

# Accurate Junction Detection and Characterization in Natural Images

Gui-Song Xia · Julie Delon · Yann Gousseau

Received: 12 October 2011 / Accepted: 26 June 2013  
© Springer Science+Business Media New York 2013

**Abstract** Accurate junction detection and characterization are of primary importance for several aspects of scene analysis, including depth recovery and motion analysis. In this work, we introduce a generic junction analysis scheme. The first asset of the proposed procedure is an automatic criterion for the detection of junctions, permitting to deal with textured parts in which no detection is expected. Second, the method yields a characterization of L-, Y- and X- junctions, including a precise computation of their type, localization and scale. Contrary to classical approaches, scale characterization does not rely on the linear scale-space. First, an *a contrario* approach is used to compute the meaningfulness of a junction. This approach relies on a statistical modeling of suitably normalized gray level gradients. Then, exclusion principles between junctions permit their precise characterization. We give implementation details for this procedure and evaluate its efficiency through various experiments.

**Keywords** Junction detection · Scale characterization · *a contrario* method · Scale-invariant keypoints · Contrast invariance

---

G.-S. Xia (✉)  
State Key Lab. LIESMARS, Wuhan University,  
129 Luoyu Road, Wuhan 430079, China  
e-mail: guisong.xia@whu.edu.cn

J. Delon · Y. Gousseau  
Telecom ParisTech, LTCI CNRS, 46, rue Barrault,  
75013 Paris, France  
e-mail: delon@telecom-paristech.fr

Y. Gousseau  
e-mail: gousseau@telecom-paristech.fr

## 1 Introduction

Junctions are of primary importance for visual perception and scene understanding. They are parts of the well known *primal sketch*, the schematic representation of images introduced by (Marr 1982). Recent approaches to the computation of this sketch, as proposed by (Guo et al. 2007), show the key role played by junctions. Depending on the number of edges they connect, junctions are often classified into L-, Y- (or T-) and X-junctions. In particular, the role of T-junctions as cues for the perception of occlusions has been thoroughly studied by (Kanizsa 1979). Later, it has been shown in (Rubin 2001) that junctions are essential *local* cues to initiate contour completion and that their specific configuration (e.g. T- or Y-junction) has to be taken into account in this process. The distinct roles of L- and T-junctions for the perception of motion, in particular through the aperture phenomenon, has been known for long (Wallach 1935). Junction types and positions are also shown to have a strong impact on the perception of brightness and transparency, as investigated in (Metelli 1974; Adelson 2000).

Junctions are therefore naturally used as important cues for various computer vision tasks. Since they reveal important occlusion relationships between objects, they are involved in figure/ground separation (Geiger et al. 1996; Ren et al. 2006; Leichter and Lindenbaum 2009; Dimiccoli and Salembier 2009). Junctions are also used for grouping edges and regions to achieve image segmentations (Fuchs and Förstner 1995; Lindeberg and Li 1997; Ishikawa and Geiger 1998). The role of junction for object recognition has been studied since the early works on automatic scene analysis, see (Guzmán 1968; Bolles and Cain 1982; Lin et al. 2007).

In the present work, we propose a principled and automatic method for the detection, localization and characterization of junctions. We define junctions as local or semi-local

structures at the center of which at least two dominant and different edge directions meet. Junctions are defined locally, and not for instance as a by-product of a global image segmentation, mostly to achieve precise description of the junction. Following observations from psychophysics (McDermott 2004), we nevertheless consider large regions for the assessment of a junction. This is made possible by an automatic scale selection rule. It should be noticed that not all perceived junctions enter this framework, and that some of them are only detected while performing scene analysis (McDermott 2004). Nevertheless, as advocated above, we believe that a precise and accurate description of junction makes them valuable information in view of a more global scene analysis.

More precisely, we take interest in a junction detection method meeting several requirements. First, junctions have to be clearly related to the geometry of the image. In particular, the description of the junction branches should be accurate. This is in strong contrast with the classical corner detectors such as Harris, that are commonly used as keypoint detectors. Second, an automatic validation rule will be derived. For this, we rely on the *a contrario* methodology (Desolneux et al. 2000), in which structures are validated by controlling the number of false detections. A key advantage of this approach is the automatic setting of detection parameters in a way that will prevent the numerous junctions that are usually detected in textured areas. Third, the position and scale of the junctions should be detected precisely and be closely related to the image geometry. This will be achieved thanks to a competition between junctions relying on a sound quality measure, the *number of false alarms* associated with a junction. A shorter version of this work has appeared in (Xia et al. 2012).

## 1.1 Related Works

### 1.1.1 Corner Detectors

Automatic junction detection has been a very active research field over the last four decades. Some of the earliest methods were introduced by (Hannah 1974) and (Moravec 1977), both considering interest points as points which are not similar to their neighbors, the similarity being assessed either by correlation or by a quadratic distance between patches. It has then been proposed, in (Förstner 1986) and (Harris and Stephens 1988), to compute derivative of similarity measures, and therefore to rely on the so-called Harris matrix to detect corners. Such approaches are widely used in practice and a large number of detectors rely on similar ideas, the detection of corners boiling down to the analysis of the eigenvalues of this matrix, see e.g. (Shi and Tomasi 1994; Förstner 1994; Kovese 2003; Kenney et al. 2005). In particular, in order to achieve contrast invariance, (Kovese 2003)

uses phase congruency to derive cornerness measurement. In this work, the image gradient is normalized over small wedges. Alternative measurements of self-similarity have also been proposed, such as the univalue segment assimilating nucleus (USAN) (Smith and Brady 1995) and its variants (Trajkovic and Hedley 1998; Rosten et al. 2010). Observe that such methods relying on a measure of cornerness actually do not distinguish between different types of junctions. They are therefore usually used to compute generic interest points and their use as local cues for occlusion analysis or figure/ground separation is less clear. Also observe that in order to identify the characteristic scale of a corner, corner detectors usually make use of the linear scale-space (Deriche and Giraudon 1993; Lindeberg 1994; Schmid et al. 2000; Lowe 2004; Mikolajczyk and Schmid 2004). One of the main shortcomings of such approaches is that they quickly lose precision both in localization and scale. A notable exception is the work of (Förstner et al. 2009) where the keypoint locations are optimized and their type are identified. For an overview of the use of linear scale-space for keypoint detection, one may consult (Dickscheid et al. 2011). An other approach is to rely on some contrast invariant multi-scale analysis. In this direction, (Alvarez and Morales 1997) analyze junctions in images using an affine morphological scale space.

### 1.1.2 Boundary Based Methods

A second popular and efficient way to detect corners relies on explicit boundaries in images. Corners are interpreted as points in the image where a boundary changes direction rapidly and points with high curvature are selected. Many works have concentrated on different and efficient ways to compute the curvature of curves for detecting corners, see (Rosenfeld and Johnston 1973; Teh and Chin 1989; Rattarangsi and Chin 1992; Wang and Brady 1995; Mokhtarian and Suomela 1998; Mohammad and Lu 2008). In a different way, it has been proposed in (Beymer 1991) to detect junctions by extending disconnected edges and by filling gaps at junctions. Junction detection can also rely on the grouping of edges in the neighborhood of a candidate junction (Freeman 1992; Rohr 1992; Perona 1992; Bergevin and Bubel 2004). In a related direction, it is proposed in (Rosenthaler et al. 1992) to combine the responses of several oriented filters to detect junctions. Several works have proposed to rely on the contrast invariant level lines of images to detect junctions. (Caselles et al. 1996) have proposed a method based on a level-set representation, detecting junctions as points where two level lines meet. Cao (2003) detects corners as points along level lines where a good continuation criterion is broken. Corners are detected as points where the curvature along a level line is abnormally high, in an *a contrario* framework. In other approaches, a global seg-

mentation of the image is first performed, followed by a heuristic grouping of edges to form junctions (Wu et al. 2007; Maire et al. 2008). It should be emphasized that the segmentation process in itself is a very challenging task. For instance, before detecting junctions, (Maire et al. 2008) finds contours by the *global probability of boundary* (gPb) detector learned from a human-annotated database. Such approaches permit to benefit from global image interpretation to find junctions. On the other hand, boundary are often imprecise in the neighborhood of a junction, so that such approaches do not enable an accurate characterization of junctions, that in turn can be used to refine edges.

### 1.1.3 Template-based Methods

Among existing approaches, the model-based or template-based ones are the most suitable for accurate local junction detection. Deriche and Blaszkia (1993) presents computational approaches for a model-based detection of junctions. A junction model is a 2D intensity function depending on several parameters. Starting from a poorly localized initialization (e.g. from the Harris detector) parameters are then optimized in view of a precise localization. Parida et al. (1998) suggest a region-based model for simultaneously detecting, classifying and reconstructing junctions. A junction is defined as an image region containing piecewise constant wedges meeting at the central point. This work relies on a template deformation framework and uses minimum description length principle and dynamic programming to obtain the optimal parameters describing the model. This work also involves junction candidates provided by a preliminary corner detector. Ruzon and Tomasi (2001) model junctions as points in images where two or more image regions meet. Regions are described by their color distributions, which allows textured objects with the same average color to be distinguished. Following the model of Parida et al. (1998), (Cazorla and Escolano 2003; Cazorla et al. 2002) propose both a region-based and an edge-based model for junction classification by using Bayesian methods. The region-based one formalizes junction detection as a radial image segmentation problem under a regions competition framework and the edge-based one detects junctions as radial edges minimizing some Bayesian classification error. In this work, the edge-based approach is shown to yield more accurate junction detections, but at the price of a large number of false detections, especially in textured regions. More recently, the work of (Sinzing 2008) first detects a set of junction candidates by using a preliminary detector (for instance the Harris detector) and then refines those candidates by relying on a radial edge energy.

The different approaches for junction detection, as well as their properties, are summarized in Table 1.

## 1.2 Contributions and Outline

As explained earlier in the introduction, the goal of this paper is to design a junction detection procedure involving an automatic decision step and permitting a geometrically accurate description of junction properties, including type, localization and scale. This is made possible thanks to the use of an *a contrario* methodology (Desolneux et al. 2000). The resulting detection method will be coined *ACJ* (for a *con-trario* junction detection). In short, we first define (in Sect. 2) junctions as geometrical structures in discrete images and we associate to each candidate junction a quantity called *strength*, which is designed to be high at locations where several edges (called branches) meet in the image. This strength relies on a well chosen normalization of the gradient, which makes the whole approach robust to local contrast changes. We then rely in Sect. 3 on a statistical framework in order to decide automatically which junctions deserve to be detected or not in a given image. More precisely, meaningful junctions are detected as those which are very unlikely under some null hypothesis  $\mathcal{H}_0$ . This detection step is followed by several exclusion criteria, designed to rule out redundant junctions and to identify precisely the correct scales, positions and complexities (number of branches) of junctions. As a by-product of the scale estimation, the whole approach is robust to resolution changes. Finally, in Sect. 4, details about implementation and parameter settings are provided. Section 5 analyzes the proposed algorithm experimentally.

## 2 Contrast Invariant Junctions

The aim of this section is to define junctions in discrete images (Sect. 2.2) and to associate to each candidate junction a *strength* (Sect. 2.3). The strength of a junction is defined through both the gradient orientation and magnitude. Robustness to local contrast changes is achieved by normalizing the gradient magnitude, as explained in Sect. 2.1. The strength of junctions will be used in Sect. 3 to decide whether junctions are meaningful or not in a given image.

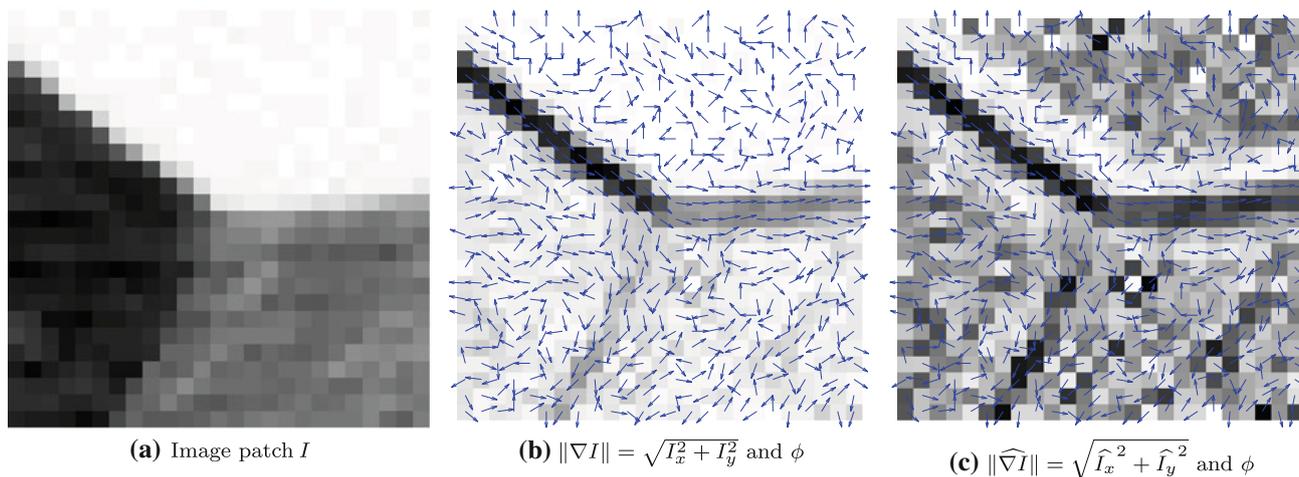
### 2.1 Gradient Normalization

Let us start with some definitions and vocabulary that will be used throughout the paper. A discrete image is a function  $I : \Omega \rightarrow \mathbb{R}$ , where  $\Omega$  is a rectangular subset of  $\mathbb{Z} \times \mathbb{Z}$ . We write  $\nabla I = (I_x, I_y)$  for the discrete gradient of the image  $I$ .<sup>1</sup> For a pixel  $\mathbf{q}$  in  $\Omega$ , we define  $\phi(\mathbf{q}) = (\arctan \frac{I_y(\mathbf{q})}{I_x(\mathbf{q})} + \pi/2) \text{ modulo } (2\pi)$ , the direction perpendicular to the gradient

<sup>1</sup> In this paper, we use the following discrete scheme to compute the gradient: for a pixel  $\mathbf{q} = (x, y)$ ,  $I_x(\mathbf{q}) = \frac{1}{2}(I(x+1, y) - I(x-1, y))$  and  $I_y(\mathbf{q}) = \frac{1}{2}(I(x, y+1) - I(x, y-1))$ .

**Table 1** Classification of junction detection methods

Method	Cornerness-based	Contour-based	Template-based	Our approach (template-based)
Some representative works	Harris and Stephens 1988; Lindeberg 1994; Smith and Brady 1995; Trajkovic and Hedley 1998; Kovesi 2003; Mikolajczyk and Schmid 2004; Förstner et al. 2009; Rosten et al. 2010	Rohr 1992; Caselles et al. 1996; Wu et al. 2007; Maire et al. 2008; Mohammad and Lu 2008	Deriche and Blaszkzka 1993; Parida et al. 1998; Cazorla and Escolano 2003; Sinzinger 2008	—
Junction detection cues	Cornerness measurements, such as the Harris matrix.	Edge map	Consistence with gradient-based or region-based templates	Amplitudes and phases of gradients
Scale of junctions	Possibility of scale detection relying on the linear scale space (Lindeberg 1994; Mikolajczyk and Schmid 2004).	No scale definition.	Junctions are detected at a given scale (except : scale selection by MDL in Parida et al. 1998)	Each junction is associated with an adaptive scale
Classification of junctions	No	No	Yes	Yes
Contrast changes	Dependent (except in Kovesi 2003)	Dependent	Dependent	Robust
Detection criterion	Thresholds on local cornerness	Heuristic grouping principles	Energy minimization from a junction model	Probabilistic perceptual principles



**Fig. 1** Gradient  $\nabla I$  and normalized gradient  $\widehat{\nabla I}$  on an image patch

at  $\mathbf{q}$  (and set  $\phi(\mathbf{q}) = \pi$  when  $I_x(\mathbf{q}) = 0$ ). We call this angle the *direction* of the pixel  $\mathbf{q}$ .

In order to be robust to local contrast changes, we locally normalize the gradient magnitude by dividing it by its average on a small neighborhood. That is, for  $\mathbf{q} = (x, y)$ , we define  $\widehat{\nabla I} = (\widehat{I}_x, \widehat{I}_y)$  as

$$\widehat{I}_x(\mathbf{q}) = \frac{I_x(\mathbf{q})}{\langle \sqrt{I_x^2 + I_y^2} \rangle_{\mathcal{N}_q}} \quad \text{and} \quad \widehat{I}_y(\mathbf{q}) = \frac{I_y(\mathbf{q})}{\langle \sqrt{I_x^2 + I_y^2} \rangle_{\mathcal{N}_q}}, \tag{1}$$

where  $\mathcal{N}_q$  is a small neighborhood around  $\mathbf{q}$  and  $\langle \cdot \rangle_{\mathcal{N}_q}$  is the linear average operator on  $\mathcal{N}_q$ . The resulting gradient is robust to contrast changes that can be approximated by affine transformations on each neighborhood  $\mathcal{N}_q$ . An example of the gradient and normalized gradient of an image, obtained with a square neighborhood of size  $5 \times 5$  around each pixel, is shown in Fig. 1. Observe that the phase of the normalized gradient is the same as the phase of the plain gradient.

In Sect. 3, we will rely on the empirical distribution of the gradient to detect meaningful junctions. More precisely, we will consider the globally *normalized gradient*

$$\widetilde{I}_x = \frac{\widehat{I}_x - \mu_x}{\sigma_x} \quad \text{and} \quad \widetilde{I}_y = \frac{\widehat{I}_y - \mu_y}{\sigma_y}, \tag{2}$$

where  $\mu_x$  (resp.  $\mu_y$ ) and  $\sigma_x$  (resp.  $\sigma_y$ ) are the empirical mean and standard deviation of  $\widehat{I}_x$  (resp.  $\widehat{I}_y$ ) over the whole image. In the following paragraphs, this globally normalized gradient will be used to weight the contribution of local orientations in order to define the strength of a junction. As we will see, the distribution of the norm  $\|\widehat{\nabla I}\| = \sqrt{\widehat{I}_x^2 + \widehat{I}_y^2}$  in natural images is well approximated by a standard Rayleigh distribution outside of 0.

### 2.2 Discrete Junction Definition

A *junction* is defined as a discrete image structure  $J$  :  $\{\mathbf{p}, r, \{\theta_m\}_{m=1}^M\}$ , characterized by its center pixel  $\mathbf{p}$ , its scale  $r \in \mathbb{N}$  and a set of branch directions  $\{\theta_1, \dots, \theta_M\}$  around  $\mathbf{p}$ . The number  $M$  of branches in the junction is called the order of the junction : when  $M = 2, 3$  or  $4$ , we speak respectively of  $L, Y$  or  $X$ -junctions. The discrete set  $\mathcal{D}(r)$  of possible directions at a given scale  $r$  is defined as

$$\mathcal{D}(r) := \left\{ \frac{2\pi k}{K(r)} ; k \in \{1, \dots, K(r)\} \right\}. \tag{3}$$

In this definition,  $K(r)$  is a prescribed number of discrete directions on the circle. In the experimental part,  $K(r)$  will be chosen as  $K(r) = \lfloor 2\pi r \rfloor$ .<sup>2</sup>

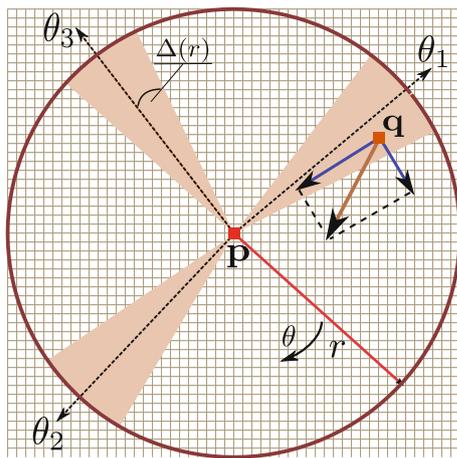
For a given scale  $r$  and a given direction  $\theta$  in  $\mathcal{D}(r)$ , we define the *branch* of direction  $\theta$  at  $\mathbf{p}$  as the disk sector

$$S_{\mathbf{p}}(r, \theta) := \left\{ \mathbf{q} \in \Omega ; \mathbf{q} \neq \mathbf{p}, \|\overrightarrow{\mathbf{p}\mathbf{q}}\| \leq r, d_{2\pi}(\alpha(\overrightarrow{\mathbf{p}\mathbf{q}}), \theta) \leq \Delta(r) \right\}, \tag{4}$$

where  $\Delta(r)$  is a precision parameter,  $\alpha(\overrightarrow{\mathbf{p}\mathbf{q}})$  is the angle of the vector  $\overrightarrow{\mathbf{p}\mathbf{q}}$  in  $[0, 2\pi]$  and where  $d_{2\pi}$  is the distance along the unit circle, defined as  $d_{2\pi}(\alpha, \beta) = \min(|\alpha - \beta|, 2\pi - |\alpha - \beta|)$ . We note  $J(r, \theta)$  the number of pixels in a sector of direction  $\theta$  at scale  $r$ . This size depends on the scale  $r$  but also slightly changes with the direction  $\theta$  since the image is defined on a discrete grid.

Finally, we require that two branches of a given junction do not intersect. It follows that the angle between two directions in a junction  $J$  must be larger than  $2\Delta(r)$ . As a consequence,

<sup>2</sup> For a circle of radius  $r$  centered at pixel  $(0,0)$ ,  $\lfloor 2\pi r \rfloor$  is a good approximation of the number of pixels  $(x, y)$  on the circle, in the sense that  $\text{round}(\sqrt{x^2 + y^2}) = r$ .



**Fig. 2** A junction with three branches. Each branch is represented by a disk sector  $S_p(r, \theta)$  with an angular aperture of  $2\Delta(r)$

the number of possible directions in a junction at scale  $r$  is smaller than  $\frac{\pi}{\Delta(r)}$ . An example of a junction with three branches is shown on Fig. 2.

### 2.3 Junction Strength

The strength of a junction will be defined from the strength of its branches. The strength of a branch is a measure of how well the corresponding angular sector agrees with the pixels it contains. A pixel  $\mathbf{q}$  contributes all the more to the strength as its direction  $\phi(\mathbf{q})$  is in agreement with the direction  $\alpha(\vec{\mathbf{p}\mathbf{q}})$  and as the normalized gradient  $\|\widetilde{\nabla}I(\mathbf{q})\|$  is large. Precisely, the strength of a branch is defined as

**Definition 1** (Strength of a branch) Let  $\mathbf{p}$  be a pixel,  $r$  a positive scale and  $\theta$  a direction in  $\mathcal{D}(r)$ . The strength of the branch  $S_p(r, \theta)$  is defined as the quantity

$$\omega_p(r, \theta) = \sum_{\mathbf{q} \in S_p(r, \theta)} \gamma_p(\mathbf{q}), \tag{5}$$

where

$$\gamma_p(\mathbf{q}) = \|\widetilde{\nabla}I(\mathbf{q})\| \cdot \max \left( |\cos(\phi(\mathbf{q}) - \alpha(\vec{\mathbf{p}\mathbf{q}}))| - |\sin(\phi(\mathbf{q}) - \alpha(\vec{\mathbf{p}\mathbf{q}}))|, 0 \right). \tag{6}$$

The *max* operator has been chosen so that the contribution of the pixel is equal to zero if the difference between  $\alpha$  and  $\phi$  is larger than  $\pi/4$ . Other choices could be made. The larger the strength of a branch, the more likely it is that the branch corresponds to an edge. Figure 3c shows the values of  $\gamma_p(\mathbf{q})$  for the image patch shown in Fig. 3a.

The strength of a junction is then derived from the strength of its branches as follows:

**Definition 2** (Strength of a junction) The strength of a junction  $J : \{\mathbf{p}, r, \{\theta_m\}_{m=1}^M\}$  is defined as

$$t(J) := \min_{m=1 \dots M} \omega_p(r, \theta_m). \tag{7}$$

Starting from this definition, a first naive algorithm for junction detection can be developed. The idea is to detect, for a fixed scale  $r$  and a given threshold  $t$ , all the junctions in  $I$  having a strength greater than  $t$ . In practice, testing all possible junctions for every point  $\mathbf{p}$  in  $\Omega$  is computationally heavy. Among all the potential branches at a given point  $\mathbf{p}$ , we restrict ourselves to the directions  $\theta$  in the discrete set  $\mathcal{D}(r)$  where the periodic function  $\omega_p(r, \cdot)$  reaches a local maximum. Moreover, in order for branches not to intersect, we impose that the local maximality holds over a length  $2\Delta(r)$ , that is  $\omega_p(r, \theta)$  is greater than  $\omega_p(r, \theta')$  for  $\theta - \Delta(r) \leq \theta' \leq \theta + \Delta(r)$ . The set of these local maxima can be computed quickly, for instance by using a non-maximum suppression (NMS) procedure (see Kitchen and Rosenfeld 1982; Neubeck and Van Gool 2006). In practice, if two local maxima are equal and located at a distance smaller than  $\Delta(r)$ , one of them is discarded.

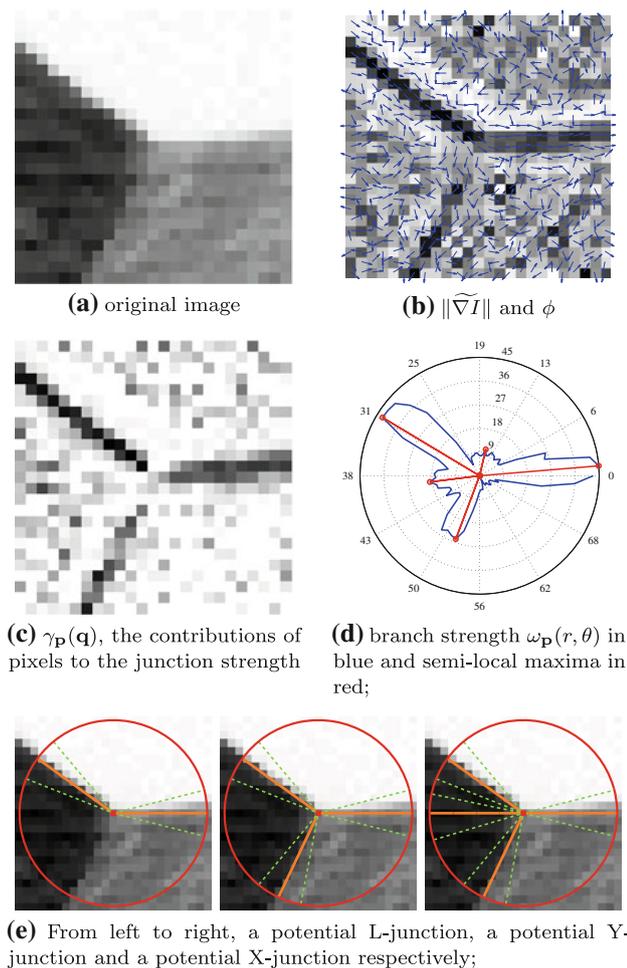
*The Case of L-junctions* In order not to detect all edge points as *L-junctions*, junctions for which  $M = 2$  and whose branches are opposite (that is, with two directions  $\theta_1$  and  $\theta_2$  such that  $d_{2\pi}(\theta_1, \theta_2 + \pi) \leq 2\Delta(r)$ ) are discarded.

The overall detection procedure is summarized in Algorithm 1.

**Algorithm 1** Junction detection in an image  $I$  at a given scale  $r$ .

- Input:** A discrete image  $I$ , an order  $M$ , a positive scale  $r$ , a number  $K$  of discrete directions, a precision  $\Delta$  and a threshold  $t$ .
- 1: compute  $\widetilde{\nabla}I$  at each pixel using Equation (1);
  - 2: set  $\mathcal{D} = \{\frac{2\pi k}{K}; k = 1, \dots, K\}$
  - 3: **for** each pixel  $\mathbf{p}$  **do**
  - 4:     (1) compute  $\omega_p(r, \theta)$  for each  $\theta$  in  $\mathcal{D}$ , using Equations (5) and (6);
  - 5:     (2) use a NMS procedure to only keep directions  $\theta$  such that  $\omega_p(r, \theta)$  is locally maximum on  $[\theta - \Delta, \theta + \Delta]$ ; call  $\Theta$  the set of these directions;
  - 6:     (3) find all the subsets  $\Theta'$  of size  $M$  in  $\Theta$  such that  $\min_{\theta \in \Theta'} \omega_p(r, \theta) \geq t$ ; for each of these subsets, mark  $J : \{\mathbf{p}, r, \Theta'\}$  as a junction if  $M \neq 2$ , or if  $M = 2$  and  $\Theta' = \{\theta_1, \theta_2\}$  with  $d_{2\pi}(\theta_1, \theta_2 + \pi) \leq 2\Delta$ .
  - 7: **end for**
  - 8: **return** The set of detected junctions.

The first parts of the procedure are illustrated in Fig. 3. In this example, the pixel  $\mathbf{p}$  is chosen as the center of Fig. 3a. Figure 3d shows in blue the values of  $\omega_p(r, \theta)$  when the angle  $\theta$  spans the periodic set  $\mathcal{D}(r)$  and shows in red the semi-local maxima kept after the NMS procedure (step (2) of the loop in Algorithm 1). Figure 3e represent respectively a candidate *L-junction*, a candidate *Y-junction* and a candidate *X-junction* at  $\mathbf{p}$ . The main drawback of this detection algorithm is that the threshold  $t$  on the junction strength remains the



**Fig. 3** Computation of junction candidates at scale  $r = 12$ , when  $\mathbf{p}$  is the center of the original image (a). The blue curve in (d) shows the strength  $\omega_{\mathbf{p}}(r, \theta)$  as a function of the direction  $\theta$  along the circle. The directions that remain after the NMS procedure are shown in red. For  $r = 12$ ,  $K(r) = 75$ , and  $2\Delta(r) = 0.289$  (see Sect. 4.1) (Color figure online)

same whatever the junction scale and order and whatever the image size. Setting such a threshold globally is not easy and can lead to over-detect at some scales and under-detect at others. The goal of the next section is to provide detection thresholds that adapt to the junction scale and order, as well as to the image size. For this purpose, we resort to an *a contrario* methodology.

### 3 An *a contrario* Approach for Junction Detection

The *a contrario* detection theory has been primarily proposed by (Desolneux et al. 2000). This methodology is inspired by geometric grouping laws governing low-level human vision, known as Gestalt laws (Kanizsa 1979), and states that meaningful structures in images are structures which are very unlikely under some hypothesis of ran-

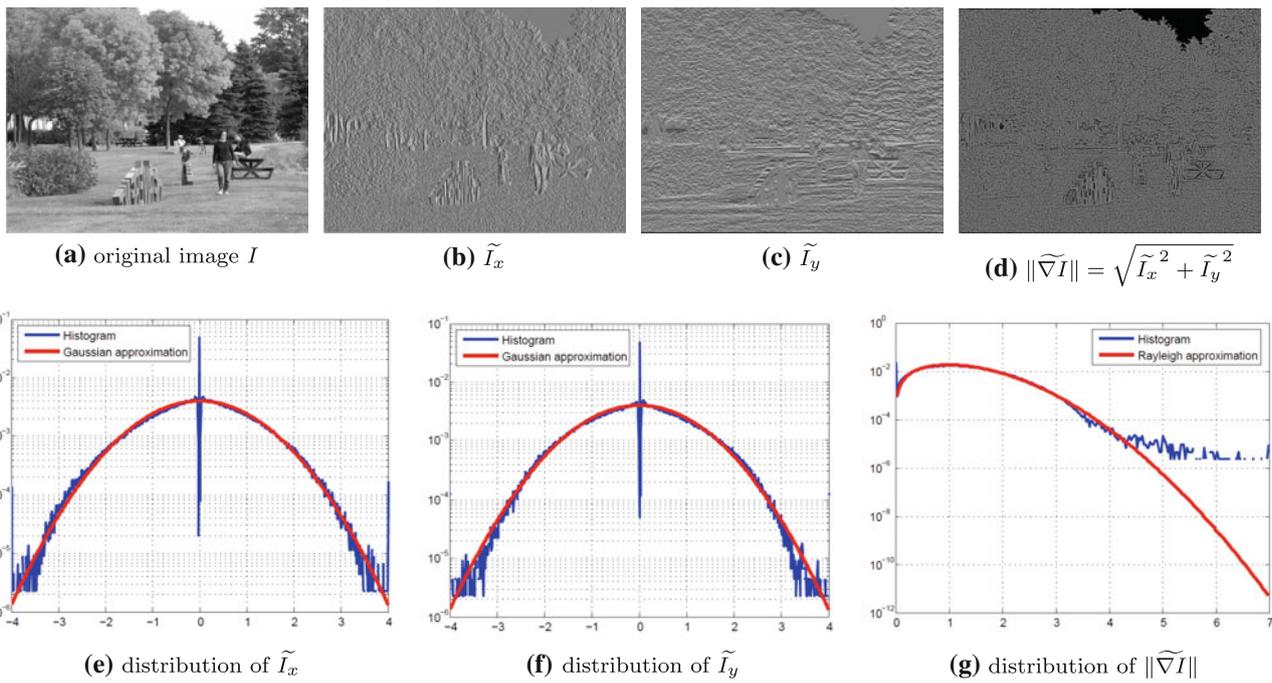
domness. The method has been extensively tested and successfully applied to various problems in image processing and computer vision, see e.g. (Cao 2003; Moisan and Stival 2004; Musé et al. 2006; von Gioi et al. 2010). A complete overview of these methods can be found in (Desolneux et al. 2008). In this section, this methodology is adapted to the detection of meaningful junctions in images.

#### 3.1 Null Hypothesis

The goal of the following sections is to set detection thresholds on junction strengths in such a way that no junction will be detected under a given null hypothesis  $\mathcal{H}_0$ . Let us precise what “null hypothesis” stands for here. For each pixel  $\mathbf{q}$  of  $\Omega$ , write  $\|\widetilde{\nabla \mathbf{I}}(\mathbf{q})\|$  and  $\phi_{\mathbf{I}}(\mathbf{q})$  the random variables corresponding to the value and orientation at this pixel. We say that these variables follow the null hypothesis  $\mathcal{H}_0$  if

1.  $\forall \mathbf{q} \in \Omega$ ,  $\|\widetilde{\nabla \mathbf{I}}(\mathbf{q})\|$  follows a Rayleigh distribution with parameter 1 ;
2.  $\forall \mathbf{q} \in \Omega$ ,  $\phi_{\mathbf{I}}(\mathbf{q})$  is uniformly distributed over  $[0, 2\pi]$  ;
3. The family  $\{\|\widetilde{\nabla \mathbf{I}}(\mathbf{q})\|, \phi_{\mathbf{I}}(\mathbf{q})\}_{\mathbf{q} \in \Omega}$  is made of independent random variables.

Let us comment on the first assumption. In Ruderman and Bialek (1994), observe that if we normalize the logarithm of the intensity of an arbitrary natural image by its local mean and standard deviation in a neighborhood  $\mathcal{N}_p$  around each pixel, “the histogram of pixel values has Gaussian tails, and the distribution of gradients in the ‘variance-normalized’ image is almost exactly the Rayleigh distribution”. A similar behavior may be observed on our case on the modified derivatives  $\widehat{I}_x$  and  $\widehat{I}_y$ : except on a small neighborhood around 0, their distribution is well approximated by a Gaussian distribution. This is illustrated for a particular image  $I$  in Fig. 4. In this figure, the empirical distributions of  $\widehat{I}_x$  and  $\widehat{I}_y$  are drawn in blue, and the best Gaussian fits are drawn in red. If we except a peak around 0, the fit is very good (the poor approximation for large values is a numerical effect: probabilities smaller than  $\frac{1}{|\Omega|}$  are not attainable empirically). It follows from this observation that  $\widetilde{I}_x$  and  $\widetilde{I}_y$  are well approximated by the standard normal distribution (that is, a Gaussian distribution with zero mean and unit variance) and the norm  $\|\widetilde{\nabla \mathbf{I}}\|$  in natural images approximately follows a central chi-distribution with 2 degrees of freedom (also known as the Rayleigh distribution of parameter 1). This is confirmed experimentally by the example shown in Fig. 4. Moreover, we performed a small scale experiment whose results are shown on Fig. 5a. In blue are displayed the empirical distributions of  $\|\widetilde{\nabla \mathbf{I}}\|$  for 50 different natural images, as well as the density of the Rayleigh distribution of parameter 1 (in red). Again, proba-



**Fig. 4** Approximations of the distributions of  $\tilde{I}_x$ ,  $\tilde{I}_y$  and  $\|\tilde{\nabla}I\|$  (given by Eqs. (1) and (2)). The blue curves represent the empirical histograms and the red curves are the corresponding Gaussian and Rayleigh distributions. Plots are displayed in a semi-log scale. The poor approximation

bilities smaller than  $\frac{1}{|\Omega|}$  cannot be reached empirically, which explains the poor fit for large values. Eventually, observe that the Gaussian approximation used for the normalized gradients  $\tilde{I}_x$  and  $\tilde{I}_y$  would not be adequate for the classical gradients  $I_x$  and  $I_y$ , whose distributions are highly non-Gaussian and accurately modeled by generalized Gaussian distributions.

### 3.2 Distribution of $\mathbf{t}(J)$ Under $\mathcal{H}_0$

Let  $J : \{\mathbf{p}, r, \{\theta_m\}_{m=1}^M\}$  be a junction in  $I$  and assume that the normalized gradients and directions of  $I$  follow the null hypothesis  $\mathcal{H}_0$ . Then the strengths  $\omega_{\mathbf{p}}(r, \theta_m)$  of the different branches  $S_{\mathbf{p}}(r, \theta_m)$  are independent random variables (since branches do not intersect). Recall also that the strength of a junction is the minimum of the strengths of its branches. Thus, if we note  $\mathbf{t}(J)$  the random variable measuring the strength of  $J$ ,

$$\begin{aligned} \mathbb{P}_{\mathcal{H}_0}[\mathbf{t}(J) \geq t] &= \mathbb{P}_{\mathcal{H}_0}[\forall m, \omega_{\mathbf{p}}(r, \theta_m) \geq t] \\ &= \prod_{m=1}^M \mathbb{P}_{\mathcal{H}_0}[\omega_{\mathbf{p}}(r, \theta_m) \geq t]. \end{aligned} \tag{8}$$

Now, the strength of each branch,  $\omega_{\mathbf{p}}(r, \theta_m)$ , is itself a sum (over the angular sector  $S_{\mathbf{p}}(r, \theta_m)$ ) of i.i.d. random variables  $\gamma_{\mathbf{p}}(\mathbf{q})$ . For two given points  $\mathbf{p}$  and  $\mathbf{q}$  in  $\Omega$ , the direction  $\alpha(\overrightarrow{\mathbf{p}\mathbf{q}})$  is a constant in  $[0, 2\pi]$ . This implies that under the hypoth-

esis  $\mathcal{H}_0$ , the random angle  $\phi_{\mathbf{1}}(\mathbf{q}) - \alpha(\overrightarrow{\mathbf{p}\mathbf{q}})$  is still uniformly distributed on  $[0, 2\pi]$ . As a consequence, each  $\gamma_{\mathbf{p}}(\mathbf{q})$  can be written as a product  $X \cdot \max(|\cos \theta| - |\sin \theta|, 0)$ , where  $X$  and  $\theta$  are independent,  $X$  follows a Rayleigh distribution and  $\theta$  is uniformly distributed on  $[0, \pi]$ . Finally, it can be shown that under the hypothesis  $\mathcal{H}_0$ , each random variable  $\gamma_{\mathbf{p}}(\mathbf{q})$  follows the distribution (see Appendix 2 for a proof)

$$\mu = \frac{1}{2} (\delta_0 + R \cdot \mathcal{L}), \tag{9}$$

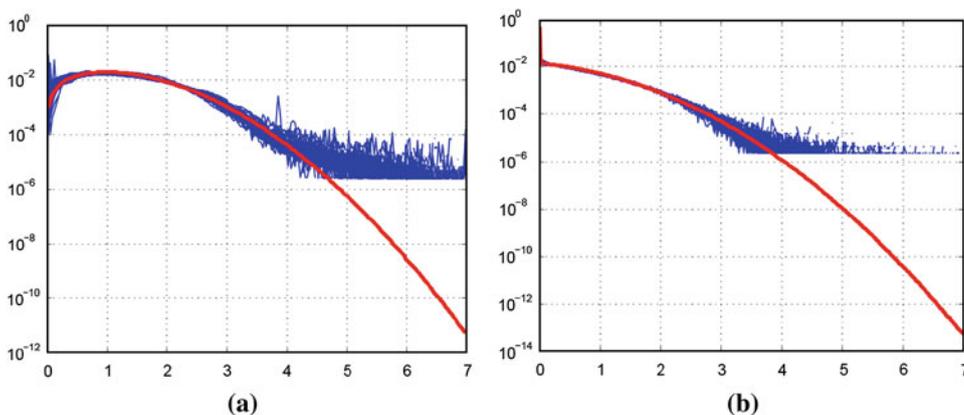
where  $\delta_0$  is a Dirac measure centered at 0 and  $\mathcal{L}$  is the Lebesgue measure on the real line. The function  $R$  can be written

$$R(z) = \frac{2}{\sqrt{\pi}} e^{-\frac{z^2}{4}} \cdot \operatorname{erfc}\left(\frac{z}{2}\right) \mathbb{1}_{z \geq 0}, \tag{10}$$

with  $\operatorname{erfc}$  the complementary error function,  $\operatorname{erfc}(z) = \frac{2}{\sqrt{\pi}} \int_z^\infty e^{-s^2} ds$ .

The empirical distribution of  $\gamma_{\mathbf{p}}(\mathbf{q})$  on 50 natural images is displayed in Fig. 5b, along with its theoretical distribution  $\mu$ , showing very good fit to the model. It remains to compute the distribution (under  $\mathcal{H}_0$ ) of the strength of a branch. Recall that under  $\mathcal{H}_0$ , branches are mutually independent. Therefore, the law of the strength of a branch  $\omega_{\mathbf{p}}(r, \theta_m)$  (that is, of the sum of pixel contributions given by Formula (5)) is obtained by convolving  $J(r, \theta_m)$  copies of the distribution  $\mu$ , where

**Fig. 5** Distributions of  $\|\widetilde{\nabla}I\|$  (blue curves in (a)) and  $\gamma_{\mathbf{p}}(\mathbf{q})$  (blue curves in (b)) for 50 different natural images. *Rayleigh(1)* density (red curve in (a)) and density of the corresponding strength distribution  $\mu$  (red curve in (b)) are displayed for comparison (Color figure online)



$J(r, \theta_m)$  is the size of a sector of orientation  $\theta_m$  at scale  $r$ . Finally,

**Proposition 1** Let  $J : \{\mathbf{p}, r, \{\theta_m\}_{m=1}^M\}$  be a junction and suppose that the hypothesis  $\mathcal{H}_0$  is satisfied, then the probability that the random variable  $\mathbf{t}(j)$  is larger than a given threshold  $t$  is

$$F_j(t) := \mathbb{P}_{\mathcal{H}_0}[\mathbf{t}(j) \geq t] = \prod_{m=1}^M \int_t^{+\infty} d \left( \begin{matrix} J(r, \theta_m) \\ \star \\ j=1 \end{matrix} \mu \right), \quad (11)$$

where  $J(r, \theta_m)$  is the size of a sector of orientation  $\theta_m$  at scale  $r$  and where  $\star$  denotes the convolution operator.

### 3.3 Meaningful Junctions

Thanks to the previous computations, we are now in a position to automatically fix detection thresholds on junction strengths. Indeed, thresholds are set so that the average number of false detections under the null hypothesis is controlled. This is obtained by thresholding the probability (11) and by taking into account the number of possible discrete junctions in the image.

#### 3.3.1 Number of Tests

In this paragraph, we assume that the order  $M$  is fixed, and we call  $\mathcal{J}(M)$  the set of all possible (possibly overlapping) junctions of order  $M$  in the discrete image  $I$ . The size of  $\mathcal{J}(M)$  depends on several parameters: the minimum and maximum authorized scales  $r_{min}$  and  $r_{max}$ , the size  $N$  (number of pixels) of  $I$  and the precision  $\Delta(r)$  of branches at each scale  $r$ . The practical setting of these parameters will be discussed in Sect. 4.1.

At a given location  $\mathbf{p}$ , once the first branch is chosen among the  $K(r)$  possible directions, the second direction must be chosen in such a way that the two branches of width  $2\Delta(r)$  do not intersect (in the sense that their sectors do not overlap), which means that only  $K(r)(1 - 2\frac{\Delta(r)}{\pi})$  directions are

authorized. Therefore, at each location  $\mathbf{p} \in \Omega$ , and for a given scale  $r$ , the number of possible junctions of order  $M$  is always smaller than

$$\frac{1}{M!} \prod_{m=1}^M K(r) \left( 1 - 2(m-1) \frac{\Delta(r)}{\pi} \right).$$

It follows that the size of the set  $\mathcal{J}(M)$  is upper bounded by

$$\#\mathcal{J}(M) = \frac{N}{M!} \cdot \sum_{r=r_{min}}^{r_{max}} \prod_{m=1}^M K(r) \left( 1 - 2(m-1) \frac{\Delta(r)}{\pi} \right). \quad (12)$$

#### 3.3.2 $\epsilon$ -meaningful Junctions

The next definition and the following proposition explain how to fix thresholds on junctions strengths in order to control the average number of false detections.

**Definition 3** ( $\epsilon$ -meaningful junction) Let  $I$  be a discrete image. For  $\epsilon > 0$ , a junction  $j$  of order  $M$  and at scale  $r$  is said to be  $\epsilon$ -meaningful if its number of false alarm (NFA) under the hypothesis  $\mathcal{H}_0$  satisfies

$$\text{NFA}(j) := \#\mathcal{J}(M) \cdot F_j(t(j)) \leq \epsilon, \quad (13)$$

The quantity  $\text{NFA}(j)$  is a measure of the meaningfulness of the junction: the smaller it is, the more meaningful the junction  $j$ . A junction of order  $M$  and scale  $r$  is detected as  $\epsilon$ -meaningful in  $I$  if its strength  $t(j)$  is larger than the threshold

$$t(r, \epsilon) := \min \left\{ t; F_j(t) \leq \frac{\epsilon}{\#\mathcal{J}(M)} \right\}. \quad (14)$$

Notice that for a fixed  $\epsilon$ , this formula yields a different threshold on  $t(j)$  for each value of the scale  $r$ . The value  $\epsilon$  is easy to interpret: it corresponds to an expected number of false detections under the hypothesis  $\mathcal{H}_0$ . Indeed,

**Proposition 2** Assume that the null hypothesis  $\mathcal{H}_0$  is satisfied in image  $I$ . Let  $M$  be a positive integer. The expectation of the number of  $\epsilon$ -meaningful junctions of order  $M$  in  $I$  is smaller than  $\epsilon$ .

This proposition is an obvious consequence of the fact that if  $X$  is a random variable and if we define  $F(t) = \mathbb{P}[X \geq t]$ , then for all  $\beta$  in  $[0, 1]$ ,  $\mathbb{P}[F(X) \leq \beta] \leq \beta$ .

Observe that in the definition of the NFA, the correcting factor  $\#\mathcal{J}(M)$  is independent of the scale  $r$ . We could have used different correcting factors, depending on the junction scale  $r$ , in order to favor some particular scales in the detection, and still have a result similar to Proposition 2.

As can be seen from Formula (13), the quantity  $\text{NFA}(J)$  is computed from the number of tests (12) and the probability (11). Its practical computation is detailed in Appendix 1.

### 3.4 Junctions or Nearby Edges?

In images, some structures may appear as junctions at large scales without being detected as such when the scale is small. For instance, when three straight edges have close endpoints and are supported by concurring lines, a  $Y$ -junction may be detected even if the actual edges do not physically meet, when the possible junction contains a gap at its center. Deciding between a real junction and interacting edges is actually far from trivial. Psychophysical experiments (Rubin 2001) suggest that human may find junctions although there is a small gap at the center. In practice, we got qualitatively better results when removing junctions with large gaps at their center. This restriction is imposed by computing for each  $\epsilon$ -meaningful junction  $J : \{\mathbf{p}, r, \{\theta_m\}_{m \in \{1, \dots, M\}}\}$  a minimum scale of detection  $r_d[J]$  and by removing all  $\epsilon$ -meaningful junctions such that  $r_d[J]$  is strictly larger than a fixed threshold  $r_{gap}$ . This minimum scale of detection is defined as the smallest scale  $r_d$  such that there exists  $\epsilon$ -meaningful junctions of order  $M$  centered at  $\mathbf{p}$  for all scales between  $r_d$  and  $r$  (see Fig. 6),

$$r_d[J] = \min \left\{ r' \leq r; \forall s \in [r', r], \exists J' : \{\mathbf{p}, s, \{\theta'_m\}_{m=1}^M\} \right. \\ \left. \text{s.t. } J' \text{ is } \epsilon\text{-meaningful.} \right\}. \quad (15)$$

In practice and for all experiments in this paper, the threshold  $r_{gap}$  is chosen as  $r_{gap} = 12$ . In order to be fully scale invariant, this threshold could be replaced by a value proportional to the junction scale  $r$ .

### 3.5 Redundant Detections and Maximality

#### 3.5.1 Redundancy: Scale and Location

As it is common when analyzing geometrical structures in images, junctions are usually detected in a redundant way.

A single structure in the image may yield many detections. First, junctions are detected over a range of scale. For a single ideal junction in the image, meaningful junctions will be detected for scales both smaller and larger than the one of the underlying structure, see Fig. 7, middle, where several junctions having the same center but different scales, and corresponding to the same ideal  $Y$ -junctions, are displayed. Second, several junctions with slightly different locations are detected for a single underlying structure. This is all the more strong as there is blur in the image. An example of such redundant detections is displayed in Fig. 7, right. Moreover, both type of over-detections (multiple scales and multiple locations) are usually combined in images. These redundancies are addressed in the next paragraph thanks to an exclusion principle.

#### 3.5.2 Maximal Junctions

In order to choose the right representative among all redundant detections, we use an exclusion principle, called maximality. We assign to each junction  $J$  a neighborhood  $\mathcal{N}'_J$ . For a given order  $M$ , we only keep the junctions not containing any more meaningful junction in its neighborhood. That is, we only keep junctions that are maximal in the following sense.

**Definition 4** (Maximal  $\epsilon$ -meaningful junction of order  $M$ ) A junction  $J : \{\mathbf{p}, r, \{\theta_m\}_{m=1}^M\}$  is said to be a maximal  $\epsilon$ -meaningful junction of order  $M$  if  $J$  is  $\epsilon$ -meaningful and if  $\text{NFA}(J) \leq \text{NFA}(J')$  for any junction  $J' : \{\mathbf{p}', r', \{\theta'_m\}_{m=1}^M\}$ , with  $\mathbf{p}' \in \mathcal{N}'_J$ .

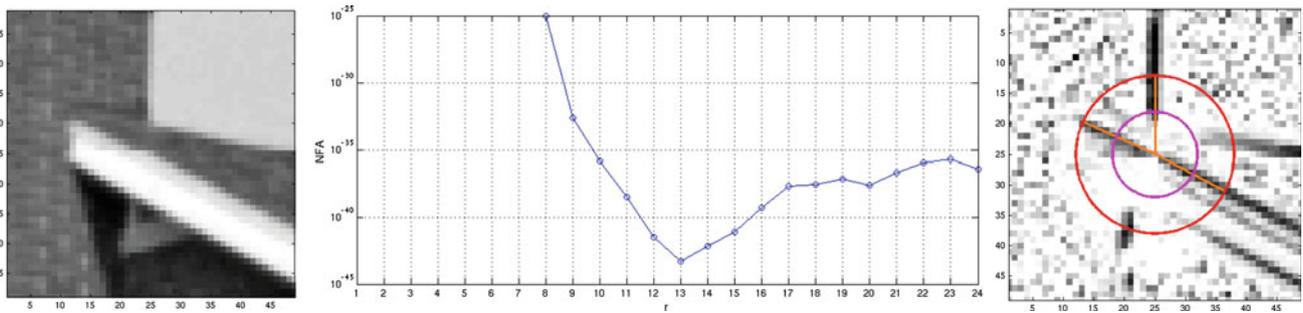
Observe that in this definition the use of the NFA is the keypoint. Indeed, it permits to compare structures at different scales. Using the strengths  $t(J)$  to carry out this comparison would require a well chosen normalization depending on the scale.

In order to select maximal meaningful junctions, the most meaningful junction is first considered. All junctions having it as a neighbor are then removed. Then we proceed to the next most meaningful junction and iterate the same procedure until all junctions have been treated.

A result of these selection rules is illustrated in Fig. 8 for  $M = 3$ . In practice, the spatial neighborhood  $\mathcal{N}'_p$  used for maximality is chosen as a disk centered at  $\mathbf{p}$ , with a radius  $r_d[J]$ , the minimum scale of detection of  $J$  as defined by Formula (15).

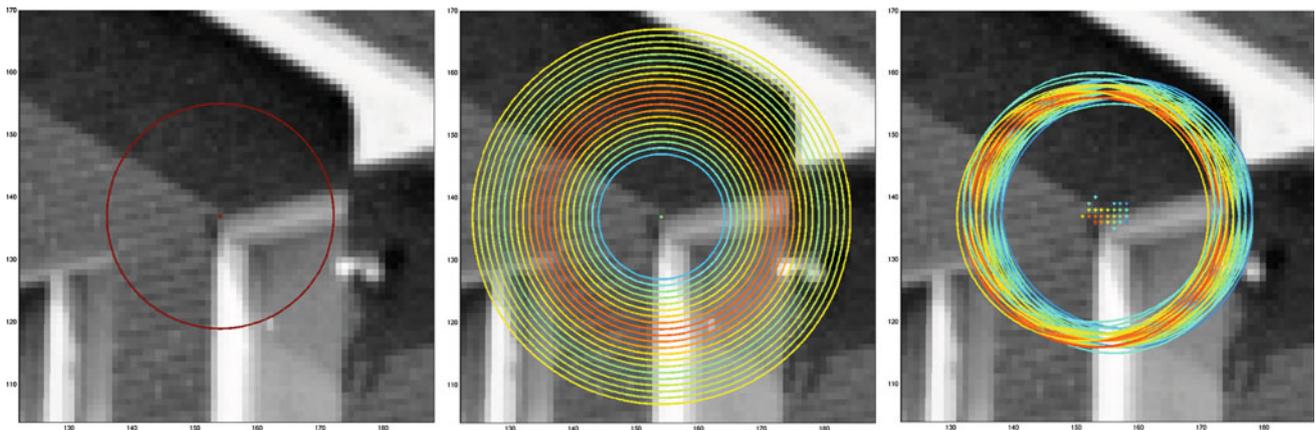
#### 3.5.3 Masking and Junction Order

When a  $Y$ -junction is perceived in a image, the underlying  $L$ -junctions are usually not perceived and we decided not to detect them. This masking phenomenon (for a given junction, no junction made of a subset of its branches should



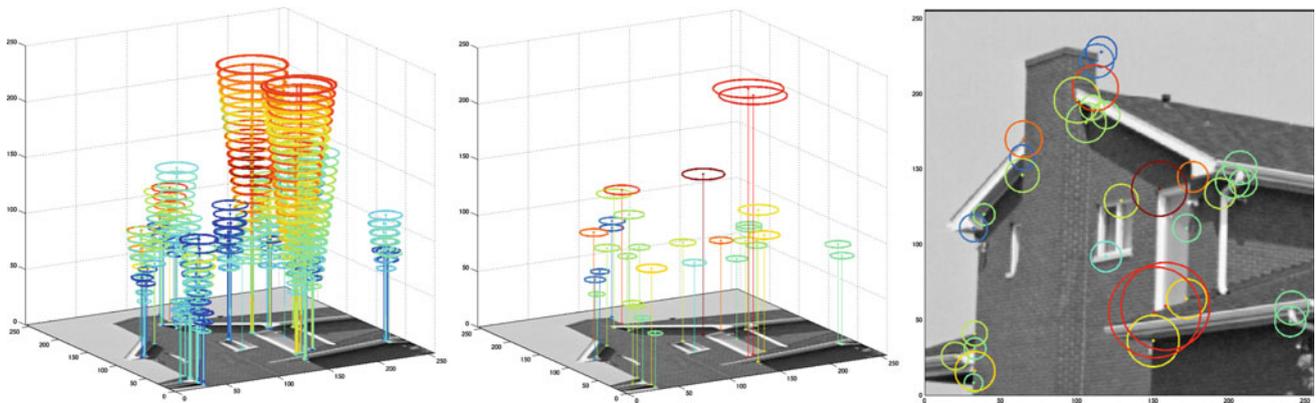
**Fig. 6** Scale of a junction. *Left* image  $I$ . The point  $\mathbf{p}$  is chosen as the center of  $I$ . *Middle* for each scale  $r$  (in abscissa) is displayed the smallest NFA( $J$ ) observed for a junction  $J$  of order 3 centered at  $\mathbf{p}$ . *Right* the

smallest scale of detection  $r_d = 8$  in magenta and the scale corresponding to the smallest NFA in red (found for  $r = 13$ ), represented on top of the strength  $\gamma_{\mathbf{p}}(\mathbf{q})$  (Color figure online)



**Fig. 7** Redundancy of junction detection. For the sake of clarity, each junction is represented by a *circle* and its center, the radius of the circle is equal to the scale of the junction, and the *color* of the circle depends on the NFA value (*red* corresponds to small values, i.e. very meaningful junctions and *blue* corresponds to high values). *Left* a junction

$J : \{\mathbf{p}, r, \{\theta_m\}_{m \in \{1, \dots, 3\}}\}$ , with  $r = 20$ . *Mid* all junctions detected at the same point  $\mathbf{p}$ , with different scales. *Right* all  $Y$ -junctions detected in the neighborhood of  $\mathbf{p}$ , with the same directions and scale (Color figure online)



**Fig. 8** Maximal meaningful junctions of order 3. Each  $Y$ -junction is represented by a *circle* and its center. The radius of the circle indicates the junction scale  $r$ , and the *color* of the circle corresponds to the NFA of the junction (the cooler the color, the larger the NFA). *Left* all

$\epsilon$ -meaningful  $Y$ -junctions, with  $\epsilon = 1$ . *Mid* the maximal meaningful  $Y$ -junctions. *Right* the maximal meaningful  $Y$ -junctions displayed over the image (Color figure online)

be detected) is easily implemented by the following second exclusion principle: locally, only the more complex junction (the one with the largest order) is kept.

**Definition 5** (*Maximal  $\epsilon$ -meaningful junction*) A junction  $J : \{\mathbf{p}, r, \{\theta_m\}_{m=1}^M\}$  is said to be a maximal  $\epsilon$ -meaningful junction if  $J$  is a maximal  $\epsilon$ -meaningful junction of order  $M$

and if there is no maximal  $\epsilon$ -meaningful junction of order  $M'$  located at  $\mathbf{p}'$  with  $M' > M$  and  $p' \in \mathcal{N}'_j$ .

### 3.6 The Three Algorithmic Steps for Junction Detection

The different steps of the junction detection procedure are summarized in Algorithms 2 (the *a contrario* detection), 3 (maximality of order  $M$ ) and 4 (maximality). The complete algorithm pipeline includes a speed-up step and an optional precision refinement and will be described in Sect. 4 and summarized in Algorithm 7.

---

#### Algorithm 2 *A contrario* junction detection

---

**Input:** An image  $I$ , a maximal order  $M'$  and parameters  $\epsilon, r_{min}, r_{max}, r_{gap}$ .

**Output:** A list of junctions  $Jlist$  and the corresponding list of minimum detection scales  $r_d[Jlist]$ .

```

1: Compute  $\widetilde{\nabla}I$  at each pixel using Equation (1);
2: For each value of  $M$  between 2 and  $M'$ , compute  $\#\mathcal{J}(M)$  by using Equation (12);
3: for each pixel  $\mathbf{p}$  do
4:   Let  $\rho$  be a  $M' \times r_{max}$  matrix (used to record the values  $r_d$ ), and fill it with zeros.
5:   for  $r = 1$  to  $r_{max}$  do
6:     Compute the branch strength  $\omega_{\mathbf{p}}(r, \theta)$  for each direction  $\theta$  in  $\mathcal{D}(r)$ , using Equations (5) and (6).
7:     Use a NMS procedure to only keep semi-local maxima of  $\omega_{\mathbf{p}}(r, \cdot)$ ; call  $\Theta = \{\theta_m\}_{m=1}^{\#\Theta}$  the set of these directions.
8:     Sort the vector  $\omega_{\mathbf{p}}(r, \Theta)$  in a descending order ( $index, v$ )  $\leftarrow$   $\text{sort}(\omega_{\mathbf{p}}(r, \Theta))$ .
9:     Suppression of alignments:
       if  $M = 2$  and  $d_{2\pi}(\theta_{index(1)}, \theta_{index(2)} + \pi) \leq 2\Delta(r)$ 
          $\theta_{index(2)} = \theta_{index(3)}$ 
       end if
10:    for  $M = 2$  to  $M'$  do
11:      Define the junction  $J : \{\mathbf{p}, r, \{\theta_m\}_{m=index(1)}^{index(M)}\}$ , with strength  $t(J) = v(index(M))$ 
12:      Compute  $\log \text{NFA}(J)$  by using Equation (13)
13:      if  $\log \text{NFA}(J) \leq \log \epsilon$  then
14:        if  $r \geq 2$  and  $\rho(M, r - 1) \neq 0$  then
15:           $\rho(M, r) = \rho(M, r - 1)$  and  $r_d[J] = \rho(M, r - 1)$ 
16:        else
17:           $\rho(M, r) = r$  and  $r_d[J] = r$ 
18:        end if
19:        if  $r_d[J] < r_{gap}$  then
20:          Accept the junction proposal  $J$ :  $Jlist \leftarrow J$ 
21:        end if
22:      end if
23:    end for
24:  end for
25: end for

```

---

Algorithm 2 does not include the two maximality steps (described in Algorithms 3 and 4). Notice however that line 11 of Algorithm 2 is a first step towards maximality, since only the best junction of a given order  $M$  is tested at each point. This permits to speed-up the algorithm by excluding junctions that will obviously not be maximal. If we wish

to compute all  $\epsilon$ -meaningful junctions in an image, and not only maximal junctions, lines 11 to 22 should be replaced by Algorithm 5.

---

#### Algorithm 3 Maximality for junctions of order $M$

---

**Input:**  $Jlist$ : the list of  $L$   $\epsilon$ -meaningful junctions of order  $M$  detected in  $I$  and the list  $r_d[Jlist]$ .

```

1: Sort  $Jlist$  in a descending order according to the NFA of junctions:
    $Jlist \leftarrow \text{Sort}(Jlist, \text{NFA})$ 
2: Remove non-maximal meaningful junctions
3: for  $j = 1$  to  $L$  do
4:    $J_1 \leftarrow$  location of junction  $Jlist[j]$ ;
5:   for  $k = j + 1$  to  $L$  do
6:      $J_2 \leftarrow$  location of junction  $Jlist[k]$ 
7:     if  $\|\mathbf{p}_{J_1} - \mathbf{p}_{J_2}\|_2 \leq r_d[J_1]$  then
8:       remove  $Jlist[j]$ , then break;
9:     end if
10:  end for
11: end for

```

---



---

#### Algorithm 4 Maximality

---

**Input:** A maximum order  $M'$ . The list  $Jlist$  of maximal  $\epsilon$ -meaningful junctions of all orders smaller than  $M'$  in  $I$  and the list  $r_d[Jlist]$ .  $L$  is the size of  $Jlist$ .

```

1: for  $j = 1$  to  $L$  do
2:    $J_1 \leftarrow$  location of junction  $Jlist[j]$ ;
3:   for  $k = 1$  to  $L$  do
4:      $J_2 \leftarrow$  location of junction  $Jlist[k]$ 
5:     if  $\|\mathbf{p}_{J_1} - \mathbf{p}_{J_2}\|_2 \leq r_d[J_1]$  and  $M_{J_1} < M_{J_2}$  then
6:       remove  $Jlist[j]$ , then break;
7:     end if
8:   end for
9: end for

```

---



---

#### Algorithm 5 Computing all $\epsilon$ -meaningful junctions

---

```

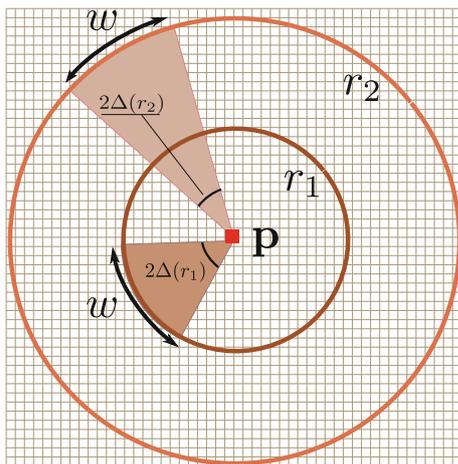
for each junction  $J : \{\mathbf{p}, r, \{\theta_m\}_{k=1}^M\}$ , with  $index(m_1) < \dots <$ 
 $index(m_M)$  and  $t(J) = v(index(m_M))$  do
  Compute  $\log \text{NFA}(J)$  by using Equation (13);
  if  $\log \text{NFA}(J) \leq \log \epsilon$  then
    Accept the junction proposal  $J$ :  $Jlist \leftarrow J$ .
  end if
end for

```

---

## 4 Implementation

The goal of this section is to provide all necessary information for the practical implementation of junction detection. First, the setting of parameters is addressed in Sect. 4.1. An optional refinement step to improve the accuracy of the detected branch directions is described in Sect. 4.2. A pre-selection of the junction candidates for speeding up the method is detailed in Sect. 4.3. Eventually, the complete detection algorithm pipeline is given in Sect. 4.4. A practical issue regarding the numerical computation of the NFA is also given in Appendix 1.



**Fig. 9**  $\Delta(r) = \omega/r$  for two distinct  $r$  values. Observe that the length of the arc defined by a sector,  $w$ , is a constant and does not depend on  $r$

### 4.1 Parameter Choices

Recall that in the definition of discrete junctions,  $K(r)$ , the number of possible directions for junction branches as defined by Formula (3), was chosen as  $K(r) = \lfloor 2\pi r \rfloor$ , in order to have a precision of roughly one pixel along the circle of radius  $r$ . The choice of the precision  $\Delta(r)$  relies on similar considerations. Visual experiments show that the perceived precision of an angle between two crossing segments in an image is better for long segments than short ones. Now, recall that  $2\Delta(r)$  is the angle of a branch (or sector) in a junction at scale  $r$ . We consider that the length of the arc defined by this sector should be a constant  $w$  and should not depend on  $r$ . This length is exactly  $2\Delta(r) \times r$ , which implies that  $\Delta(r)$  should be chosen as inversely proportional to  $r$ . In practice, we choose  $w = 5$ . Thus

$$2\Delta(r) = \frac{w}{r} = \frac{5}{r}. \tag{16}$$

Figure 9 illustrates the corresponding angular sectors for two different scales.

It follows that for a given order  $M$ , the number of tests  $\#\mathcal{J}(M)$  can be computed as

$$\#\mathcal{J}(M) = \frac{N}{M!} \sum_{r=r_{min}}^{r_{max}} \prod_{m=1}^M \lfloor 2\pi r \rfloor \left( 1 - \frac{5(m-1)}{r\pi} \right), \tag{17}$$

where  $N$  is the total number of pixels in the image. In the experimental section, the maximum order of junctions will be  $M = 4$  and the smallest possible radius is  $r_{min} = 1$  for all experiments. The maximal scale  $r_{max}$  is chosen as 5% of the diagonal of the natural image. For synthetic images,  $r_{max}$  can be set with a prior. The threshold on the minimum scale of detection is set to  $r_{gap} = 12$  for all experiments in this paper.

### 4.2 Optional Direction Refinement

Since directions in a junction are bisectors of angular sectors (see Eq. (4) for the definition of  $S_p(r, \theta)$ ) and since the set of possible directions is discrete, it may happen that the directions of some branches in a detected junction remain slightly imprecise. In the following, we describe a simple refinement in the computation of junction directions.

For a branch of direction  $\theta$  centered at  $\mathbf{p}$ , a refined direction  $\hat{\theta}$  is computed as follows

$$\hat{\theta} = \arctan \frac{\mathcal{O}_y}{\mathcal{O}_x}, \tag{18}$$

with  $\mathcal{O}_x = \sum_{\mathbf{q} \in S_p(r, \theta)} \gamma_p(\mathbf{q}) \cos \psi_{\mathbf{q}};$

$$\mathcal{O}_y = \sum_{\mathbf{q} \in S_p(r, \theta)} \gamma_p(\mathbf{q}) \sin \psi_{\mathbf{q}};$$

and  $\psi_{\mathbf{q}} = \begin{cases} \phi_{\mathbf{q}} & \text{if } d_{2\pi}(\phi_{\mathbf{q}}, \theta) < \frac{\pi}{2} \\ \phi_{\mathbf{q}} + \pi & \text{otherwise.} \end{cases} \tag{19}$

Notice that after this refinement, two sectors of a given junction may overlap. If this happens, the detected junction is removed. The refinement process is described in Algorithm 6.

---

#### Algorithm 6 Direction refinement

---

```

Input: A junction  $J : (\mathbf{p}, r, \{\theta_k\}_{k=1}^M)$ 
1: (1) Refinement of junction branches:
2: for  $k = 1$  to  $M$  do
3:    $\mathcal{O}_x = 0, \mathcal{O}_y = 0;$ 
4:   for  $\mathbf{q} \in S_p(r, \theta_k)$  do
5:     Compute  $\gamma_p(\mathbf{q})$  using Equation (6);
6:     Compute  $\psi_{\mathbf{q}}$  using Equation (19);
7:   end for
8:    $\mathcal{O}_x \leftarrow \mathcal{O}_x + \sum_{\mathbf{q} \in S_{B_k}} \gamma_p(\mathbf{q}) \cos \psi_{\mathbf{q}};$ 
9:    $\mathcal{O}_y \leftarrow \mathcal{O}_y + \sum_{\mathbf{q} \in S_{B_k}} \gamma_p(\mathbf{q}) \sin \psi_{\mathbf{q}};$ 
10:  Update the branch direction  $\theta_k = \arctan \frac{\mathcal{O}_y}{\mathcal{O}_x}.$ 
11: end for
12: (2) Check that branches are still disjoint.
13: for  $m = 1$  to  $M$  do
14:   for  $k = m + 1$  to  $M$  do
15:     if  $d_{2\pi}(\theta_m, \theta_k) < \Delta(r)$  then
16:       Remove  $J$ , then break;
17:     end if
18:   end for
19: end for

```

---

Let us stress that an extra refinement on the spatial position of the junction could be added to the algorithm. However we decided not to make any specific assumptions about the image structure. Therefore, our method does not yield sub-pixel positions, as may be achieved by e.g. the detector from (Deschenes et al. 2004) or the SFOP detector from (Förstner et al. 2009).

**Fig. 10** Junction candidates using LSD. *Left* the input image; *mid* all line segments detected with  $\lambda = 10^4$ ; *right* all junction candidates



**Table 2** Number of detections and running time when using or not the LSD pre-selection

Image			Using LSD candidates		Using all candidates	
Class	Name	Size	Num (L/T/X)	Time (s)	Num (L/T/X)	Time (s)
Geometry+texture	House	256 × 256	28/21/3	2.17	29/21/3	8.16
	Lena	256 × 256	79/44/12	3.90	81/44/12	9.77
	Window	768 × 576	120/224/51	10.36	122/230/52	64.72
Texture	Autumn	576 × 768	52/25/8	38.52	54/27/8	62.07
	Park	576 × 768	97/98/15	37.20	100/100/15	62.18
	Branches	536 × 819	78/100/18	35.97	80/101/19	62.08
Geometry	Geometric	655 × 518	27/17/2	1.48	28/17/2	44.19
	Composition	413 × 300	3/60/36	1.94	3/62/37	17.46
	Suprematism	400 × 640	40/33/5	1.57	42/34/5	34.66

The comparisons are implemented on three types of images: images with a strong geometrical content, images containing mostly textures and photographs of abstract paintings. Num (L/T/X) stands for the numbers of detected L/T/X junctions respectively. The experiments are implemented on a computer with a 2.0GHz Intel Core 2 Duo Processor and 2GB RAM. The maximum scale in this testing is set to 15

### 4.3 Speed Up

In order to speed-up the algorithm, and following (Parida et al. 1998; Cazorla et al. 2002; Cazorla and Escolano 2003; Sinzinger 2008), we propose to apply a pre-processing step to select potential junction candidates.

We take advantage of a fast segment detector, the Line Segment Detector (LSD) as introduced in (von Gioi et al. 2010),<sup>3</sup> whose complexity is linear in the size of the image. In order to make sure that we won't miss some junction candidates, the detection threshold  $\lambda$  of the LSD is set to a large value. The value  $\lambda = 10^4$  has been used for all experiments in this paper. Once all possible line segments have been found, potential junction locations are restricted to a small neighborhood around each endpoint of those line segments. Figure 10 displays all line segments detected with  $\lambda = 10^4$  for a given image and shows the corresponding junction candidates in red.

Table 2 shows the number of detections and the running time of the complete junction detection procedure, with and without use of the LSD preprocessing. The comparisons are

implemented on three types of images: images containing both geometry and texture, images containing mostly textures and photographs of abstract paintings (see Fig. 11). Using candidates from LSD clearly reduces the computing time, while the quantity of detections is not affected. Notice that this reduction strongly depends on image structures. The simpler the structures contained within the images, the larger the achieved reduction.

### 4.4 Algorithm Pipeline

The pipeline of the whole *a contrario* junction detection algorithm is summarized in Algorithm 7. Remark that the parameters to be set in this algorithm are: the detection threshold  $\epsilon$ , the maximum order of junctions  $M'$  and the radiuses  $r_{min}$ ,  $r_{max}$ ,  $r_{gap}$  (see Sect. 4.1).

In the rest of the paper, we refer to this algorithm by the acronym ACJ (for A Contrario Junction detection).

## 5 Experimental Analysis

This section gathers experiments illustrating the performances of the *a contrario* junction detection (ACJ). When

<sup>3</sup> The code of LSD can be downloaded from the IPOL website: [http://www.ipol.im/pub/algo/gjmr\\_line\\_segment\\_detector/](http://www.ipol.im/pub/algo/gjmr_line_segment_detector/).

**Fig. 11** Test images for the speed up evaluation. *Top* house, Lena and window; *mid* autumn, park and branches; *bottom*: images of geometric paintings: *Geometric* by John Cooper, *Composition* by Charmion von Wiegand, and *Suprematism* by Kazimir Malevich




---

#### Algorithm 7 Complete junction detection procedure

---

**Input:** Image  $I$ , threshold  $\epsilon$ , maximal order  $M'$ , parameters  $r_{min}$ ,  $r_{max}$ ,  $r_{gap}$ .

- (1) Use LSD to preselect junction candidates  $\mathcal{C}$ ;
  - (2) Detect  $\epsilon$ -meaningful junctions for all orders smaller than  $M'$  and for all scales between  $r_{min}$  and  $r_{max}$  by using Algorithm 2. Store junctions of order  $M$  in a list  $Jlist(M)$ .
  - (3) For each  $M \leq M'$ , refine all junctions in  $Jlist(M)$  by using Algorithm 6.
  - (4) Keep only maximal junctions in the lists  $Jlist(M)$  by using Algorithm 3 and 4.
- 

possible, performances will be compared with the classical Harris detector (Harris and Stephens 1988) and the recent “*Pj on gPb*” method (Maire et al. 2008) and “*SFOP*” method (Förstner et al. 2009). For the Harris detector, the “cornerness” is defined as  $det(H) - k \cdot Tr(H)^2$ , where  $H$  is the Harris matrix and the default value  $k = 0.06$  is chosen. For *Pj on gPb* and *SFOP*, the codes kindly provided by the authors are used.

Section 5.1 illustrates the stability of the detection for different images and the control of the number of false detections in a random noise image. We then investigate the invariance properties of the method: scale and contrast changes are respectively studied in Sects. 5.2 and 5.3. In particular, we

show the ability of the method to accurately detect both the scale and the position of a junction. Evaluation on a benchmark is given and discussed in Sect. 5.4 and some more results are displayed in Sect. 5.5.

#### 5.1 Stability and Control of the Number of False Detections

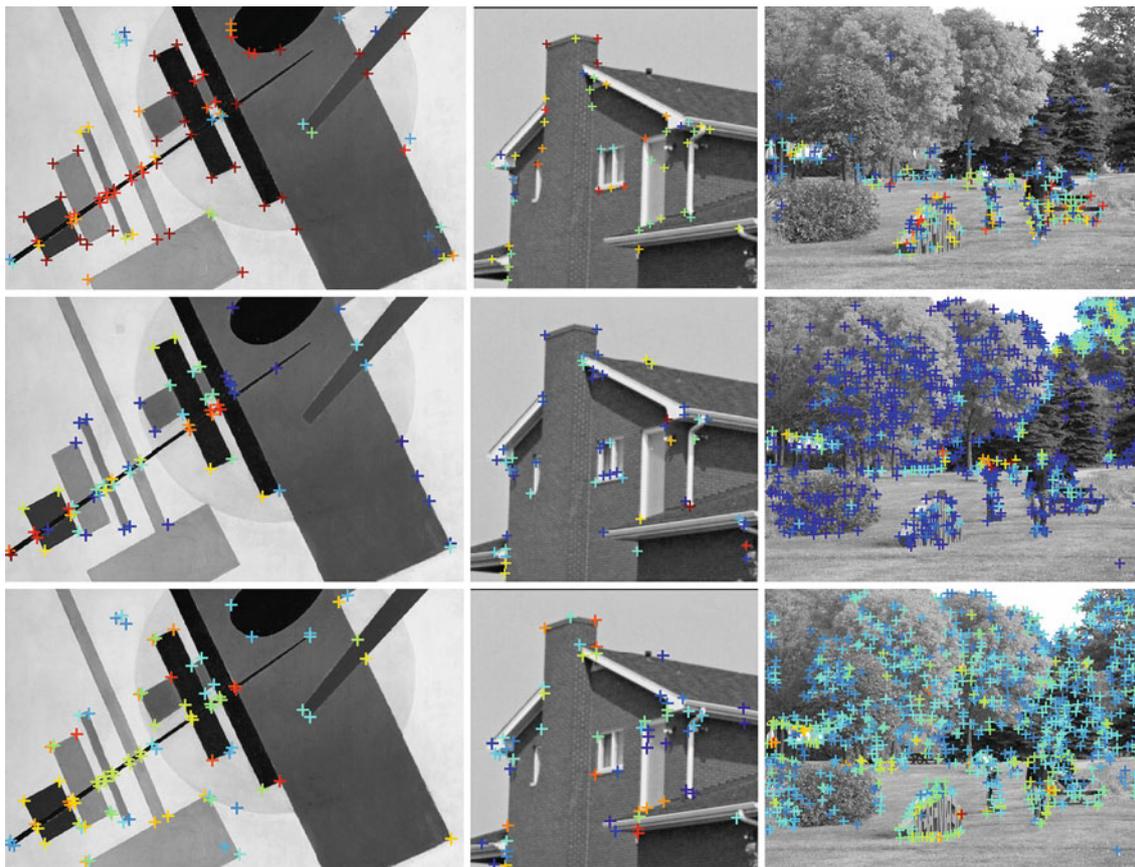
One great quality of a *contrario* detection methods relies in the fact that the threshold  $\epsilon$  has an intuitive meaning: it is an upper bound of the average number of false detections in an image following  $\mathcal{H}_0$ . In this section, we first check that the average number of false detections in Gaussian noise images is less than  $\epsilon$  (Proposition 2). We then illustrate how the threshold  $\epsilon$  controls the number of detections in natural images.

##### 5.1.1 False Alarms Under Hypothesis $\mathcal{H}_0$

In order to test the a *contrario* model, we first consider 1000 large Gaussian noise images  $J$  (pixel values are i.i.d. random variables following a standard normal distribution). Each of these images has a size  $700 \times 700$ . The gradient of these images is then normalized locally (on  $5 \times 5$  windows) and globally, as described in Sect. 2.1. In such images, the norm of

**Table 3** Average numbers of  $\epsilon$ -meaningful junctions when  $\mathcal{H}_0$  is satisfied

$\epsilon$	$10^{-3}$	$10^{-2}$	$10^{-1}$	$10^0$	$10^1$	$10^2$	$10^3$
L-junctions	0	0.008	0.078	0.778	7.150	71.006	678.463
Y-junctions	0	0.005	0.045	0.539	5.106	42.207	332.966
X-junctions	0.001	0.004	0.026	0.216	1.805	13.708	103.531



**Fig. 12** Junctions obtained by ACJ (*top row*), Harris (*mid row*) and “*Pj on gPb*” (*bottom row*). Parameters are fixed for the three images, yielding the same number of detections on the house image (this correspond to  $\epsilon = 1$ ). Observe that only ACJ prevent from over-detection

in textured areas. The *color* of the junction depends on the NFA value (*red* corresponds to small values, i.e. very meaningful junctions and *blue* corresponds to high values) (Color figure online)

the gradient at each pixel follows approximately a Rayleigh distribution and the orientation of the gradient is uniformly distributed. However, the values of the gradient norms and directions at neighboring pixels are not independent. We thus downsample each  $\nabla J$  by a factor of 7 to obtain a new gradient image  $\nabla I$ . This ensures that the values of  $\|\nabla I\|$  are independent and follow approximately a Rayleigh distribution. The gradient orientations are also independent and follow a uniform distribution on  $[0, 2\pi[$ .

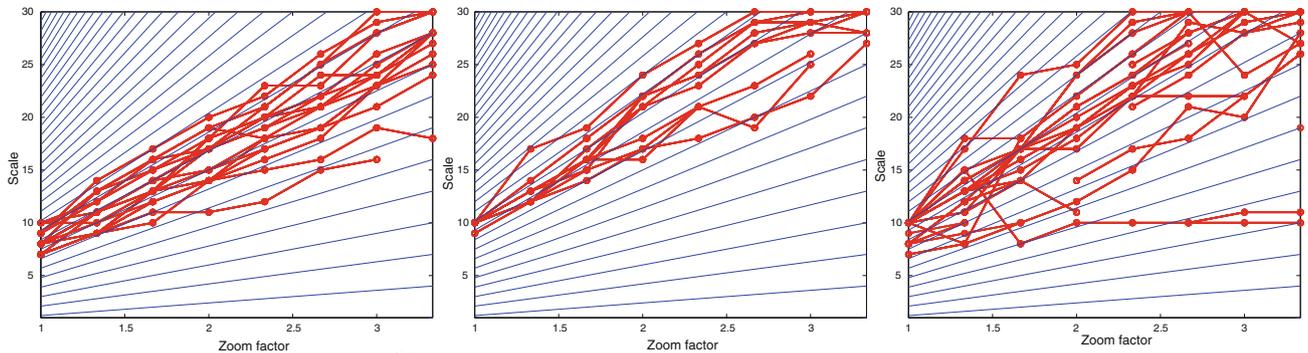
On each of these gradient images, the validation experiment consists of the following procedure: for different values of the detection threshold  $\epsilon$  and for all images, all  $\epsilon$ -meaningful L-, Y- and X-junctions are detected. This is done thanks to a slight modification of Algorithm 2 : in the

algorithm, all possible junctions of order  $M$  are tested, and not only the best one at each point, since we wish to count all  $\epsilon$ -meaningful junctions and not only maximal ones. Table 3 illustrates the results of this experiment, averaged over the 1,000 images, with  $\epsilon$  changing from  $10^{-3}$  to  $10^3$ . Observe that the average number of detections is always smaller than  $\epsilon$ .

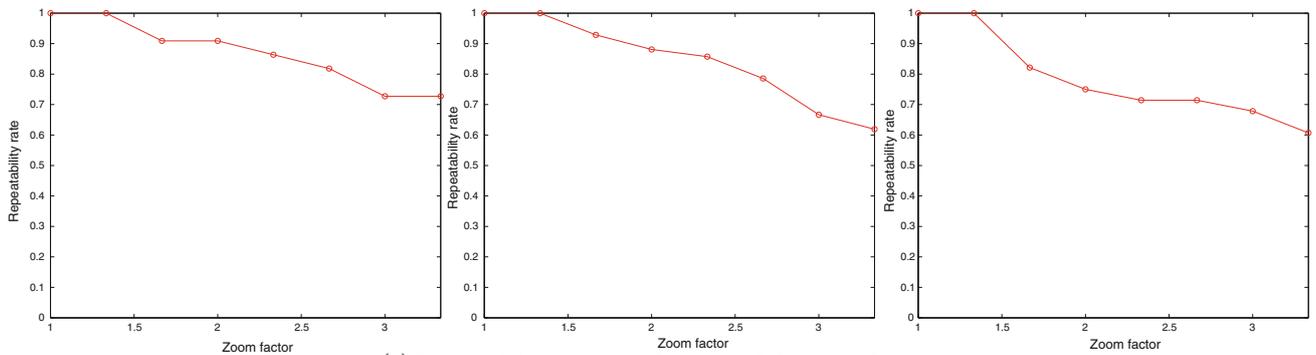
Observe that the previous experiment is actually independent of the noise level, since we globally normalize gradients in the algorithm. Invariance to the noise level is actually stronger. Indeed, the local normalization of gradients makes the detection invariant to local affine contrast transforms, thus to local variations in noise level, as may for instance be observed with Poisson photon noise. A similar property may



(a) original images



(b) junction scales as a function of the zoom factor  $s_i$



(c) Repeatability rate as a function of the zoom factor  $s_i$

**Fig. 13** Illustration of the scale invariance along the scale space. The *first row* shows the tested images. The *second row* shows the scales of junctions detected by ACJ along all the trajectories (curves in red) as a function of the zoom factor  $s_i$  (the abscissa is  $s_i/0.3$ ). The baselines

$\{y = r \cdot s_i\}$ , where  $r$  changes from 1 to 90, are displayed in blue. The *bottom row* presents the repeatability rate of the junctions as a function of the zoom factor

be obtained by locally estimating the noise variance, as in (Förstner 1994).

### 5.1.2 Stability and Choice of $\epsilon$

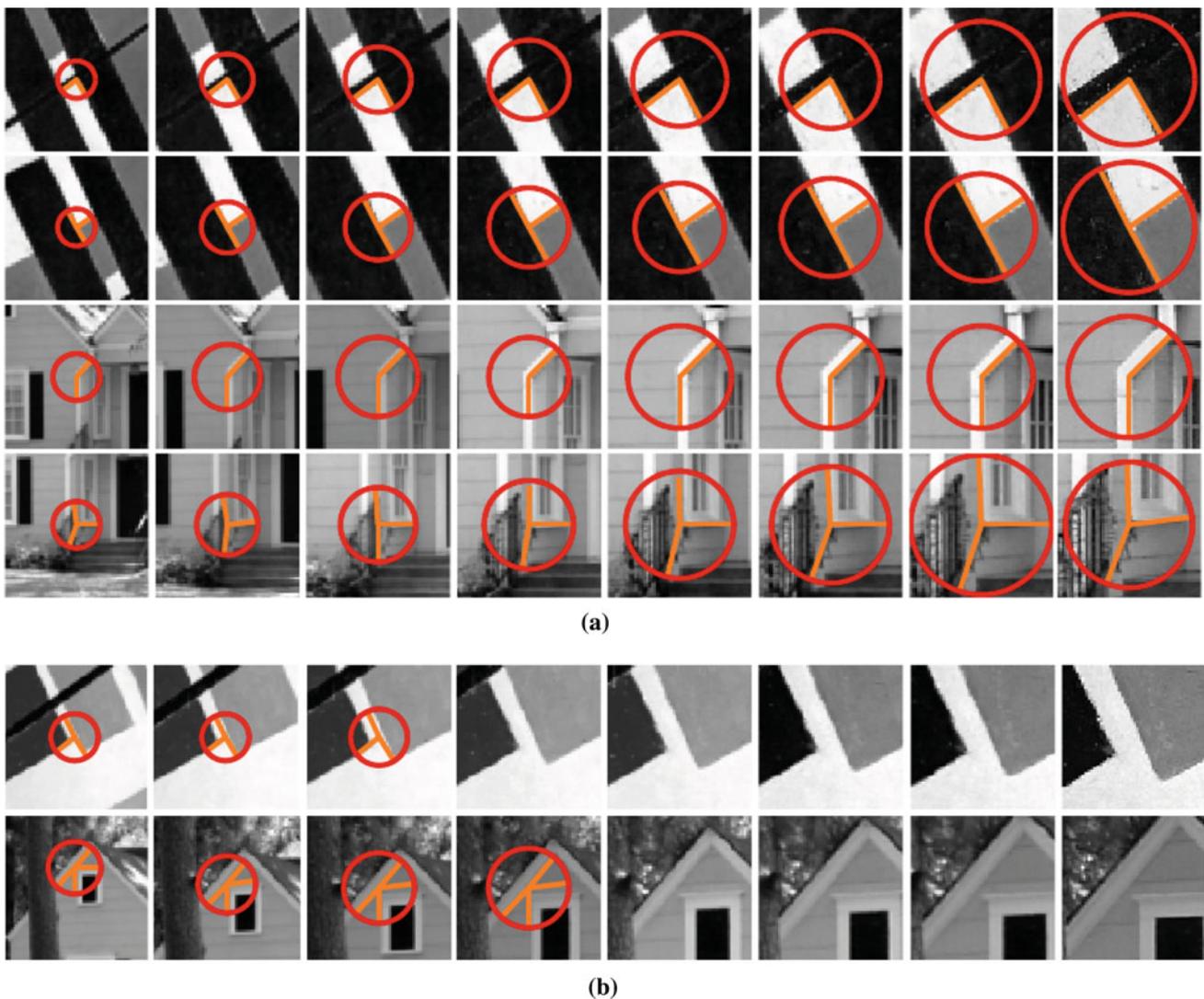
The control of the number of false alarms makes the setting of  $\epsilon$  quite easy in practice: by default, and unless otherwise indicated,  $\epsilon$  is set to 1 in all experiments.

The real strength of the ACJ approach comes from the fact that the value  $\epsilon = 1$  yields very satisfying detections in natural images, whatever their content, size or resolution. In contrast, other approaches such as Harris or “*Pj on gPb*” tend to strongly over-detect in textured areas when using a fixed

parameter, see Fig. 12. This stability property is all the more interesting as it remains valid through scales. For a given choice of  $\epsilon$ , Formula (14) yields different thresholds  $t(r, \epsilon)$  that adapt to the scales of the junctions. As a consequence, by choosing  $\epsilon = 1$ , results are simultaneously satisfying at all scales. This would not be possible with a fixed threshold on junction strengths.

### 5.2 Scale Invariance

This section focus on the behavior of our approach with respect to scale changes. Recall that the scale of a junction is defined as the scale at which the junction is most meaning-



**Fig. 14** Examples of detected junctions along several junction trajectories: *each row* shows a list of junctions detected at the same relative locations in images, from the coarsest to the finest resolution. **(a)** In these examples, junctions can be followed along *complete* trajectories,

and their scales remain roughly proportional to the image resolution. **(b)** These two examples illustrate the difficulty of junction detection in the presence of nearby edges. Observe that as shown in **(b)**, a T-junction might be split into two L-junctions with the increasing of scale

ful and therefore strongly differs from classical approaches relying on the linear scale space. In Sect. 5.2.1, we show that such scales change linearly with the resolution of images, a clearly desirable property. In Sect. 5.2.2, we discuss the interpretation of these scales in images.

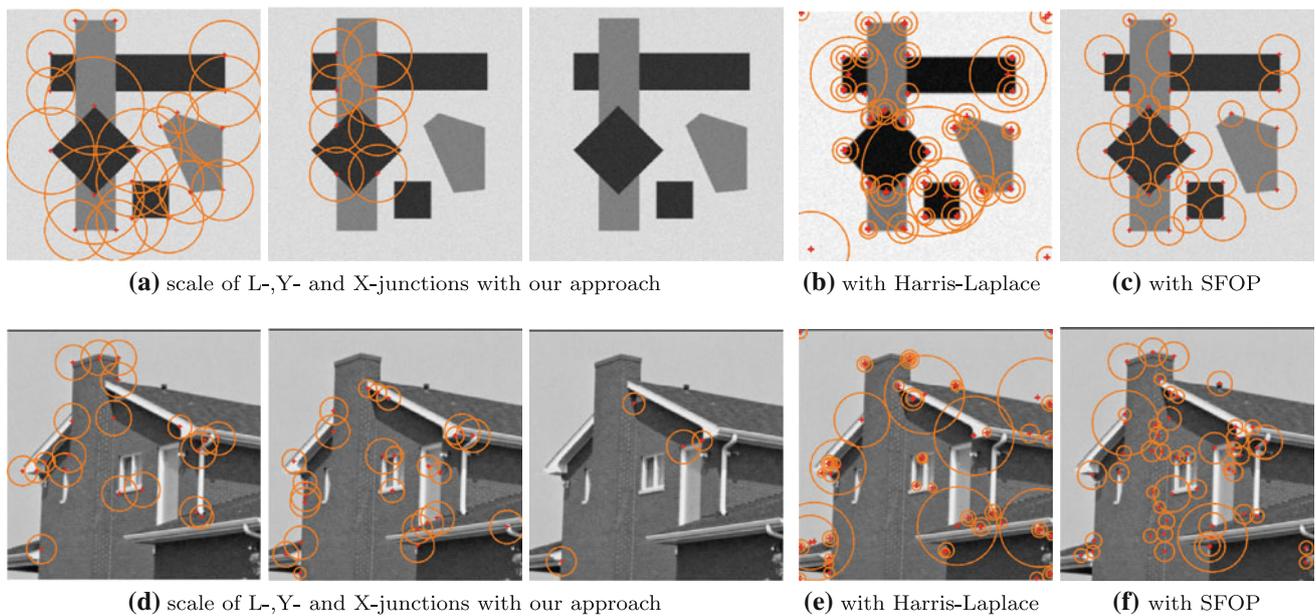
### 5.2.1 Scale and Resolution

In order to investigate the coherence of detections through scale changes, we apply the proposed junction detection algorithm to a sequence of images with different resolutions. An original image  $I$  is resized with eight different zoom factors  $s_0 > \dots > s_7$ , using a bilinear interpolation. The set of factors is chosen as  $\{1, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3\}$ . Algo-

rithm 7 is then applied to each image  $s_i I$  of the sequence and yields a junction list  $\mathcal{J}_{s_i}$ . Then, each junction  $J_0$  of  $\mathcal{J}_{s_0}$  is tracked through resolutions. For this purpose, we define an angular distance between two junctions:

$$\mathcal{S}(J, J_0) = \max_{\theta \in \{\theta_m\}_{m=1}^{M_0}} \min_{\theta' \in \{\theta'_m\}_{m=1}^{M_0}} d_{2\pi}(\theta, \theta'). \quad (20)$$

A junction  $J_0$  in  $\mathcal{J}_{s_0}$  is then matched with  $J_i$  in  $\mathcal{J}_{s_i}$  if they have the same order  $M_0$ , if their centers are close enough, in the sense that  $\|\mathbf{p} - s_i \cdot \mathbf{p}_0\|_2 < 3$ , and if their angular distance  $\mathcal{S}(J_0, J_i)$  is smaller than  $\frac{\pi}{20}$ . If several junctions in  $\mathcal{J}_{s_i}$  satisfy these properties,  $J_0$  is matched with the one minimizing the angular distance  $\mathcal{S}(J_0, J_i)$ . If there is no junction in  $\mathcal{J}_{s_i}$  satisfying these properties, no junction in  $\mathcal{J}_{s_i}$  is matched with  $J_0$ .



**Fig. 15** Scale selection. (a) shows the characteristic scale of L-, Y- and X-junctions given by the proposed approach, and (b), c show the characteristic scale of junctions given by Harris-Laplace and SFOP (Förstner et al. 2009) on a synthetic image, respectively. (d), (f) show the same

comparison on the house image. The location of each junctions is indicated by a red cross and circles have a radius equal to the corresponding scale. Refer to the text for more explanations (Color figure online)

For each junction  $j_0$  in the original image, we call *trajectory* the list of its corresponding junctions through the different resolutions.

Several such trajectories are shown on Fig. 13b. In this experiment, we only consider junctions potentially having complete trajectories in the scale space. Therefore, the maximal scale considered in the coarsest image is  $r_{max} \times 0.3 = 30 \times 0.3 = 9$ . The red curves show the scales of junctions along all the trajectories as functions of  $s_i/0.3$ . The baselines  $\{y = r \cdot s_i\}$ , where  $r$  changes from 1 to 90, are displayed in blue. These correspond to an ideal behavior with respect to resolution changes. Notice that the red curves remain close to the baselines: this implies that the scales of maximal meaningful junctions are quite robust to resolution changes. Figure 14a gives some concrete examples of junctions detected by the ACJ algorithm along their trajectories. Once more, their scales increase linearly with resolution. To the best of our knowledge, the only approaches which permit to obtain a similar coherence between scale detection and resolution are those which rely on linear scale-spaces (Lindeberg 1998; Luo et al. 2007). However, it should be underlined that the use of linear scale-spaces inevitably leads to poor location precisions at large scales. Figure 14b illustrates the aforementioned dilemma (see Sect. 3.4) to detect or not a junction in the presence of nearby edges. In these examples, junctions are not detected at large scale because of the constraint  $r_{gap}$  (see Sect. 3.5). This behavior seems consistent with human perception: when several

edges meet around a large gap, we probably disregard it as a junction.

In a second experiment, shown on Fig. 13c, we compute the repeatability rate of detections with respect to image resolution. More precisely, if we note  $\mathcal{J}_{s_0}(s_0I \rightarrow s_iI)$  the list of junctions that are matched for all intermediate resolutions between  $s_0I$  and  $s_iI$ , the repeatability rate is defined as

$$R(s_i) = \frac{\#\mathcal{J}_{s_0}(s_0I \rightarrow s_iI)}{\#\mathcal{J}_{s_0}}. \quad (21)$$

Observe that, for the three images, the repeatability rate always remains above 60%.

### 5.2.2 How to Interpret the Detected Scales ?

Most of the junction detection approaches in the literature do not provide characteristic scales for their junctions. One notable exception is the Harris-Laplace interest point detector, which makes use of a linear scale space in order to detect keypoints at different scales. Detected points are those which maximize the Laplacian of Gaussian (LoG) in the scale-space and the Harris corner measure in a local space neighborhood. Some results of this detector on a synthetic and a real image are given in Fig. 15b, e. Observe that the scales of junctions do not have a clear interpretation in these images. Another recent approach addressing the scales of junctions is the SFOP detector (Förstner et al. 2009). This method, which draws on a junction model, extends the detector of

(Förstner 1994) in a scale-space framework. Results of this detector are shown on Fig. 15c, f. Although the junctions obtained are less redundant and far easier to interpret than with Harris-Laplace, the detected scales are still difficult to link with the underlying geometry of images. In contrast, the scales detected by ACJ (see Fig. 15a, d) arguably correspond to the optimal size at which one can observe the junction in the image. For instance, the scale of the L-junctions located at the corner of a rectangle is generally chosen as the length of the smaller side of the rectangle, see Fig. 15a. More generally, we observed that *the scale of a junction usually corresponds to the length of its shortest branch*. The fact that the use of the *NFA* as a tool to select the most prominent scale yields such simple geometric behavior is a strong asset of the proposed approach. The same coherence between the characteristic scales of junctions and the sizes of local structures can be observed in the examples of Fig. 14.

### 5.3 Contrast Invariance

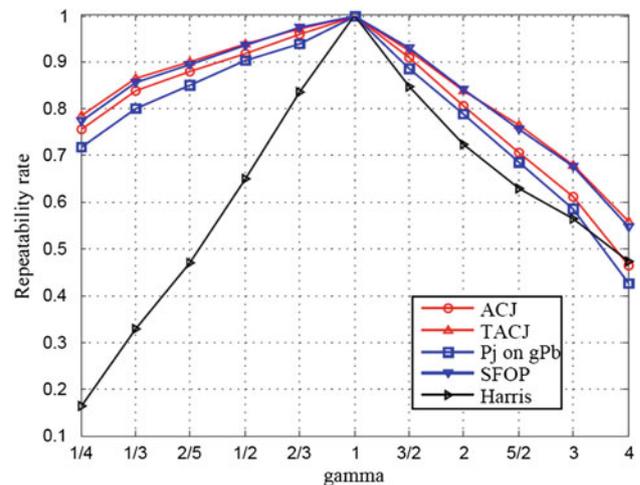
The goal of this section is to evaluate the robustness to contrast changes of different junction detection approaches. To this aim, we create a sequence of images by applying different gamma corrections to an original image  $I$ , with  $\gamma$  in  $\{\frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{2}{3}, 1, 1.5, 2, 2.5, 3, 4\}$ . For each image in the sequence, we detect junctions with

- our ACJ approach, with  $\epsilon = 1$ ,
- Harris corner detector, with a threshold 0.06 on local maxima,
- Maire’s approach using “*Pj on gPb*” (Maire et al. 2008),
- the *SFOP* detector from (Förstner et al. 2009)
- a totally contrast invariant version of ACJ (denoted TACJ), obtained by removing the normalized gradient amplitudes in the definitions (5),(6) of the branch strength.<sup>4</sup>

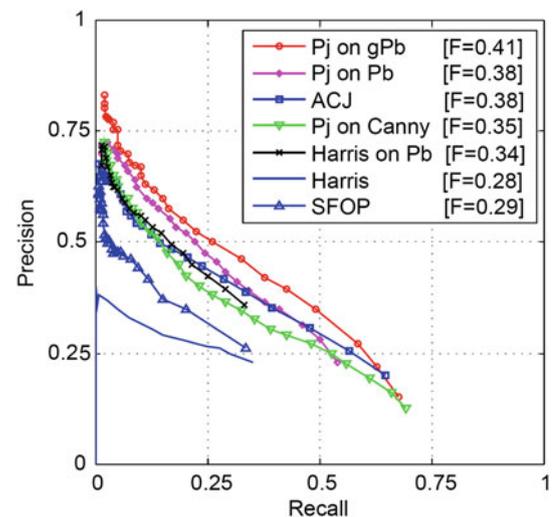
For both “*Pj on gPb*” and the *SFOP* detector, we use the default values of the codes provided by the authors.<sup>5</sup> In order to evaluate the different results, we compute the repeatability rate of each method on the image sequence. More specifically, if  $\mathcal{J}_0$  is the list of junctions in the original image ( $\gamma = 1$ ), and  $\mathcal{J}_i$  the list of junctions in the  $i$ -th image, we note  $\mathcal{J}_0 \cap \mathcal{J}_i$  the set of junctions in  $\mathcal{J}_i$  that can be matched with junctions in  $\mathcal{J}_0$ , where the matching criterion is the

<sup>4</sup> This boils down to replace the distribution of the strength at a pixel with  $\mu = \frac{1}{2}\delta_0 + \frac{2}{\pi} \frac{1}{\sqrt{2-z^2}} \mathbb{1}_{z \geq 0} \cdot \mathcal{L}$  in Eq. (11). The resulting junction detection algorithm is similar to Algorithm 7.

<sup>5</sup> We use the codes kindly provided by the authors: [http://vision.caltech.edu/~mmaire/software/gpb\\_src.tar.gz](http://vision.caltech.edu/~mmaire/software/gpb_src.tar.gz) and <http://www.ipb.uni-bonn.de/sfop/>



**Fig. 16** Repeatability rate of different approaches regarding contrast changes. The curves are averaged over 100 image sequences. Each sequence is obtained by applying different gamma corrections (as specified in the text) to a test image. Each position on the  $x$ -axis corresponds to a given  $\gamma$  correction



**(a)** Precision-Recall curves for different methods

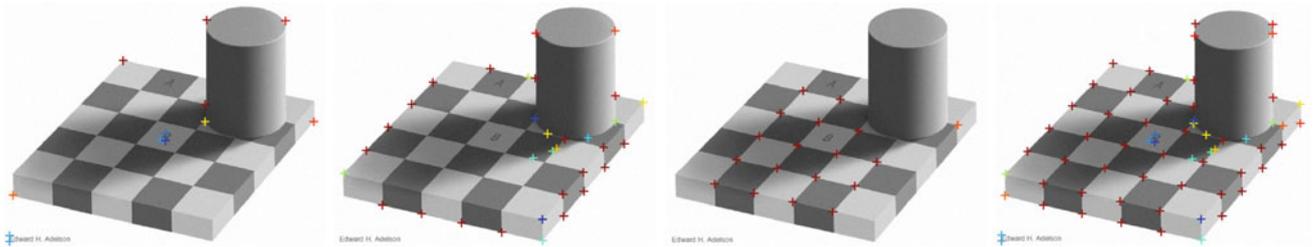
**Fig. 17** The precision-recall curves of different methods on the BSDS-based junction benchmark

one defined in Sect. 5.2.1. Following the same protocol as in (Schmid et al. 2000), the repeatability rate of the  $i$ -th image is then calculated as

$$Repeat(i) = \frac{\#\{\mathcal{J}_0 \cap \mathcal{J}_i\}}{\#\mathcal{J}_0}. \quad (22)$$

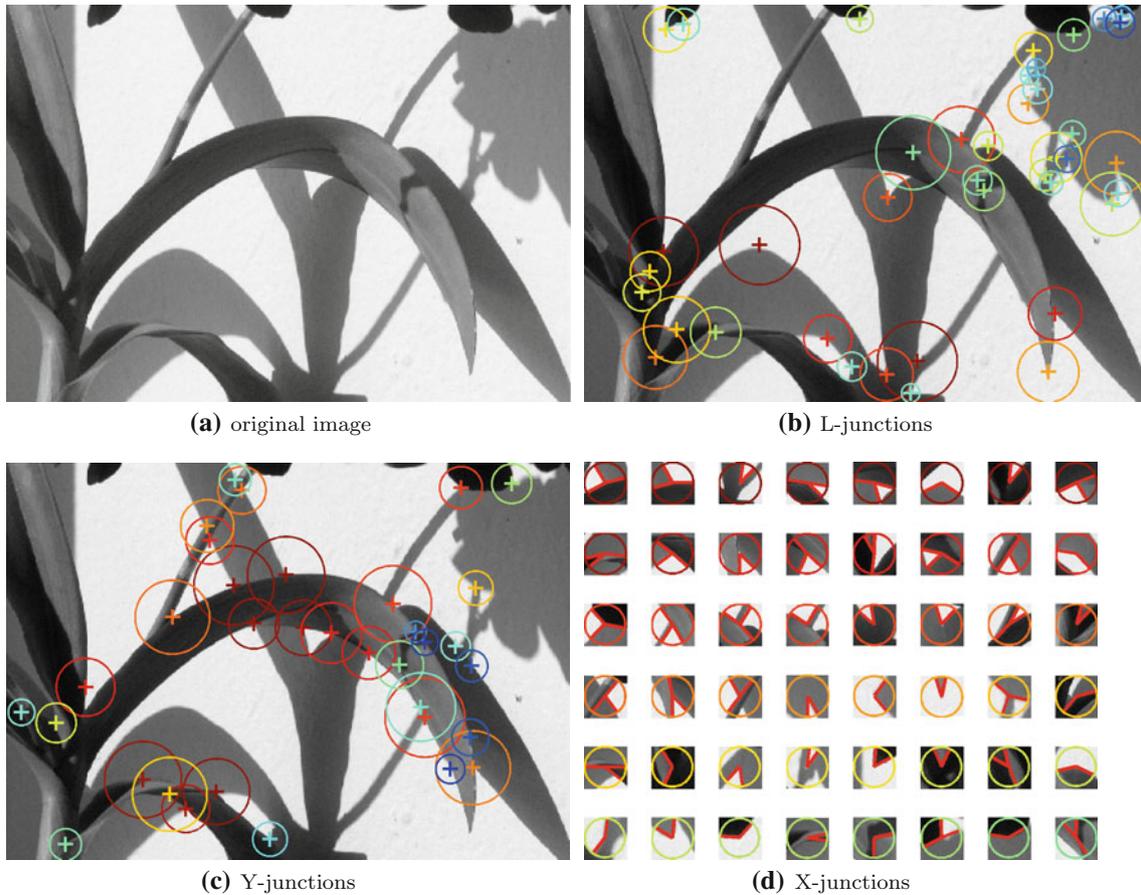
The curves of repeatability rates for the different methods are shown on Fig. 16. Results are averaged on the 100 first images from the natural image database provided by (Rabin et al. 2009).<sup>6</sup> We can see that the Harris detector has the worst

<sup>6</sup> This natural images database is available at <http://perso.telecom-paristech.fr/~gousseau/db/imageDB.zip>



**Fig. 18** Some results of the proposed ACJ approach, with  $\epsilon = 1$ . From left to right L-, Y- and X- junctions. On this simple synthetic image (from E. H. Adelson), junctions are well detected and classified.

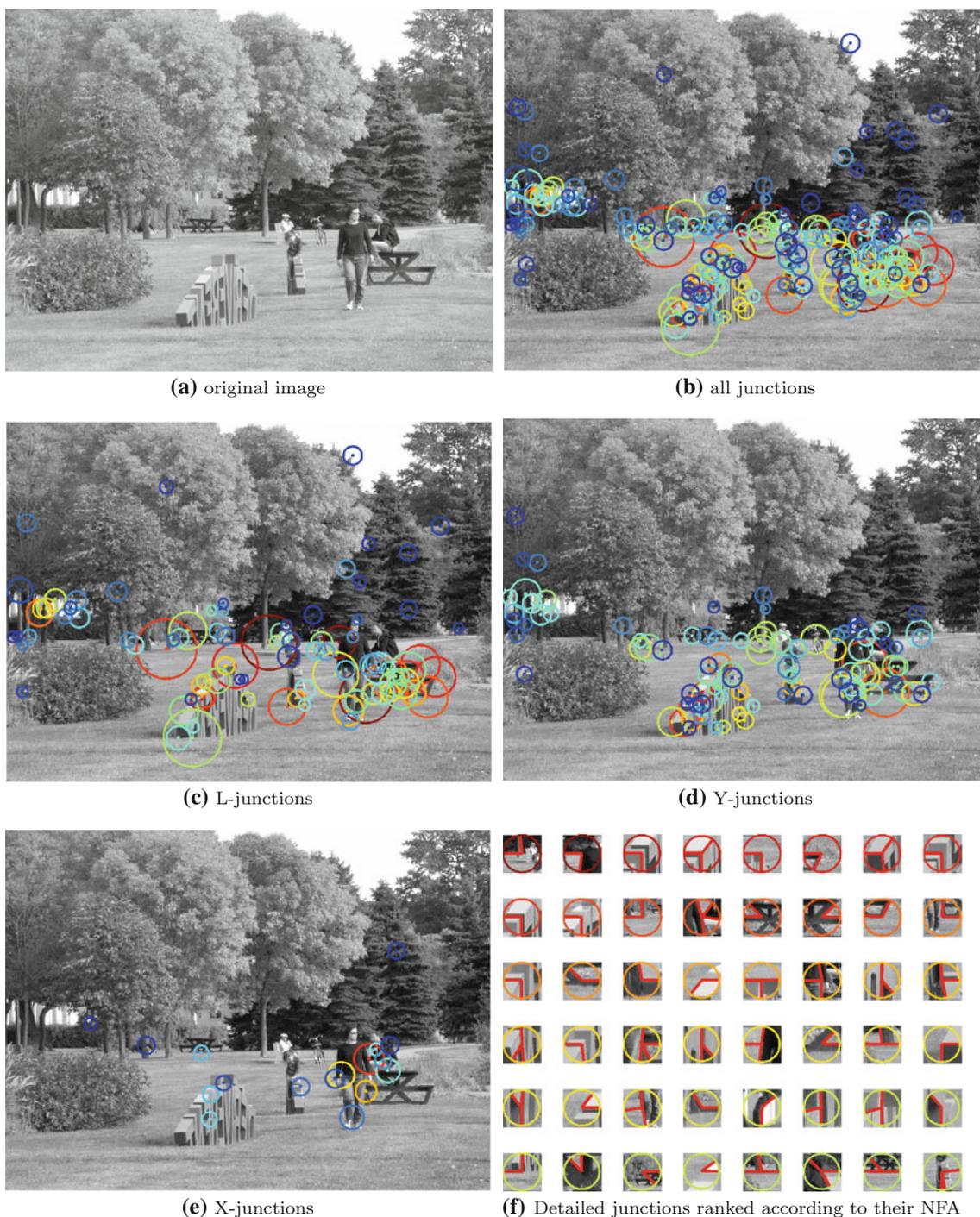
The *color* of the junction depends on the NFA value (*red* corresponds to small values, i.e. very meaningful junctions and *blue* corresponds to high values) (Color figure online)



**Fig. 19** Specific junction detections on a natural image. The color of the junction depends on the NFA value (*red* corresponds to small values, i.e. very meaningful junctions and *blue* corresponds to high values) (Color figure online)

performance with respect to contrast changes, which could be expected since it is the most contrast dependent. Maire’s approach (Maire et al. 2008) gives better results, possibly because it relies on an edge detector that is tuned to match boundaries annotated by humans, which may somehow eliminate contrast variations. The three other approaches perform better, SFOP and TACJ providing slightly better results than ACJ. This could advocate for the choice of TACJ in practice. The good performances of the SFOP detector are inherited from a local estimation of the noise variance, used to

set the detection parameter. The good performances of the ACJ and TACJ detectors are obtained by design and inherited from invariance property of the gradient orientation. In practice, the totally contrast invariant approach appears to be less robust than ACJ to small image modifications, such as those created by JPEG compression. A more in depth study of the specific impact of compression procedures on contrast invariant detection would be of interest. The choice of ACJ results in a compromise between contrast invariance requirements and robustness.



**Fig. 20** Specific junction detections on a natural image. The color of the junction depends on the NFA value (*red* corresponds to small values, i.e. very meaningful junctions and *blue* corresponds to high values) (Color figure online)

#### 5.4 Junction Benchmark

A practical way to evaluate the consistency of our approach with human perception would be to rely on a human annotated junction database. To the best of our knowledge, such a database does not exist. Nevertheless, the well known Berke-

ley segmentation dataset<sup>7</sup> has been used by the authors of (Maire et al. 2008) in order to evaluate different junction detectors. In their paper, Maire et al. use the human anno-

<sup>7</sup> The Berkeley dataset can be downloaded from [www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/](http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/).

tated boundaries to create a ground truth of junctions in the following way: L-junctions are locations of high curvature along these boundaries and Y-junction are locations where more than three regions intersect.

As mentioned before, these authors also propose a procedure called *Pj* to detect junctions from any given set of boundaries. This procedure is applied to the results of different edge detectors, including their detectors *Pb* and *gPb*. The quality of the resulting junctions is assessed by comparing them to the previously created ground truth. We applied the same protocol to the ACJ method proposed in this paper. In order to overcome the strong compression artifacts of the Berkeley database, to which we are sensitive because of strong contrast robustness, we apply a small amount of blur and noise to images before proceeding. Figure 17 presents the results for several junction detection algorithms on the benchmark. The ACJ approach yields a performance of  $F = 0.38$ ,<sup>8</sup> which is better than the baseline Harris detector ( $F = 0.28$ ), the SFOP detector ( $F = 0.29$ ) and *Pj* on *Canny* ( $F = 0.35$ ), but remains below the performance of *Pj* on *Pb* ( $F = 0.38$ ) and *Pj* on *gPb* ( $F = 0.41$ ). The human agreement on the database yields  $F = 0.47$ . The performance of our approach in comparison to *Pj* on *gPb* has to be tempered by two facts. First the junctions of the ground truth stem from a set of human annotated boundary, and the junctions detected by *Pj* stem from a computer segmentation. Therefore, both are boundary-based. It is likely that any detector using a more local junction definition (including ACJ or even Harris) will be penalized in the benchmark. As explained in the introduction and shown by psychophysical studies Rubin (2001), both local and global cues are at play for the perception of junctions. Second, and most important, the parameters of *Pb* and *gPb* are optimized in order to match human detected edges on the same database, which introduces a bias in favor of these methods.

### 5.5 More Results

This last section aims at illustrating the proposed approach with several more visual experiments. Figure 18 presents all the junction detected on a synthetic image. We can observe that the junctions are found with high accuracy and are mostly correctly classified. Figures 19 and 20 present the results on two natural images. Observe that junctions are accurately characterized through their type, localization and scale. Again, we emphasize that the proposed ACJ algorithm yields very few detections in texture areas, see Fig. 20.

<sup>8</sup>  $F$  is defined as the highest value of the quantity  $2 \frac{Precision \cdot Recall}{Precision + Recall}$  along the curve.

## 6 Conclusion and Perspectives

In this work, we have introduced a generic and principled approach for the detection and characterization of junctions in digital images. The proposed procedure is able to inhibit junction detection in textured areas, in contrast with classical approaches. Junctions are accurately characterized through their type, localization and scale. In particular, the method does not rely on the linear scale-space for scale computation, permitting geometric precision.

This work opens several perspectives. First, the accuracy of junction characterization is of importance for depth recovery or motion interpretation. The characterization of junction's type, the difference between T- and Y- junctions or their precise localization claim to be tested in the framework of such applications. Second, and as advocated in the introduction, several psychophysical studies show that the perception of junctions relies on both local and global cues. Therefore the proposed, local, procedure for junction detection could benefit from more global image analysis schemes. The modeling of interactions between junction detection and segmentation procedures is far from trivial, but similar principled approaches could be applied to junction definitions building from color and textured-based region analysis. In the opposite direction, segmentation methods can benefit from meaningful junction detections. Preliminary tests show that the boundary saliency, as for instance defined in (Golubchyc and Lindenbaum 2006), can benefit from a meaningful detection of junctions, in particular by solving ambiguities in boundary connections near these junctions. The characterization of junctions also shows potentials in the automatic interpretation of high-resolution remote sensing images, for instance, in the analysis of built-up areas (Liu et al. 2013).

**Acknowledgments** The authors would like to thank Stéphane Belardi, who implemented a primary version of this algorithm, Michael Lindenbaum for his comments and suggestions and Wolfgang Förstner and Michael Maire, who kindly made their code available. We thank the anonymous reviewers for their interesting comments and suggestions that helped improving the paper. This work has benefited from the support of the ANR project *Callisto* as well as from the FUI project *CEDCA*.

### Appendix 1: Computing NFA in Practice

The NFA of a junction  $J : \{ \mathbf{p}, r, \{ \theta_m \}_{m=1}^M \}$  has been defined as  $NFA(J) := \# \mathcal{J}(M) \cdot F_J(t(J))$ , where

$$F_J(t) = \prod_{m=1}^M \int_t^{+\infty} d \left( \begin{matrix} J(r, \theta_m) \\ \star \\ j=1 \end{matrix} \mu \right),$$

with

$$\mu = \frac{1}{2} \delta_0 + \frac{1}{2} R \cdot \mathcal{L},$$

where  $\mathcal{L}$  is the Lebesgue measure and where  $R$  is the function

$$R(z) = \frac{2}{\sqrt{\pi}} e^{-\frac{z^2}{4}} \cdot \operatorname{erfc}\left(\frac{z}{2}\right) \mathbb{1}_{z \geq 0}.$$

In practice, the numerical values taken by  $F_J(t(J))$ , as a product of  $M$  possibly small values, can become smaller than the precision of the computer when the strength  $t(J)$  is too high. In order to overcome this limitation, we compute instead  $\log(\#\mathcal{J}(M)) + \log(F_J(t(J)))$  and compare this expression to  $\log(\epsilon)$ . The quantity  $\log(F_J(t(J)))$  can indeed be rewritten

$$\log(F_J(t(J))) = \sum_{m=1}^M \log \left( \int_t^{+\infty} d \left( \begin{matrix} J(r, \theta_m) \\ \star \\ j=1 \end{matrix} \mu \right) \right).$$

In order to compute these terms, we discretize  $\mu$  on the interval  $[0, 6]$  (the weight of the distribution  $\mu$  above 6 being negligible), with a precision of 120 bins. Each term of the sum is then computed thanks to a discrete convolution of the distribution  $\mu$ , followed by a discrete integration.

### Appendix 2: Distribution of $\gamma_p(\mathbf{q})$ Under the Null Hypothesis $\mathcal{H}_0$

The goal of this appendix is to compute the distribution of the random variable  $\gamma_p(\mathbf{q})$  under the hypothesis  $\mathcal{H}_0$ . Recall that

$$\gamma_p(\mathbf{q}) = \|\widetilde{\nabla} I(\mathbf{q})\| \cdot \max \left( |\cos(\phi(\mathbf{q}) - \alpha(\vec{p}\vec{q}))| - |\sin(\phi(\mathbf{q}) - \alpha(\vec{p}\vec{q}))|, 0 \right). \tag{23}$$

In the following, we first derive the distribution of  $\max(|\cos \theta| - |\sin \theta|, 0)$  when  $\theta$  is a random variable following a uniform distribution on  $[0, 2\pi[$ , then the full distribution of  $\gamma_p(\mathbf{q})$ .

Distribution of  $\max(|\cos \theta| - |\sin \theta|, 0)$

Assume that  $\theta$  is a random variable following a uniform distribution on  $[0, 2\pi[$ . If  $Y = \max(|\cos \theta| - |\sin \theta|, 0)$ , we have  $Y \in [0, 1]$  with probability 1 and

$$Y = \begin{cases} |\cos \theta| - |\sin \theta| & \text{if } \theta \in [0, \frac{\pi}{4}] \cup [\frac{3\pi}{4}, \frac{5\pi}{4}] \cup [\frac{7\pi}{4}, 2\pi], \\ 0 & \text{otherwise.} \end{cases} \tag{24}$$

Note  $\Theta = [0, \frac{\pi}{4}] \cup [\frac{3\pi}{4}, \frac{5\pi}{4}] \cup [\frac{7\pi}{4}, 2\pi]$ , then  $(|\cos \theta| - |\sin \theta|)^2 = 1 - |\sin 2\theta|$  for  $\theta \in \Theta$ . The cumulative distribution function  $F_Y$  of  $Y$  can then be computed as follows.

For  $y < 0$ ,  $F_Y(y) = 0$ , for  $y > 1$ ,  $F_Y(y) = 1$ .

$$\begin{aligned} \text{For } 0 \leq y \leq 1, F_Y(y) &= \mathbb{P} \left( \max(|\cos \theta| - |\sin \theta|, 0) \leq y \right) \\ &= \frac{1}{2} + \mathbb{P} \left( 0 < |\cos \theta| - |\sin \theta| \leq y \right) \\ &= \frac{1}{2} + \mathbb{P} \left( \theta \in \Theta \text{ and } |\sin 2\theta| \geq 1 - y^2 \right) \\ &= \frac{1}{2} + 4 \cdot \frac{1}{2\pi} \cdot \left( \frac{\pi}{4} - \frac{1}{2} \arcsin(1 - y^2) \right) \\ &= 1 + \frac{1}{\pi} \arcsin(y^2 - 1). \end{aligned} \tag{25}$$

Finally,

$$F_Y(y) = \left( 1 + \frac{1}{\pi} \arcsin(y^2 - 1) \right) \mathbb{1}_{0 \leq y \leq 1}. \tag{26}$$

The distribution of the random variable  $Y$  is obtained by differentiating this expression, which gives

$$\mu_Y = \frac{1}{2} \delta_0 + \frac{2}{\pi} \frac{1}{\sqrt{2 - y^2}} \mathbb{1}_{0 \leq y \leq 1} \mathcal{L}, \tag{27}$$

where  $\delta_0$  is the Dirac mass centered at 0 and where  $\mathcal{L}$  is the Lebesgue measure on  $\mathbb{R}$ .

Distribution of  $|\widetilde{\nabla} I| \cdot \max(|\cos \theta| - |\sin \theta|, 0)$

We compute in this section the distribution of the product  $Z = XY$  when  $X$  and  $Y$  are two independent random variables,  $X$  following a Rayleigh distribution of parameter 1 and  $Y$  following the distribution (27). We recall that the density of the Rayleigh distribution is

$$f_X(x) = \chi(x, 2) = x e^{-x^2/2}, \quad x \in [0, \infty). \tag{28}$$

Now, notice that if  $Z = XY$ , then  $Z \in [0, +\infty)$  with probability 1 and the cumulative distribution function  $F_Z(z)$  can be computed as follows,

$$\begin{aligned} \text{For } z < 0, F_Z(z) &= 0. \\ \text{For } z = 0, F_Z(z) &= \frac{1}{2}. \\ \text{For } z > 0, F_Z(z) &= \mathbb{P}(0 \leq XY \leq z) \\ &= \int_0^1 \int_0^{z/y} f_X(x) dx \mu_Y(dy) \\ &= \frac{1}{2} + \frac{2}{\pi} \int_0^1 (1 - e^{-\frac{z^2}{2y^2}}) \frac{1}{\sqrt{2 - y^2}} dy. \end{aligned}$$

We differentiate this expression with respect to  $z$  to obtain the distribution of the random variable  $Z$ :

$$\mu_Z = \frac{1}{2} \delta_0 + \mathbb{1}_{z \geq 0} \cdot \frac{2z}{\pi} \left( \int_0^1 e^{-\frac{1}{2}(\frac{z}{y})^2} \frac{1}{y^2 \sqrt{2 - y^2}} dy \right) \mathcal{L}. \tag{29}$$

In addition,

$$\begin{aligned} &\int_0^1 \frac{1}{y^2} e^{-\frac{1}{2}(\frac{z}{y})^2} \frac{1}{\sqrt{2 - y^2}} dy \\ &= \int_1^\infty e^{-\frac{z^2 x^2}{2}} \frac{x}{\sqrt{2x^2 - 1}} dx, \text{ with } x = \frac{1}{y}, \\ &= \frac{1}{2} \int_1^\infty e^{-\frac{z^2}{4} \cdot (t^2 + 1)} dt, \text{ with } t = \sqrt{2x^2 - 1}, \\ &= \frac{\sqrt{\pi}}{2z} e^{-\frac{z^2}{4}} \cdot \left( 1 - \operatorname{erf}\left(\frac{z}{2}\right) \right) = \frac{\sqrt{\pi}}{2z} e^{-\frac{z^2}{4}} \cdot \operatorname{erfc}\left(\frac{z}{2}\right), \end{aligned} \tag{30}$$

where  $\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt$  is the error function and where  $\operatorname{erfc}(z) = 1 - \operatorname{erf}(z)$  is the complementary error function. Finally,

$$\mu_Z = \frac{1}{2}\delta_0 + \frac{1}{2}R.\mathcal{L}, \quad (31)$$

where  $\mathcal{L}$  is the Lebesgue measure and where  $R$  is the function

$$R(z) = \frac{2}{\sqrt{\pi}} e^{-\frac{z^2}{4}} \cdot \operatorname{erfc}\left(\frac{z}{2}\right) \mathbb{1}_{z \geq 0}.$$

Observe that the measure  $\mu_Z$  is a mixture between a Dirac mass function and an absolutely continuous distribution.

## References

- Adelson, E. H. (2000). Lightness perception and lightness illusions. In M. Gazzaniga (Ed.), *The new cognitive neurosciences* (2nd ed., pp. 339–351).
- Alvarez, L., & Morales, F. (1997). Affine morphological multiscale analysis of corners and multiple junctions. *International Journal of Computer Vision*, 25(2), 95–107.
- Bergevin, R., & Bubel, A. (2004). Detection and characterization of junctions in a 2d image. *Computer Vision and Image Understanding*, 93(3), 288–309, ISSN 1077–3142.
- Beymer, D. J. (1991). Finding junctions using the image gradient. In *Proceedings of Computer Vision and Pattern Recognition* (pp. 720–721).
- Bolles, R. C., & Cain, R. A. (1982). Recognizing and locating partially visible objects: The local-feature-focus method. *International Journal of Robotics Research*, 1(3), 57–82.
- Cao, F. (2003). Good continuations in digital image level lines. In *Proceedings of International Conference on Computer Vision* (pp. 440–446).
- Caselles, V., Coll, B., Morel, J.-M. (1996). Junction detection and filtering: A morphological approach. In *Proceedings of International Conference on Image Processing* (pp. 493–496).
- Cazorla, M. A., & Escolano, F. (2003). Two Bayesian methods for junction classification. *IEEE Transactions on Image Processing*, 12(3), 317–327.
- Cazorla, M. A., Escolano, F., Gallardo, D., & Rizo, R. (2002). Junction detection and grouping with probabilistic edge models and Bayesian A\*. *Pattern Recognition*, 35(9), 1869–1881.
- Deriche, R., & Blaszk, T. (1993). Recovering and characterizing image features using an efficient model based approach. In *Proceedings of Computer Vision and Pattern Recognition* (pp. 530–535).
- Deriche, R., & Giraudon, G. (1993). A computational approach for corner and vertex detection. *International Journal of Computer Vision*, 10(2), 101–124.
- Deschenes, F., & Djemel, Ziou. (2004). Detection of lines, line junctions and line terminations. *International Journal of Remote Sensing*, 25(3), 511–535.
- Desolneux, A., Moisan, L., & Morel, J.-M. (2000). Meaningful alignments. *International Journal of Computer Vision*, 40(1), 7–23.
- Desolneux, A., Moisan, L., & Morel, J.-M. (2008). *From Gestalt theory to image analysis: A probabilistic approach*. Berlin: Springer.
- Dickscheid, T., Schindler, F., & Förstner, W. (2011). Coding images with local features. *International Journal of Computer Vision*, 94(2), 154–174.
- Dimiccoli, M., & Salembier, P. (2009). Exploiting t-junctions for depth segregation in single images. In *Proceedings of ICASSP* (pp. 1229–1232).
- Förstner, W. (1986). A feature based correspondence algorithm for image matching. In *ISP Commission III*.
- Förstner, W. (1994). A framework for low level feature extraction. In *Proceedings of European Conference on Computer Vision ECCV 1994* (Vol. II, pp. 383–394).
- Förstner, W., Dickscheid, T., Schindler, F. (2009). Detecting interpretable and accurate scale-invariant keypoints. In *ICCV* (pp. 2256–2263).
- Freeman, W. T. (1992). *Steerable filters and local analysis of image structure*. PhD thesis, MIT, Cambridge, MA, USA.
- Fuchs, C., & Förstner, W. (1995). Polymorphic grouping for image segmentation. In *Proceedings of International Conference on Computer Vision* (pp. 175–182).
- Geiger, D., Kumaran, K., & Parida, L. (1996). Visual organization for figure/ground separation. In *Proceedings of Computer Vision and Pattern Recognition* (pp. 155–161).
- Golubchyc, R., & Lindenbaum, M. (2006). Improving the saliency algorithm by cue optimization. Technical Report MSC-2006-07, Technion.
- Guo, C.-E., Zhu, S.-C., Wu, Y.-N. (2007). Primal sketch: Integrating structure and texture. *Computer Vision and Image Understanding*, 106(1):5–19. Special issue on Generative Model Based Vision.
- Guzmán, A. (1968). Decomposition of a visual scene into three-dimensional bodies. In *Proceedings of Fall Joint Computer Conference* (part I, pp. 291–304).
- Hannah, M. J. (1974). *Computer matching of areas in stereo images*. PhD thesis, Stanford University.
- Harris, C., & Stephens, M. (1988). A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vision Conference* (pp. 147–151).
- Ishikawa, H., & Geiger D. (1998). Segmentation by grouping junctions. In *Proceedings of Computer Vision and Pattern Recognition* (pp. 125–132).
- Kanizsa, G. (1979). *Organization in vision: Essays on gestalt perception*. New York: Praeger.
- Kenney, C. S., Zuliani, M., Manjunath, B. S. (2005). An axiomatic approach to corner detection. In *Proceedings of Computer Vision and Pattern Recognition* (pp. 191–197).
- Kitchen, L., & Rosenfeld, A. (1982). Non-maximum suppression of gradient magnitudes makes them easier to threshold. *Pattern Recognition Letters*, 1(2), 93–94.
- Kovesi, P. (2003). Phase congruency detects corners and edges. In *Proceedings of DICTA* (pp. 309–318).
- Leichter, I., & Lindenbaum, M. (2009). Boundary ownership by lifting to 2.1d. In *IEEE 12th International Conference on Computer Vision 2000* (pp. 9–16). 29 Nov to 2 Oct 2009. doi:10.1109/ICCV.2009.5459208.
- Lin, L., Peng, S., Porway, J., Zhu, S.-C., & Wang, Y. (2007). An empirical study of object category recognition: sequential testing with generalized samples. In *Proceedings of International Conference on Computer Vision*.
- Lindeberg, T. (1994). Junction detection with automatic selection of detection scales and localization scales. In *Proceedings of International Conference on Image Processing* (pp. 924–928).
- Lindeberg, T. (1998). Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2), 79–116.
- Lindeberg, T., & Li, M.-X. (1997). Segmentation and classification of edges using minimum description length approximation and complementary junction cues. *Computer Vision and Image Understanding*, 67(1), 88–98.
- Liu, G., Xia, G.-S., Huang, X., Yang, W., Zhang, L. (2013). A perception-inspired building index for automatic built-up area detection in high-resolution satellite images. In *Proceedings of International Geoscience and Remote Sensing Symposium* (pp. 1–4).
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91–110.

- Luo, B., Aujol, J.-F., Gousseau, Y., Ladjal, S., & Maître, H. (2007). Resolution-independent characteristic scale dedicated to satellite images. *IEEE Transactions on Image Processing*, 16(10), 2503–2514.
- Maire, M., Arbelaez, P., Fowlkes, C., Malik, J. (2008). Using contours to detect and localize junctions in natural images. In *Proceedings of Computer Vision and Pattern Recognition* (pp. 1–8).
- Marr, D. (1982). *Vision. A computational investigation into the human representation and processing of visual information*. San Francisco: W.H. Freeman and Company.
- McDermott, J. (2004). Psychophysics with junctions in real images. *Perception*, 33, 1101–1127.
- Metelli, F. (1974). The perception of transparency. *Scientific American*, 4(230), 90–98.
- Mikolajczyk, K., & Schmid, C. (2004). Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60, 63–86.
- Mohammad, A., & Lu, G. (2008). An improved curvature scale-space corner detector and a robust corner matching approach for transformed image identification. *IEEE Transactions on Image Processing*, 17(12), 2425–2441.
- Moisan, L., & Stival, B. (2004). A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. *International Journal of Computer Vision*, 57, 201–218.
- Mokhtarian, F., & Suomela, R. (1998). Robust image corner detection through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20, 1376–1381.
- Moravec, H. P. (1977). Towards automatic visual obstacle avoidance. In *Proceedings of IJCAI* (pp. 584).
- Musé, P., Sur, F., Cao, F., Gousseau, Y., & Morel, J.-M. (2006). An contrario decision method for shape element recognition. *International Journal of Computer Vision*, 69(3), 295–315.
- Neubeck, A., & Van Gool, L. (2006). Efficient non-maximum suppression. In *Proceedings of International Conference on Pattern Recognition* (pp. 850–855).
- Parida, L., Geiger, D., & Hummel, R. A. (1998). Junctions: Detection, classification, and reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(7), 687–698.
- Perona, P. (1992). Steerable-scalable kernels for edge detection and junction analysis. *Image and Vision Computing*, 10(10), 663–672.
- Rabin, J., Delon, J., & Gousseau, Y. (2009). A statistical approach to the matching of local features. *SIAM Journal on Imaging Sciences*, 2(3), 931–958.
- Rattarangsi, A., & Chin, R. T. (1992). Scale-based detection of corners of planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14, 430–449.
- Ren, X., Fowlkes, C. C., Malik, J. (2006). Figure/ground assignment in natural images. In *Proceedings of European Conference on Computer Vision* (Vol. 2, pp. 614–627).
- Rohr, K. (1992). Recognizing corners by fitting parametric models. *International Journal of Computer Vision*, 9(3), 213–230.
- Rosenfeld, A., & Johnston, E. (1973). Angle detection on digital curves. *IEEE Transactions on Computers*, 22(9), 875–878.
- Rosenthaler, L., Heitger, F., Kübler, O., von der Heydt, R. (1992). Detection of general edges and keypoints. In *Proceedings of the Second European Conference on Computer Vision, ECCV 1992* (pp. 78–86). ISBN 3-540-55426-2.
- Rosten, E., Porter, R., & Drummond, T. (2010). Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1), 105–119.
- Rubin, N. (2001). The role of junctions in surface completion and contour matching. *Perception*, 3(30), 339–366.
- Ruderman, D. L., & Bialek, W. (1994). Statistics of natural images: Scaling in the woods. *Physical Review Letters*, 73(6), 814–817.
- Ruzon, M. A., & Tomasi, C. (2001). Edge, junction, and corner detection using color distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11), 1281–1295.
- Schmid, C., Mohr, R., & Bauckhage, C. (2000). Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2), 151–172.
- Shi, J., & Tomasi, C. (1994). Good features to track. In *Proceedings of Computer Vision and Pattern Recognition* (pp. 593–600).
- Sinzinger, E. D. (2008). A model-based approach to junction detection using radial energy. *Pattern Recognition*, 41(2), 494–505.
- Smith, S. M., & Brady, J. M. (1995). SUSAN: A new approach to low level image processing. *Proceedings of International Journal of Computer Vision*, 23, 45–78.
- Teh, C. H., & Chin, R. T. (1989). On the detection of dominant points on digital curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8), 859–872.
- Trajkovic, M., & Hedley, M. (1998). Fast corner detection. *Image and Vision Computing*, 16(2), 75–87.
- Grompone von Gioi, R., Jakubowicz, J., Morel, J.-M., & Randall, G. (2010). LSD: A fast line segment detector with a false detection control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32, 722–732.
- Wallach, H. (1935). Ueber visuell whargenommene bewegungsrichtung. *Psychologische Forschung*, 20, 325–380.
- Wang, H., & Brady, M. (1995). Real-time corner detection algorithm for motion estimation. *Image and Vision Computing*, 13(9), 695–703.
- Wu, T.F., Xia, G.-S., Zhu, S.-C. (2007). Compositional boosting for computing hierarchical image structures. In *Proceedings of Computer Vision and Pattern Recognition* (pp. 1–8).
- Xia, G.-S., Delon, J., & Gousseau, Y. (2012). An accurate and contrast invariant junction detector. In *Proceedings of International Conference on Pattern Recognition*.