A framework for junction detection using local Hough transforms

Andrzej SLUZEK
Dept of Electrical and Computer Engineering
Khalifa University
Abu Dhabi, UAE
andrzej.sluzek@kustar.ac.ae

Abstract— **A novel method of junction detection in images is investigated. Junctions are modeled using locally applied Hough transforms. Two types of the Hough transform are proposed: (a) 1D transforms to characterize orientations of the junction segments and (b) 2D transforms to additionally estimate lengths of the segments. For any location of the circular window, the best-fit junction can be found and its *strength* (visual prominence) can be calculated. Then, actual junctions are detected as the local maxima of the *strength* function. To reduce computational complexity, junction detection and localization is based on the 1D model, while the 2D model can be used to further characterize detected junctions. The paper presents both theoretical issues of the method and its practical limitations. Relations to alternative techniques of junction detection are highlighted.**

Keywords-component; local features, line junctions, Hough transform, image gradient, interest points

I. INTRODUCTION

From the early days of machine vision, junctions of lines have been considered one of the most important local features for image processing and analysis. However, the initial works were more focused on interpretation of line-drawing images containing junctions (e.g., Waltz in [25] or Cooper in [3]) and/or on efficient processing of such drawings (e.g., Freeman in [9]) rather than on the practical issues of junction detection in images on natural quality. Results were usually demonstrated on synthesized/simplified images where basic techniques (often assisted by humans) provided satisfactory junction detection. In the following years, however, techniques for junction detection in more realistic conditions have been proposed.

In general, we can classify junction detection methods into two categories: direct methods and indirect ones. Alternative classification criteria also exist (e.g., Sinzinger in [21] identifies signal-based and model-based techniques) so that alternative categorizations of junction detection algorithms can be obtained by combining various criteria.

Indirect algorithms of junction detection require a pre-detection of other features from which the presence of junctions can be revealed. Typically, junctions are built from pre-detected line segments (e.g. Matas and Kittler in [15]) or extracted from already found contours (e.g., Deschenes and Ziou in [6]). Sometimes, junctions are identified using more complex pre-detected structures. For example, Faas and van Vliet in [8] detect junctions in images containing thick lines using streamlines of flow vector fields, while Mahadevan and Casasent in [14] identify junctions in microscopic images as meeting points of pre-detected regions.

Direct algorithms detect junctions (and estimate their characteristics, i.e. the number and orientation of segments) directly from the image intensities. Simple detectors (that only localize junctions) are practically equivalent to corners detectors (e.g., [5], [10], [13] or [24]) although they often better localize corners than multi-segment junctions. For example, typical keypoint detectors (see a survey in [16]) usually miss the actual locations of T-junctions (more details in Section 5). Advanced junction detectors both localize and characterize junctions, i.e. determine the number and orientation of segments (e.g., [12] or [21]); sometimes they additionally reconstruct intensities/colours of adjacent regions (e.g., [1], [19] or [23]). Direct junction detectors are local operators, i.e. junctions are identified within (almost always circular) windows. The size of windows is rather small for simple detectors (e.g. a 37-pixel window in SUSAN – see [22]) to avoid excessive computational complexity, while advanced direct detectors need larger windows to acquire enough data to approximate the junctions' geometry (e.g. 10-25 pixel radii in [19], [21] and [23]). Sometimes, the size of window is adaptively determined by the detector (e.g., Lindeberg in [13]).

The method proposed in this paper can be classified as an advanced direct detector of junctions, reconstructing the junction's geometry only. We use circular windows of a significant radius (typically 15-30 pixels, although windows of any larger size can be alternatively used) and a standard radial model of junctions. Actually, our approach is similar to the ideas presented by Sinzinger in [21] and we often compare our method to this reference. In particular, we explain that in junction detection the concept of *radial energy* is practically equivalent to the *gradient-weighted Hough transform* (*GWHT*) originally proposed by O'Gorman and Clowes in [18] and later applied and/or improved in other works (e.g., Cucchiara and Filicori in [4]). By using the Hough transform, we can simplify the mathematical complexity of the Sinzinger's model and

introduce additional useful functionalities that are not available in his work.

In Sections II and III, the principles of junction fitting based on locally applied Hough transforms (in 1D and 2D parameter spaces, correspondingly) are introduced. Section IV presents the mechanism of junction localization, including representative exemplary results. Finally, in Section V we discuss some other related issues, for example how to use junctions in the context of keypoint detection.

II. 1D Model of Junction Fitting

The Hough transform, [7], is a popular tool for fitting line/curve features onto scattered contour data. For example, it has been used as a mechanism to determine geometry of previously identified corners or other junctions (e.g. Shen and Wang in [20] or Barrett and Petersen in [2]). In the Hough transform model, each instance $\boldsymbol{C}$ of a curve family specified by an n-parameter equation

$$f(x, y, p_1, ..., p_n) = 0 \tag{1}$$

is represented by a single point $\{p_1^C, \ldots, p_n^C\}$ in an n-dimensional parameter space $P_1 \times P_2 \times \ldots \times P_n$. Therefore, a k-segment junction located at (x_0, y_0) coordinates would be, in general, represented by a point in $(k+2)$-dimensional space $X \times Y \times \Theta_1 \times \ldots \times \Theta_k$, where dimensions $\Theta_1, \ldots, \Theta_k$ specify orientations of the corresponding segments (see Fig. 1A). However, if the junction point coordinates (x_0, y_0) are known (either predefined or already found) we actually need only a 1D space Θ to characterize a k-segment junction by k points $\{\theta_1, \ldots, \theta_k\} \in \Theta$, where each value θ_i defines the orientation of the corresponding i^{th} segment.

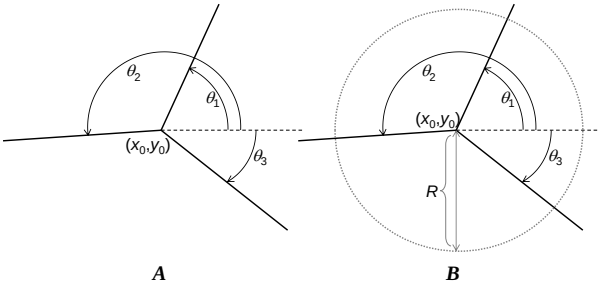

Figure 1. Specification of an exemplary 3-segment junction (A) and the same junction confined to a circle of radius $\boldsymbol{R}$ (B).

The junction model is usually confined to a circle of a certain radius $\boldsymbol{R}$ around the junction point (Fig. 1B) because distant parts of contours joining in the junction point are usually less interesting. Therefore, the junction-fitting problem can be formulated as follows:

Given a circular window of a radius $\boldsymbol{R}$ *located at* (x_0, y_0) *coordinates, determine the number and orientation of segments originating from* (x_0, y_0) *that form the junction approximating the window's content (i.e. find the optimum values of k and* $\theta_1, \ldots, \theta_k$ *orientation angles).*

A. Hough transforms for junction fitting

The Hough transform (often abbreviated as *HT*) converts a 2D image into a set of values in the corresponding parameter space. Given a curve family specified by Eq.1, a pixel (x_c, y_c) contributes a vote to a parameter space point $\{p_1^A, \ldots, p_n^A\}$ if the corresponding curve intersects (x_c, y_c), i.e.

$$f(x_c, y_c, p_1^A, ..., p_n^A) = 0 \tag{2}$$

In practice the parameter space is decomposed into a discrete set of accumulators (bins) so that the best-fit approximations are found by identifying bins (rather than points) with the largest numbers of votes accumulated.

In case of using a 1D orientation space Θ for detection of the best-fit junctions within circular windows, any pixel (x_c, y_c) can vote for only one angle $\theta_c = \text{Atan2}(y_c, x_c)$. Therefore, the Hough transform $HT(\theta)$ for junction fitting can be actually obtained by:

(a) conversion of the Cartesian coordinates (x, y) into the polar representation (θ, r), and

(b) integration of the pixel votes along radial directions, i.e.

$$HT(\theta) = \int_0^R v(\theta, r)\, dr \tag{3}$$

where v is a function defining how individual pixels vote.

Originally, the Hough transform was built over contour images and the v function was binary (**1** for contour pixels and **0** otherwise). A modified approach was proposed by O'Gorman and Clowes in [18]; so-called the *gradient-weighted Hough transform* (*GWHT*). The pixels vote for the *GWHT* space bins proportionally to the magnitude of their gradient projections onto the normal vectors of the corresponding curves (this is actually a generalization of the original idea of O'Gorman and Clowes):

$$v(x, y) = \left| \vec{\nabla}(x, y) \circ \vec{n}(x, y) \right| \tag{4}$$

where $\vec{\nabla}(x, y)$ is the image gradient and $\vec{n}(x, y)$ is the unit normal vector of a given instance of fitted curves (see Fig. 2).

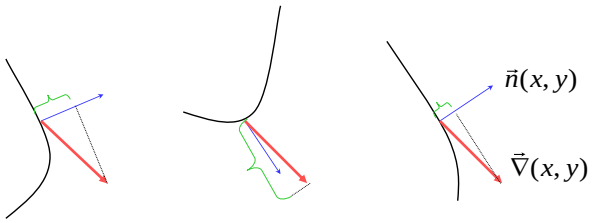

Figure 2. Diversified pixel contributions depending on the angle between the normal vector (blue arrows) and the image gradient vector (red arrows).

In case of junction fitting, the normal vectors are always orthogonal to the window's radii (see Fig. 3) so that the formula for calculating *GWHT* (based on Eqs 3 and 4) is:

$$GWHT(\theta) = \int_0^R \left| \nabla_x(\theta, r) \cdot \cos(\theta + \pi/2) + \nabla_y(\theta, r) \cdot \sin(\theta + \pi/2) \right| dr = \\ = \int_0^R \left| -\nabla_x(\theta, r) \cdot \sin\theta + \nabla_y(\theta, r) \cdot \cos\theta \right| dr \quad (5)$$

It can be noticed that the above definition of the pixel votes is equivalent to the *radial energy* of pixels used in Sinzinger, 2008 (he additionally rejects contributions of pixels with gradient vectors deflected by less than 45° from the radial directions).

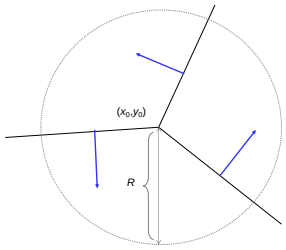

Figure 3. Normal vectors of an exemplary junction.

In case of junction fitting, the computational complexities of *HT* and *GWHT* are very similar (although in Eq.3 we use a binary voting function *v*, we still need to calculate image gradients to identify contour pixels) but *GWHT* performs much better in case of images of natural quality (e.g. textured, noised, etc.). As an illustration, Fig. 4 shows a circular image (with a clearly visible corner over a textured background) and the corresponding 1D profiles of both *HT* and *GWHT*.

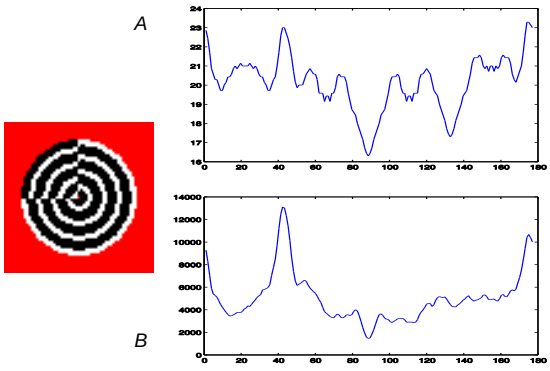

Figure 4. An exemplary circular image (25-pixel radius) and its HT and GWHT profiles (2° wide bins). Contour pixels contributing to the *HT* profile (A) are extracted by Sobel detector (the gradient strenght is thresholded at 50). However, all image pixels contribute to the *GWHT* profile (B), i.e. there is no pre-detection of contour pixels.

In the *HT* profile (Fig. 4A) two most prominent maxima accurately correspond to the orientations of corner segments, but there are also several spurious maxima of comparable heights. They are produced by a number of contour points accidentally located along the same radius (even though their gradient vectors have different orientations). In the *GWHT* profile (Fig. 4B) the same two maxima exist, but there is no other maximum of a similar prominence. Thus, a simple thresholding can be used to reliably detect all actual segments of the junction existing in the image.

Our tests on images of diversified types have shown, nevertheless, that the *GWHT*-based approach is not fully satisfactory. In particular, the maxima magnitudes in the *GWHT* profiles depend not only on the actual visual quality of the corresponding segments but also on how strongly the image is contrasted along these segments (some examples of this effect can be seen in [21]). It is even possible that small highly-contrasted intrusions can produce stronger maxima in *GWHT* profiles than the actual segments (an example in Fig. 5A).

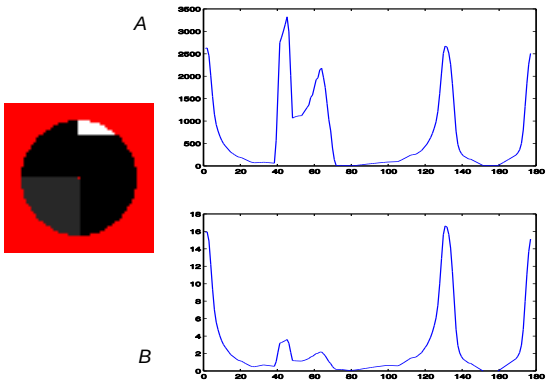

Figure 5. An exemplary circular image (25-pixel radius) and its *GWHT* (A) and *TGWHT* (B) profiles (bins are 2° wide). The gradient threshold for the *TGWHT* profile is 50 (using the Sobel gradient estimates).

Thus, we eventually propose a technique (which can be referred to as *thresholded gradient-weighted Hough transform*, *TGWHT*) where the values casted by individual pixels still depend, similarly to *GWHT*, on the gradient directions but the gradient vectors are normalized to the unit vector (if the norm of gradient exceeds a pre-defined threshold). It is, therefore, a mixture of the *HT* and *GWHT* approaches. *TGWHT* can be calculated using the following modification of Eq. 5:

$$TGWHT(\theta) = \int_0^R \varepsilon \left| -\cos\alpha \sin\theta + \sin\alpha \cos\theta \right| dr \quad (6)$$

where α is the orientation angle of the gradient vector and ε is either **1**, if the gradient magnitude exceeds the threshold, and if α differs from θ by at least $\pi/4$ (similarly to the Sinzinger's approach) or **0** otherwise.

The *TGWHT* profile shown in Fig. 5B is more accurate than the *GHWT* profile of Fig. 5A. The intruding fragment produces only a minor local maximum, while two segments of the actual corner produce the most prominent maxima (even though contrasts along the corner edges are rather weak).

B. Detection of radial and diameter segments

Junctions within circular windows actually consist of two types of lines: radial segments (spanning from the junction

point to the window boundary) and diameter segments (intersecting the junction point and spanning across the whole window). Corners are composed of two radial segments, *T*-junctions contain one radial and one diameter segment, *X*-junctions consist of two diameter segments, *Y*-junctions have three radial segments, etc (see Fig. 6). Even though it can be argued that a diameter segment is a union of two radial segments of the opposite directions, we should not ignore differences between them. For example, corners are junctions of two radial segments, but an infinite number of corners would be localized along any straight line if we do not differentiate between radial and diameter segments (this is a typical problem of segment-based corner detectors where additional mechanisms are needed to eliminate "near-180° corners").

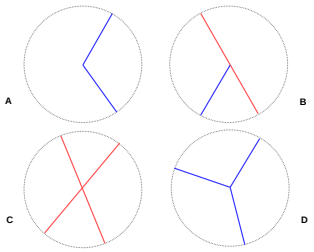

Figure 6. Decomposition of various types of junctions into radial segments (blue) and diameter segments (red).

Therefore, we propose two modifications of the above Hough transforms so that radial and diameter segments could be distinctively detected.

A radial segment of θ_S orientation would produce a local maximum of $TGWHT(\theta)$ profile (of course $HT(\theta)$ or $GWHT(\theta)$ can be alternatively used) for θ_S, but if the segment does not continue on the opposite side of the junction point, the value of $TGWHT(\theta_S+\pi)$ should be very small. However, if there is a continuation (i.e. the segment is actually a diameter one) the value of $TGWHT(\theta_S+\pi)$ would also be a local maximum (presumably of a similar magnitude). Thus, we propose the following two modifications of *TGWHT*, i.e. $TGWHT_R$ and $TGWHT_D$ transforms (the same modifications apply to *HT* and *GWHT*) for fitting, correspondingly, radial segments and diameter segments into circular windows:

$$TGWHT_R(\theta) = \begin{cases} TGWHT(\theta) - TGWHT(\theta+\pi) \\ \quad \text{if } TGWHT(\theta) > TGWHT(\theta+\pi) \\ 0 \\ \quad \text{otherwise} \end{cases} \quad (7)$$

$$TGWHT_D(\theta) = \min\left(TGWHT(\theta), TGWHT(\theta+\pi)\right). \quad (8)$$

Note that for $TGWHT_D$ the range of θ is <0; π>.

Fig. 7 shows an exemplary image fragment containing a T-junction, and the corresponding profiles of *TGWHT*, $TGWHT_R$ and $TGWHT_D$ transforms. It can be seen that *TGWHT* detects both radial and diameter segment (the latter as a pair of radial segments) while $TGWHT_R$ has only one prominent maximum representing the radial segment, and $TGWHT_D$ has also only maximum corresponding to the diameter segment.

It should be noted that a smooth transition between radial and diameter segments actually exists, i.e. diameter segments produce low-magnitude peaks in the $TGWHT_R$ transform (as shown in Fig. 7B). Sometimes, radial segments can also appear as low magnitude maxima of the $TGWHT_D$ transform.

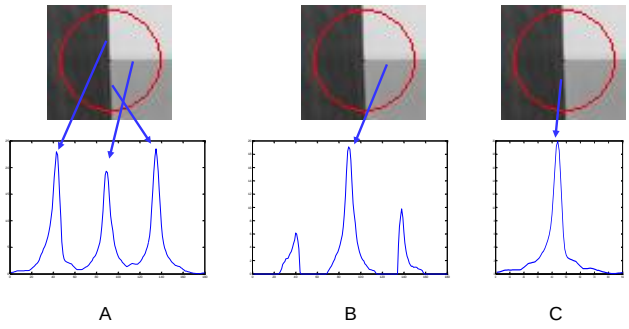

Figure 7. Comparison between *TGWHT* (A), *TGWHTR* (B) and *TGWHTD* (C) transforms of a given circular image of 25-pixel radius. Bins are 2° wide.

We propose to use $TGWHT_D$ and $TGWHT_R$ transforms as the fundamental tools for fitting junctions (i.e. compositions of radial and diameter segments) into given circular images of radius ***R***. More details of junction fitting will be discussed in Section IV.

III. 2D Model of Junction Fitting

If we fit junctions into circular windows of larger diameters, a simple model shown in Fig. 1B might not be accurate. Junction segments can be of diversified lengths (not always reaching the circle's perimeter) so that a 2D parameter space $\Lambda\times\Theta$ is needed to model geometry of junctions in circular windows of radius ***R*** (as shown in Fig. 8).

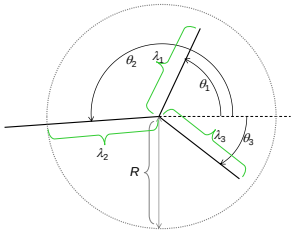

Figure 8. Specification of an exemplary 3-segment junction using a 2D parameter space $\Lambda\times\Theta$.

Therefore, 2D Hough transforms are needed to fit such junctions into the window contents. The transforms are actually very similar to the transforms discussed in Section II. The only difference is that a pixel can only vote for bins with the radii longer than the pixel's distance from the circle origin. Thus, instead Eq. 3 we can use

$$HT(\lambda,\theta) = \int_0^{\lambda} v(\theta, r)dr \quad \text{where} \quad 0 < \lambda \le R \quad (9)$$

and the following expression instead of Eq. 6:

$$TGWHT(\lambda,\theta) = \int_0^{\lambda} \varepsilon \left| -\cos\alpha \sin\theta + \sin\alpha \cos\theta \right| dr \,. \qquad (10)$$

where $0 < \lambda \le R$

Fig. 9 shows a simple synthesized image containing a junction with diversified lengths of segments, and the corresponding *TGWHT* obtained from Eq. 10. The image segments produce monotonically ascending ridges spanning across the whole range of λ (i.e. from 0 to 30 for Fig. 9 image). However, ridges for segments shorter than the window radius saturate at the actual segment's length and are flat for the remaining larger values of λ (this effect can be obviously expected based on Eq. 10). Therefore, the length estimates for such segments by detecting the local maxima of the profiles would be very inaccurate (in particular for images of natural quality when the flat sections of the ridges may have some fluctuations).

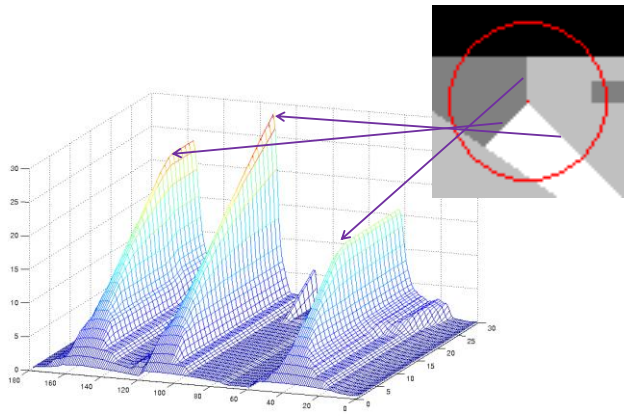

Figure 9. Exemplary circular image (***R*** = 30 pixels) and its 2D *TGWHT* profile. Bins are 2° ×1 pixel.

In order to more accurately localize the maxima we propose to multiply the transform by the exponential damping function. Eventually:

$$TGWHT(\lambda,\theta) = e^{-\lambda/R} \cdot \int_0^{\lambda} \varepsilon \left| -\cos\alpha \sin\theta + \sin\alpha \cos\theta \right| dr \,. \qquad (11)$$

where $0 < \lambda \le R$

After this modification, the maxima of *TGWHT* accurately correspond to orientation and length of segments existing within the image. Fig. 10 can be compared to Fig. 9 to observe the effects of the exponential damping.

The 2D model of junction fitting can be easily adapted for detection of radial and diameter segments (similarly to the 1D model modification outlined in Subsection II.B). However, we envisage that the 2D model would be particularly useful for junction post-processing. After the junction has been fitted into a smaller window by using the 1D, we can apply the 2D model within a larger window to estimate the junction's geometry more accurately. In such cases, the junction lines are already identified as either radial or diameter segments.

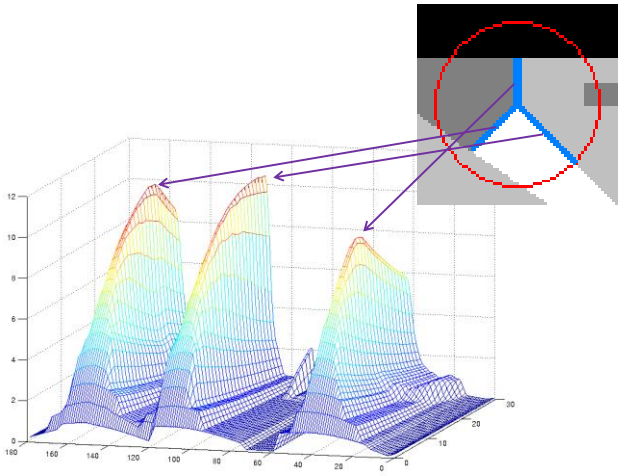

Figure 10. 2D *TGWHT* profile of Fig. 9 image after the exponential damping. The junction segments clearly correspond to profile maxima.

IV. Detection and Localization of Junctions

The exemplary circular images discussed in Sections 2 and 3 actually contain junctions located in the circle origins. However, the proposed Hough transforms can be applied to circular windows of any content. If the window contains a number of high-contrast pixels (i.e. their gradients exceed a threshold) the *TGWHT* profiles (both 1D and 2D) can be built and the profile maxima would define some radial or diameter segments (even if there are no visible junctions there). In particular, if the window is positioned at a location near the actual junction, the junction can be reasonably well fitted into that window. Therefore, we have to discuss the following two problems:

a) Given a *TGWHT* profile for a circular window, determine (based on the maxima of the profiles) if it actually contains a junction and characterize the junction's category (L-junction, X-junction, T-junction, etc.). This is discussed in Subsection IV.A.

b) Assuming that a circular window scans the image, determine locations of the best-fit junctions (non-maxima suppressing discussed in Subsection IV.B).

Compared to other techniques designed for the same tasks (e.g., [19] or [21]) our aim is to reduce the computational complexity to the level of typical Hough transform algorithms. In particular, instead of using optimization techniques, we just adapt the proposed method to varying conditions by using diversified thresholds (similarly to the original Hough transform, where first the gradient is adaptively thresholded to modify the number of contour pixels and, secondly, the local maxima are adaptively thresholded to determine the number of fitted lines/curves, see [7]).

Moreover, to reduce the complexity, we consider basically only the 1D approach (more details in Subection IV.C).

A. Local detection and classification of junctions

Any gradient-based edge detector can be used to detect contour pixels (see a survey by Ziou and Tabbione in [26]) but we arbitrarily decide to use a 3×3 Sobel operator. The conducted tests have indicated that in most cases the Sobel gradient magnitude should be thresholded at 50-100 level. Exemplary results confirming this is a reasonable selection of the threshold are given in Fig. 11. Obviously, other thresholds and/or other edge detectors can be alternatively applied, depending on constraints and characteristics of the problem.

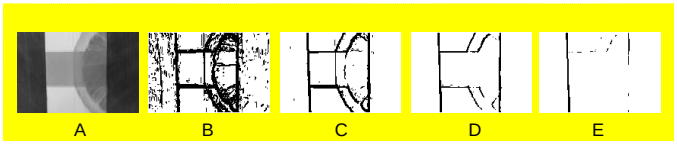

Figure 11. Exemplary image (A) and its contours extracted by Sobel detector. The gradient is thresholded at 20 (B), 50 (C), 100 (D) and 200 (E).

Subsequently, we determine a suitable threshold for the local maxima of $TGWHT_R$ ($TGWHT_D$) profiles to identify genuine segments (radial or diameter). From Eqs 6, 7 and 8, it can be concluded that the maximum value of the transform is limited by the window radius $\boldsymbol{R}$. Therefore, we define the threshold as a percentage of $\boldsymbol{R}$.

The top row of Fig. 12 shows a few images (of 25 pixel radius) from a sequence of image with gradually deteriorating prominence of a vertical edge. Below, the profile is given showing how the magnitude of the local maximum of *TGWHT* changes for the whole sequence of images. The subjective impression of a "vertical edgeness" fades somewhere between images Fig. 12E and Fig. 12F, where the maximum falls below 12 (i.e. approx. 45% of the radius $\boldsymbol{R}$).

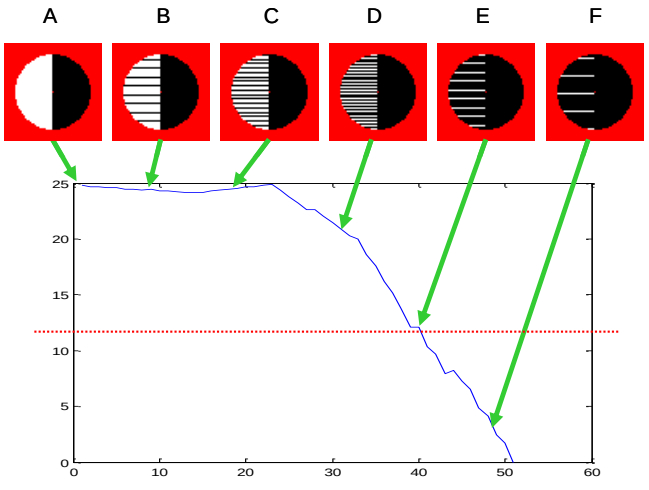

Figure 12. Fluctuations of the *TGWHT* maximum of the vertical segment for a sequence of gradually degraded images (A-F).

Similar results have been obtained for other types of gradually degraded images so that it has been decided to interpret local maxima of $TGWHT_R$ profiles as actual radial segments, if the maximum exceeds at least 40% of the window's radius $\boldsymbol{R}$ (in some cases higher values are recommended, see Subection V.A). Such a value provides detection of almost all visually detectable radial segments.

In case of $TGWHT_D$ profiles, the approach is the same. However, the $TGWHT_D$ profiles are obtained as the minima of two *TGWHT* profiles of the opposite orientation (see Eq. 8) so that their magnitudes are generally lower that for $TGWHT_R$. Therefore, we recommend a lower threshold (e.g. at least 35%) for extracting diameter segments.

Altogether, the algorithm of detecting and classifying a junction located in a circular window of radius $\boldsymbol{R}$ can be summarized as follows:

Algorithm A

Step 1: Detect contour pixels by thresholding outputs of a gradient-based detector (Sobel detector with the threshold Th_1 between 50-100 is used).

Step 2: Build 1D profiles (histograms) of $TGWHT_R$ and $TGWHT_D$ for the window content. Detect the local maxima (winning bins) in both profiles.

Step 3: Local maxima of the $TGWHT_R$ profile exceeding the threshold Th_R (the proposed value is at least $\mathbf{0.4{\times}}\boldsymbol{R}$) are recognized as radial segments and local maxima of the $TGWHT_D$ profile that exceed the threshold Th_D (the proposed value is at least $\mathbf{0.35{\times}}\boldsymbol{R}$) are recognized as diameter segments. The contents of the selected winning bins are used as *strength* of the corresponding segments.

alternatively (if *n* radial and *m* diagonal segments must be detected)

Step 3*A*: Identify *n* top local maxima of the $TGWHT_R$ profile as radial segments, and *m* local maxima of the $TGWHT_D$ profile as diameter segments. The contents of the selected winning bins are used as *strength* of the corresponding segments.

Step 4: Based on the results of Step 3 (or Step 3A) fit the requested types of junctions. For example, the strongest radial segment and the strongest diameter segment (if existing) define the fitted *T*-junction, the strongest three radial segments (if detected) define the fitted *Y*-junction, etc. (additional constraints can be added). The *average weighted strength* of the contributing segments defines the *strength* of the fitted junction. Experimentally established weights are **1.0** and **1.2** for radial and diameter, correspondingly.

As an illustration, a circular image of 25-pixel radius (a fragment of Fig. 11A image) is shown in Fig. 13 with its $TGWHT_R$ and $TGWHT_D$ profiles. The corresponding fitted junctions are given in Fig. 14. It can be seen that all fitted junctions can be detected by a human vision. Based on the strengths of various junctions (given in the Fig. 14 caption) we can conclude that the *T*-junction is the best fit for this window. The majority of a random group of students asked to identify the junction seen in this image also selected *T*-junction.

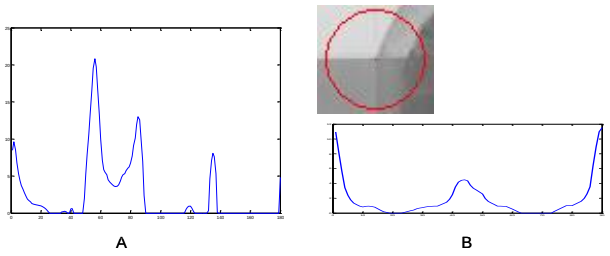

Figure 13. A circular image and its *TGWHTR* (A) and *TGWHTD* (B) profiles.

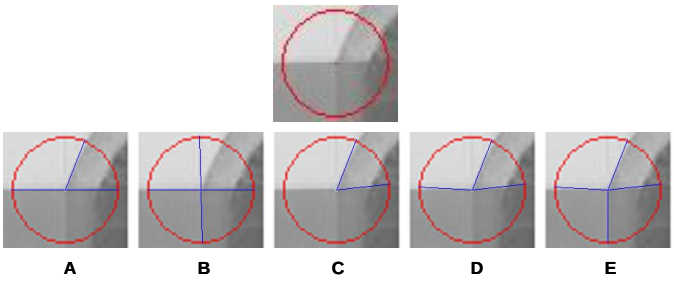

Figure 14. The best-fit junctions and their strengths for Fig. 13 image. (A) *T*-junction (*s*=19.70), (B) *X*-junction (*s*=12.15), (C) corner (*s*=16.92), (D) *Y*-junction (*s*=14.48) and (E) 4-segment junction (*s*=12.88).

B. Global detection and localization of junctions

Algorithm A can be applied to any circular window with a number of contour points extracted. Thus, a junction can be fitted into the window even if the actual junction is not exactly at the window's location or (especially when *Step* 3*A* of the algorithm is used) and even if there is no visible junction at all. Therefore, when *Algorithm A* is applied to contents of a window scanning an image, another method is needed to detect and accurately localize actual junctions, and to reject junctions that are not prominent enough.

The proposed approach is a simple detection of local maxima of the junction *strength* function (if a maximum is below a predefined threshold, which is particularly possible if Step 3A is used, it would be rejected). Formally, the junction detection and localization within a scanned image can be specified as follows:

Algorithm B

Step I: For the selected junction type ***J*** (e.g. *T*-junction, corner, *Y*-junction, etc.) apply *Algorithm A* to fit a junction to each location (x, y) of the scanning window; determine the corresponding junction strength $s_J(x, y)$.

Step II: Preliminarily locate a ***J***–type junction at any coordinates (x_M, y_M) where the junction strength s_J reaches a local maximum, i.e.

$$s_J(x_M, y_M) = \max\{s_J(x, y) : (x, y) \in U(x_M, y_M)\} \quad (12)$$

where $U(x_M, y_M)$ is a neighbourhood of (x_M, y_M).

Step III: Delete a preliminarily located junction if its strength is below a threshold value Th_J.

Step III is particularly important if *Step* 3*A* is used in the preceding *Algorithm A*. If *Step* 3 is applied, all fitted segments are sufficiently prominent and, subsequently, all junctions preliminarily located in *Step* II should be prominent as well. Nevertheless, *Step* III can always be run to reject junctions of poorer quality.

Fig. 15 shows exemplary results obtained by *Algorithm B* in Fig. 11 image. The most prominent *T*-junctions and *X*-junctions are superimposed over the original image. Some of these junctions can be easily identified visually (their *strengths* are higher) while the other ones can be eventually noticed only after a careful inspection of the image (these are junctions with lower *strengths*). The corresponding map of the *T*-junction strength s_T is given in Fig. 16.

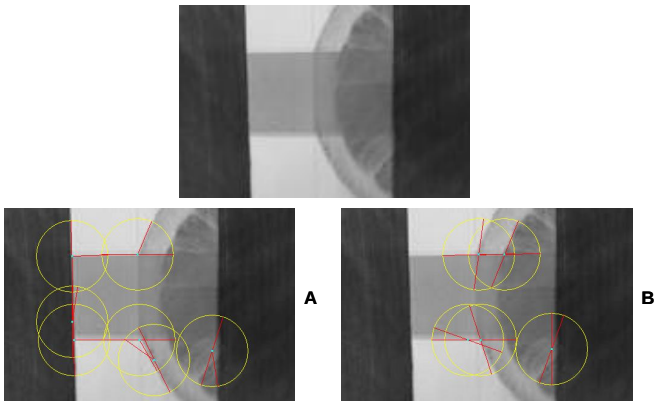

Figure 15. *T*-junctions (A) and *X*-junctions (B) located in an exemplary image. Scanning windows of 20-pixel radius are used.

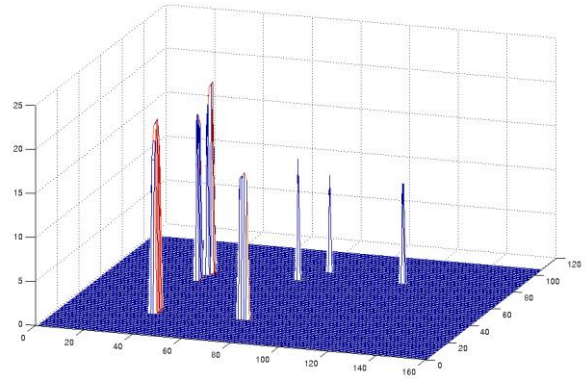

Figure 16. The map of the *T*-junction strength function s_T for Fig. 15A. The detected junctions correspond to the function spikes.

In Fig. 15, an image of a relatively poor quality and fuzzy content has been deliberately selected. For images of clear contents (even if they are distorted by typical effects, e.g. blurring) the results usually fully correspond to the visual impression. This is illustrated in Fig. 17 which shows challenging fragments of images used to benchmark corner and junction detectors (e.g., by Smith and Brady in [22] and by Sinzinger in [21]). Even though some junctions are replicated

(because of heavily blurred edges) the localization and geometry of the detected junctions are very good.

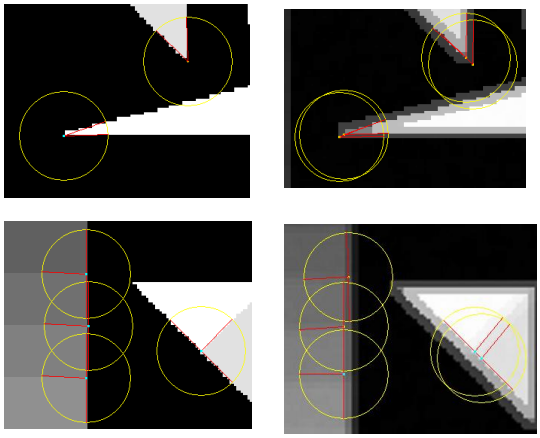

Figure 17. Results of corner (top row) and *T*-junction (bottom row)detection for selected fragments of benchmark images. The scanning window of 20-pixel radius is used.

C. Junction description in 2D parameter space

Section III presented a 2D model of junctions (and the corresponding 2D Hough transform). Such a model can be used as an alternative approach to junction detection and localization. However, we consider it impractical because:

- ✓ The size of scanning window should be limited to avoid excessively high computational complexity. We typically use windows of 20-pixel radius (or even smaller).
- ✓ If the scanning window is small, most visually meaningful junctions would stretch to the window's border (i.e. the segment length is always equal to the window radius and the 1D model is sufficient).

However, after a junction of a particular category has been localized, it might be further analyzed in a wider context (i.e. within a window of a longer radius). We can, therefore, rescan the image near the junction's location using a larger window only and apply the 2D model of junction fitting. The results could be used: (a) to correct inaccuracies of the junction fitting within smaller windows and (b) to obtain the full geometry of the junction (including the lengths of the junction's segments).

In such an operation the junction category is already known so that we do not have to distinguish between radial and diameter segments; an ordinary 2D profile of *TGWHT* (Eq. 11) provides all necessary information.

As an illustrative example of this concept, we consider one of *T*-junctions identified (using a 20-pixel window) in Fig. 15A. Fig. 18 shows the same image fragment with the originally reconstructed *T*-junction and the junction reconstructed by the 2D model 2D model (the *TGWHT* profile is calculated from the window of 30-pixel radius). His example also shows that within the 30-pixel window another model of T-junction would be actually built (there are two profile maxima higher than the maximum corresponding to the originally radial segment (which is shorter so that its contribution to 30-pixel window profile is limited).

Nevertheless, we do not change the model and, instead, determine its 2D characteristics.

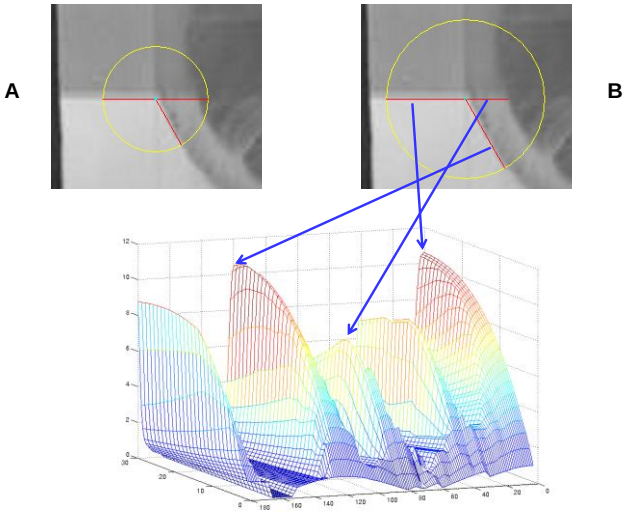

Figure 18. A *T*-junction found by *Algorithm B* using windows of 20-pixel radius (A) and the same junction fitted by using 2D *TGWHT* in a window of 30-pixel radius (B). The profile of *TGWHT* shows the maxima representing segments of the original junction.

V. DISCUSSION

A. Implementation issues

The proposed technique follows the principles of the Hough transforms. Therefore, the properties (both advantages and drawbacks) of Hough transforms are, in general, inherited. In particular, the expected results strongly depend on the selection of thresholds and fixed parameters. The following thresholds/parameters are set in the method:

a) **Radius *R* of the scanning window**. This is the only parameter that does not exist in the original Hough transforms. It determines the sense of "locality" and obviously it renders the method not invariant to image rescaling. However, in junction detection scale is usually of secondary significance (junctions can be similarly detected within a wide range of scales). Moreover, when the 2D model of junction fitting (see Section III and Subsection IV.C) is applied, the lengths of junction segments are determined so that the image scale is (at least partially) eventually reflected in the.

b) **Gradient magnitude threshold Th_1** (see Fig. 11 and *Algorithm A*) determines the set of contour pixels. Lower thresholds mean that less contrasted image fragments can contribute to junction fitting.

c) **Segment strength thresholds Th_R and Th_D** (used in *Algorithm A*). They determine the visual prominence of the locally fitted junctions. Alternatively, the number of detected segments can be used and the strongest segments are always fitted (regardless their strength values).

d) **Junction strength threshold Th_J** (used in *Algorithm B*) plays a supplementary role. We can apply it to delete too weak junctions.

Relations between the gradient threshold Th_1 and the segment strength thresholds (Th_R and Th_D) are critical for the effective use of the method. In general, a low gradient threshold should be combined with higher segment thresholds and another way around. If all thresholds are low, too many junctions can be detected (most of them hardly visible). If all thresholds are high, some visually perceivable junctions might be missed. Exemplary results for *T*-junction detection (i.e. the junction strength maps s_T – their maxima define detected junctions) are given in Fig. 19.

The computational complexity is, in general, comparable to the complexity of other algorithms based on the Hough transforms, i.e. it depends on the image size and the parameter space dimensionality. Thus, the complexity of *Algorithm A* for a single window is $O(k \times R^2)$ where k is the number of orientation bins and R is the window radius, while for $n \times m$ images the combined complexity of *Algorithms A* and *B* is $O(n \times m \times k \times R^2)$. It can be reduced by using a grid of window locations (instead of scanning the image with 1-pixel increments).

The method has been originally coded in Matlab and later converted into Java executables. Thus, the execution time incorporates the communication/data transmission overhead. For images of the resolution ranging from 320×240 to 1024×768 the typical execution times are within 1sec (excluding communication overheads if the application is run on a server) to detect one type of junctions. If more types of junctions are simultaneously detected, the timing deteriorates only insignificantly because results of the most time-consuming operations (i.e. segment fitting within the scanning window) are almost 100% reusable for other types of junctions.

B. Junctions as interest points

Historically, corner detectors (note that simple corner detectors do not distinguish between corners and other junctions) were used for image matching in vision-based navigation (e.g. Moravec in [17]). In the following years, image matching techniques evolved and currently keypoints (interest points) extracted by more sophisticated detectors dominate. However, there is still a strong interest in using junctions as image-matching features because they provide some visual semantics to their neighbourhoods. Several papers suggest to apply junction fitting (or fitting other geometric features) to the results of keypoint detection (e.g., [21] and [23]). It should be noted, however, that the most popular and effective keypoint detectors (see a survey by Mikolajczyk in [16]) very seldom localize keypoints at the actual junctions.

Examples given in Fig. 20 indicate that keypoints are often detected at a certain distance (proportional to the scale of a keypoint) from actual junctions. Therefore, junction fitting must be attempted within areas surrounding the keypoints. Otherwise the characteristics of fitted junctions would be inconsistent with the actual image contents.

C. Concluding remarks

The paper presents a novel method of junction detection in images (including images of low quality). The method is based on a simple model of the Hough transforms. Although fundamentals are similar to the techniques previously reported, several novel contributions can be listed:

- ✓ The local (attached to the window centre) Hough transform are applied so that the parameter space needs only 1 or 2 dimensions for fitting junctions of any type.
- ✓ The Hough transforms are modified to allow a distinctive fitting of radial and diameter segments. Subsequently, junctions of various types can be distinctively detected and localized using the same mechanism.
- ✓ 2D Hough transforms allow a full description of junction's geometry (including the segment lengths) so that scale invariance (within the size of scanning windows) can be achieved.

The method is structured into two algorithms (one for the local junction fitting and the other one for the global junction localization); both are based on a simple concept of local maximum detection.

The primary intended application area is CBVIR (content-based visual information retrieval, see a survey by Kherfi et al. in [11]) where junctions would be used as features for a semantics-based image matching.

ACKNOWLEDGMENT

The author thanks Dr Mariusz Paradowski for his assistance in converting Matlab source codes into Java executables.

REFERENCES

[1] Baker, S. and Nayar, S.K., 1998, Parametric Feature Detection, Int. J. Comput. Vision 27(1), 27–50.

[2] Barrett, W.A. and Petersen, K.D., 2001, Houghing the Hough: Peak collection for detection of corners, junctions and line intersections, in: Proc, IEEE Conf. Comp. Vision & Pattern Rec. 2, 302–309.

[3] Copper, M.C., 2001, The interpretation of line drawings with contrast failure and shadows, Int. J. Comput. Vision 43(2), 75–97.

[4] Cucchiara, R. and Filicori, F., 1998, The vector-gradient Hough transform. IEEE Trans. PAMI 20(7), 746–750.

[5] Deriche, R. and Giraudon, G., 1993, A computational approach for corner and vertex detection, Int. J. Comput. Vision 10(2), 101–124.

[6] Deschenes, F. and Ziou, D., 2000, Detection of line junctions and line terminations using curvilinear features, Pattern Recognition Letters 21, 637–649

[7] Duda, R.O. and Hart, P.E., 1972, Use of the Hough transformation to detect lines and curves in pictures, Comm. ACM 15, 11–15.

[8] Faas, F.G.A. and van Vliet, L.J., 2007, Junction detection and multi-orientation analysis using streamlines, LNCS 4673, 718–725.

[9] Freeman, H., 1974, Computer processing of line-drawing images, Computing Surveys **7**(1), 57–97.

[10] Harris, C. and Stephens, M., 1988, A combined corner and edge detector, in: Proc. 4th Alvey Vision Conference, 147–151.

[11] Kherfi, M.L. , Ziou, D. and Bernardi, A., 2004, Image retrieval from the world wide web: issues, techniques, and systems, ACM Comput. Surv 36 (1), 35–67.

[12] Laganiere, R. and Elias. R., 2004, The detection of junction features in images, in: Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Proc. 3, 573–576.

[13] Lindeberg, T., 1994, Junction detection with automatic selection of detection scales and localization scales, in: Proc. 1st Int. Conf. Image Proc. 1, 924–928.

[14] Mahadevan, S. and Casasent, D., 2002, Detection of sample parameters in secondary electron microscope images (test results), In: Proc. 8th Conf. Hybrid Image and Signal, 104–112.

[15] Matas, J. and Kitler, J., 1993, Junction detection using probabilistic relaxation, Image & Vision Computing 11(4), 197–202.

[16] Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. and van Gool, L., 2005, A comparison of affine region detectors, Int. J. Comput. Vision 65 (1–2), 43–72.

[17] Moravec, H., 1980, Obstacle avoidance and navigation in the real world by a seeing robot rover, Tech Report CMU-RI-TR-3, Carnegie Mellon University, Pittsburgh.

[18] O'Gorman, F. and Clowes, M.B., 1976, Finding picture edges through collinearity of feature points, IEEE Trans. Computers 25(4), 449–456.

[19] Parida, L., Geiger, D. and Hummel, R., 1998, Junctions: detection, classification, and reconstruction, IEEE Trans PAMI 20(7), 687–698.

[20] Shen, F. and Wang, H., 2001, A local edge detector used for finding corners, LNCS 2229, 65–71.

[21] Sinzinger, E., 2008, A model-based approach to junction detection using radial energy, Pattern Recognition 41, 494–505.

[22] Smith, S.M. and Brady,J.M., 1997, SUSAN: a new approach to low level image processing, Int. J. Comput. Vision 23 (1), 45–78.

[23] Sluzek, A., 2005, On moment-based local operators for detecting image patterns, Image & Vision Computing 23(3), 287–298

[24] Smith, S.M. and Brady, J.M., 1997, SUSAN: a new approach to low level image processing, Int. J. Comput. Vision 23 (1), 45–78.

[25] Waltz, D.I., 1972 Generating semantic description from drawings of scenes with shadows, AITR-271, MIT, Cambridge.

[26] Ziou, D., Tabbone, S, 1998, Edge detection techniques - an overview, Int. J. Pattern Recognition and Image Analysis 8 (4), 537–559.

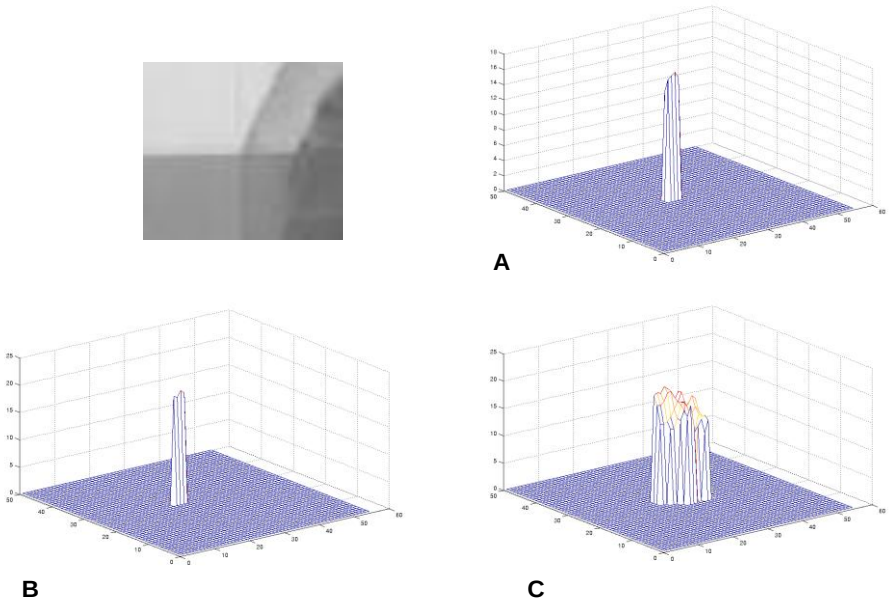

Figure 19. The *strength maps* for *T*-junction detection in an exemplary image. (A) – a higher gradient threshold (Th_1=**100**) and lower segment thresholds (Th_R=Th_D=**0.4×R**); (B) – a lower gradient threshold (Th_1=**20**) and higher segment thresholds (Th_R=Th_D=**0.8×R**); all thresholds are low (Th_1=**20** and Th_R=Th_D=**0.4×R**). When all thresholds are high, no *T*-junction is detected.

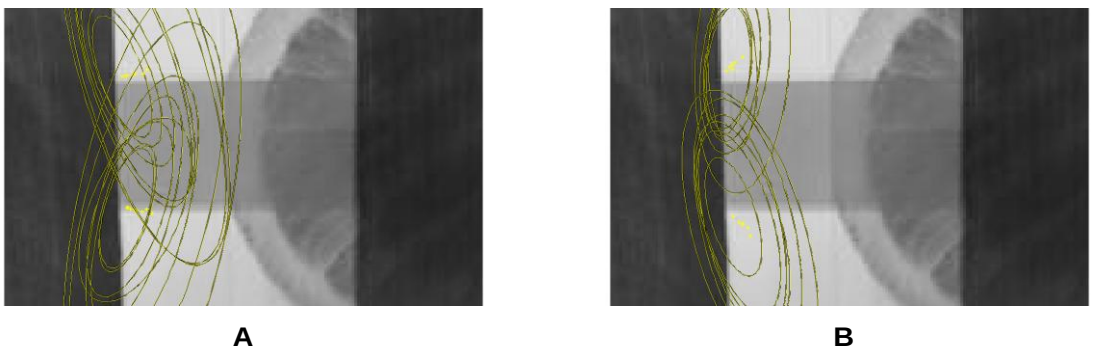

Figure 20. Keypoints detected in the Fig. 15 image by using (A) Harris-Affine detector and (B) Hessian-Affine detector.