures and their scale with a scale–space primal sketch: a method for focus-of-attention, IJCV 11 (3) (1993) 283–318.
[260] A. Goshtasby, On edge focusing, IVC 12 (4) (1994) 247–256.
[261] J. Sun, D. Gu, Y. Chen, S. Zhang, A multiscale edge detection algorithm based on wavelet domain vector hidden Markov tree model, PR 37 (7) (2004) 1315–1324.
[262] S.V. Raman, S. Sarkar, K.L. Boyer, Tissue boundary refinement in magnetic resonance images using contour-based scale space matching, IEEE T-Med. Images 10 (2) (1991) 109–121.

[263] X.M. Pardo, M.J. Carreira, A. Mosquera, D. Cabello, A snake for CT image segmentation integrating region and edge information, IVC 19 (7) (2001) 461–475.
[264] N. Ray, B. Chanda, J. Das, A fast and flexible multiresolution snake with a definite termination criterion, PR 34 (7) (2001) 1483–1490.
[265] W. Liu, J.A. Zagzebski, T. Varghese, C.R. Dyer, U. Techavipoo, T.J. Hall, Segmentation of elastographic images using a coarse-to-fine active contour model, Ultras. Med. Biol. 32 (3) (2006) 397–408.
[266] M. Li, P.S. Wu, Pyramid edge detection for color images, Opt. Eng. 36 (5) (1997) 1431–1437.
[267] F. Sjöberg, F. Bergholm, Extraction of diffuse edges by focusing, PRL 7 (3) (1988) 181–190.
[268] W. Zhang, F. Bergholm, Multi-scale blur estimation and edge type classification for scene analysis, IJCV 24 (3) (1997) 219–250.
[269] T. Lindeberg, Edge detection and ridge detection with automatic scale selection, IJCV 30 (2) (November 1998) 117–156.
[270] D.J. Park, K.M. Nam, R.H. Park, Multiresolution edge detection techniques, PR 28 (2) (1995) 211–229.
[271] C.F. Olson, Adaptive-scale filtering and feature detection using range data, IEEE T-PAMI 22 (9) (2000) 983–991.
[272] M. Heath, S. Sarkar, T. Sanocki, K.W. Bowyer, Comparison of edge detectors, CVIU 69 (1) (1998) 38–54.
[273] M.D. Heath, S. Sarkar, T. Sanocki, K.W. Bowyer, A robust visual method for assessing the relative performance of edge-detection algorithms, IEEE T-PAMI 19 (12) (1997) 1338–1359.
[274] M.C. Shin, D.B. Goldgof, K.W. Bowyer, S. Nikiforou, Comparison of edge detection algorithms using a structure from motion task, IEEE T-SMC 31 (4) (2001) 589–601.
[275] L. Kitchen, A. Rosenfeld, Edge evaluation using local edge coherence, IEEE T-SMC 11 (9) (1981) 597–605.
[276] Q. Zhu, Efficient evaluations of edge connectivity and width uniformity, IVC 14 (1) (1996) 21–34.
[277] R.M. Haralick, Digital step edges from zero crossing of second directional derivatives, IEEE T-PAMI 6 (1) (1984) 58–68.
[278] R.M. Haralick, J.S.J. Lee, Context dependent edge detection and evaluation, PR 23 (1–2) (1990) 1–19.
[279] Y. Yitzhaky, E. Peli, A method for objective edge detection evaluation and detector parameter selection, IEEE T-PAMI 25 (8) (2003) 1027–1033.
[280] Y.T. Zhou, V. Venkateswar, R. Chellappa, Edge detection and linear feature extraction using a 2-d random field model, IEEE T-PAMI 11 (1) (1989) 84–95.
[281] K. Bowyer, C. Kranenburg, S. Dougherty, Edge detector evaluation using empirical ROC curves, CVIU 84 (1) (2001) 77–103.
[282] S. Chabrier, H. Laurent, B. Emile, C. Rosenberger, P. Marche, A comparative study of supervised evaluation criteria for image segmentation, “EUSIPCO”, 2004, pp. 1143–1146, (EURASIP).
[283] D.P. Huttenlocher, G.A. Klanderman, W.J. Rucklidge, Comparing images using the Hausdorff distance, IEEE T-PAMI 15 (9) (1993) 850–863.
[284] C. Odet, B. Belaroussi, H. Benoit-Cattin, Scalable discrepancy measures for segmentation evaluation, ICIP, 2002, pp. 785–788.
[285] Q. Tang, N. Sang, T. Zhang, Contour detection based on contextual influences, IVC 25 (8) (2007) 1282–1290.
[286] R.A. Hummel, S.W. Zucker, On the foundation of relaxation labeling process, IEEE T-PAMI 5 (3) (1983) 267–286.
[287] Daniel Crevier, A probabilistic method for extracting chains of collinear segments, CVIU 76 (1) (1999) 36–53.
[288] M.H.F. Wilkinson, Optimizing edge detectors for robust automatic threshold selection: coping with edge curvature and noise, Graphical Models and Image Process. 60 (5) (1998) 385–401.
[289] P.V. Henstock, D.M. Chelberg, Automatic gradient threshold determination for edge detection, IEEE Trans. Image Process. 5 (5) (1996) 784–787.
[290] S. Venkatesh, P. Rosin, Dynamic threshold determination by local and global edge evaluation, Graphical Models Image Process. 57 (2) (1995) 146–160.
[291] L. Itti, C. Koch, A saliency-based search mechanism for overt and covert shifts of visual attention, Vis. Res. 40 (10–12) (2000) 1489–1506.
[292] M. Rousson, N. Paragios, Prior knowledge, level set representations & visual grouping, Int. J. Comput. Vision 76 (3) (2008) 231–243.

View publication stats