2012 10th IAPR International Workshop on Document Analysis Systems

A robust approach for local interest point detection in line-drawing images

The-Anh Pham, Mathieu Delalandre, Sabine Barrat and Jean-Yves Ramel
Laboratoire d'Informatique 64, Avenue Jean Portalis, 37200 Tours - France.
the-anh.pham@etu.univ-tours.fr,{mathieu.delalandre, sabine.barrat, ramel}@univ-tours.fr

***Abstract*—In this paper, we propose a new method to detect local interest points as junctions in line-drawing images. Our approach takes advantages of different aspects. Firstly, we extract skeleton of image and then construct a *Skeleton Connective Graph* with the expectation that it provides a first level of junction detection from shapes. Secondly, instead of employing low-level operators to detect junctions as described in many traditional techniques, our method works at path level taking different skeleton branches into account to gain robustness. Thirdly, we exploit the benefits of wavelet transform (e.g. multi-resolution analysis, discontinuity detection, fast computation, less sensitive to noises) to efficiently detect the dominant points from 1D representations of the paths. Finally, a post-process of pruning and connecting the skeleton segments is performed to discard false detected points and to refine the skeleton. We present in experiments interesting results compared to different methods.**

***Keywords*-Interest point detection, junction point detection, multi-resolution analysis, document image analysis.**

I. Introduction

A *local interest point* or *keypoint* is an image point which differs from its immediate neighbors in terms of some specific image properties such as intensity, color, orientation, texture, curvature, etc. Keypoint detection is one the most important topics from computer vision field since they are regarded as useful features to address the problems of image matching, object recognition, etc. For that reason, lots of work has been dedicated in literature to detect different types of keypoints. Some examples of well-known keypoint detectors are DOG, LOG, Harris-Corner, SURF, SUSAN, etc [7]. Nevertheless, these techniques are not well adapted to the field of document image analysis (DIA) due to several important issues. At the first place, since the popular keypoint detectors (SURF, DOG, LOG, etc.) and the corresponding descriptors focus on describing the local structures of images based on intensities, orientation and location histogram, their performance is likely to be reduced when facing to bi-level document images. On the other place, the detectors of local features such as blobs, corners, edges, Haars, etc., tend to create a large number of keypoints and this is particularly unsuitable for document images which contain various complex objects, for example, texts, graphics, equations, seals, logos, tables, charts, etc.

In general, there are two stream lines of work addressing the issue of keypoint detection in DIA. In the first place, several works as in [1] try to adapt the techniques from computer vision field to detect local interest points in document images. Results represented in these works highlight weak precision rates, not suitable for document image content. On the other hand, the works in [10], [11], [12], [13], [14], [15] were proposed to focus on detecting some specific types of local interesting points in DIA such as junctions, corners, high curvature points, etc.

In [10], the author first proposed a classical low-level technique to detect ended-points and fork-points from skeleton of image by introducing the definition of connectivity as the number of transitions from foreground point to background point and vice versa in counterclockwise order. This technique was then widely applied in many other applications.

In [11], the author constructed a tensor voting framework to detect ended-points, T- and L- junctions. Each token or pixel is supposed to support for its neighbors within a local radius in the means of first and second order votes computed based on its proximity, colinearity and cocurvilinearity. The power of tensor voting is then used to label ended-points, T- and L- junctions. This method is scale-dependent, low-level processing and thus is sensitive to noises. Another low-level operator based approach for ended-points and junctions detection was presented in [12] by using template based matching which is sensitive to artifacts introduced during the thinning process. A post-processing of barb removing is performed based on morphological spurring operation in which the optimum number of iterations is determined based on the average line thickness.

In [13], the author presented a method to detect junctions from contours of image by combining local operators and matching. The local operator is applied to every edge point to compute its 1D profile within a circular window of size R ($15 < R < 30$). At junction location, the 1D profile presents several spikes and the position of the spikes indicate the geometry of the junction which is used to either discard or select the junction by matching it with template ones. This method is subjected to the limited range of scales and the threshold of selecting the number of spikes.

In [14], [15], the authors applied wavelet transform to detect corners from contours of image. At first, the contour is represented in the means of 1D signal based on an orientation function. Then, the signals are decomposed at different scales by using wavelet transform and the corners are identified at some levels of decomposition representation. Their results are shown to be much better than those in the

978-0-7695-4661-2/12 $26.00 © 2012 IEEE
DOI 10.1109/DAS.2012.9

well-known dominant point detectors presented in [8]. Nevertheless, both these methods have to cope with a problem of how to combine the corners detected from different contours at crossing locations and ended-segments. In addition, their 1D representations are sensitive to noises because of using a fixed threshold.

In this paper, we aim at constructing a robust technique for interest point detection as junctions in line-drawing images in the senses that it is fast computation, less sensitive to noises and less dependent on the number of predefined thresholds. In addition, we justify our approach by comparing our results on some standard datasets with well-known junction detectors in literature. To conduct our characteristics, we take advantages of different aspects. Firstly, we extract skeleton of image and then construct the Skeleton Connective Graph (SCG) which provides a natural way to detect junctions. Secondly, we define a path of the SCG taking different skeleton branches into account and present how to extract the paths from the SCG. Thirdly, we perform a detail analysis about different methods for constructing 1D representations of 2D plane curves and then derive a good 1D representation for each path which satisfies three desirable properties: translation-, rotation- and shift-invariant. Fourthly, instead of employing low-level operators to detect the junctions, our method looks for the most dominant points of the paths by applying multi-resolution wavelet analysis to efficiently identify the junctions from 1D representations. Finally, a post-process of refining the results is performed at two levels: single-path and inter-paths.

The rest of this paper is organized as follows. The main stages of our approach are presented in section II. Experimental results are presented in section III. We give some key conclusions, perspective works and several potential applications of the proposed method in section IV.

II. Our approach

In Figure 1, we briefly present our approach consisting of five main stages: pre-processing (section A), path definition and extraction (section B), path representation (section C), dominant point detection based on wavelet analysis (section D) and post-processing (section E). We are now going to detail all these stages in the following sections.

A. Pre-processing stage

This state consists of the following processes: skeletonization, branch extraction and branch linking. Our method is to be applied on binary images. These images could be obtained following some enhancement processes such as noise filtering, binarization, etc., depending on specific applications. Then, the skeleton of input image is extracted based on the technique of G.S. di Baja [2] because of many its advantanges as presented in [9]. Starting from the skeleton chaining, we present here some basic definitions.

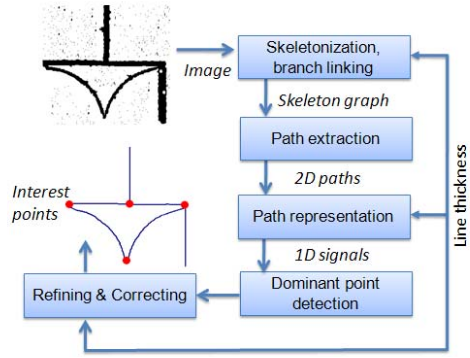

Figure 1. The flow-work of our approach

A *skeleton branch* is a sequence of consecutive and distinct skeleton points such that the first and the last points are either an *ended-point* or a *crossing-point* which are defined as the skeleton points that have only one or at least three 8-connected neighbors, respectively. A skeleton branch is defined as a *long* (respectively, *short*) one if its length is greater than (respectively, less than) the average line thickness from all the skeleton points in the branch.

A *closed curve* is a sequence of consecutive skeleton points in which all points are distinct except the first and last ones.

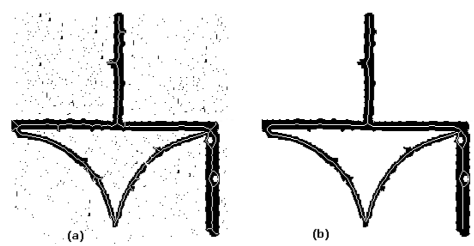

Figure 2. (a) Some common case sceneries of noises, (b) the skeleton chaining after post-processing.

The branches could be simply extracted by using the technique in [6] and then we perform a step of branch linking to reconnect some disjointed branches together (e.g. due to noises). The criterion for linking two disjointed branches is based on the line thickness. Furthermore, a process of skeleton smoothing by employing a mean circular filter is carried out to obtain nice skeleton in addition to the processes of hole filling, barb removing and beautifying as presented in [2]. Figure 2a presents the skeleton of an input image with some common case sceneries of noises and Figure 2b presents the skeleton chaining obtained after pre-processing.

The skeleton chaining is then modeled as a *Skeleton Connective Graph* (*SCG*) which has following properties:

- Its nodes are ended-points or crossing-points. Each node is marked as either *ended-node* or *crossing-node*, respectively.

- An edge connects two nodes if there is a skeleton branch associated to these nodes.
- A closed curve is considered as an edge connecting one node (i.e. an arbitrary skeleton point of this curve) to itself. This node is referred as a *dummy-node* of *SCG*.

Figure 3a presents a *SCG* for the skeleton chaining in Figure 2b. It is a notable point that the image could contain objects at different locations resulting in a graph consisting of different connected components.

B. Path extraction

As presented in the Introduction section, most of the methods for junction detection in literature employed low-level operators. Several methods are applied to contours of image. The contour-based methods need to cope with the issue of how to combine junctions detected from different contours at crossing locations as well as ended-segments. The low-level processing based methods are sensitive to noises and artifacts. To gain robustness, we work at neither contours nor low-level operators. Instead, we treat our approach at context level by defining a path of the *SCG* and analyze the paths at different natural resolutions based on wavelet decomposition representation. Particularly, a path is defined as a sequence of consecutive edges of the *SCG* and the length of the path is the number of edges traversed. We consider three types of paths which are *stroke-*, *circuit-* or *hybrid-path*. These paths are defined as follows:

- A path P is a stroke if the first and the last nodes of P are two ended-nodes and all edges associated to any one from the nodes of P remain the same side in the spatial plane.
- P is a circuit if its first node is identical to its last node and P does not contain any other smaller circuits.
- P is a hybrid-path if it exactly consists of one dummy-node with (or without) another ended-node.

These paths could be extracted by adapting the technique *Line Follower* to the nodes of *SCG* [6]. Particularly, starting from each node of SCG, we always clockwise traverse to next node until we stop at either some ended-node or one already traversed node. Figure 3 presents some paths detected from a skeleton graph.

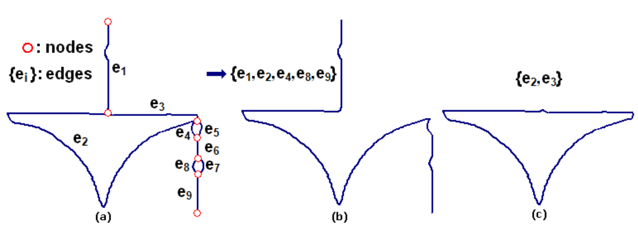

Figure 3. (a) is a *SCG* for the skeleton chaining in Figure 2(b), (b) and (c) are the stroke- and circuit-paths detected from (a), respectively.

C. Path representation

In order to detect dominant points using wavelet analysis, it is necessary to represent 2D paths in the means of 1D signals. In this section, we analyze different methods of constructing 1D representations for 2D plane curves and then derive good 1D representations for our paths. Given a path P consisting of N skeleton points $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$, the author in [5] constructed an 1D representation of P based on a distance function $f(t)$ in which $f(t)$ is defined as Euclidian distance from each point (x_t, y_t) to a reference point. This function was shown to be well adapted to the problem of zero-crossing representation but is not good in the domain of high curvature point detection [5]. Instead, an 1D representation of P should be designed as a curvature function $f(t)$ such that $f(t)$ represents a curvature degree at the point (x_t, y_t). In [14], the authors constructed an 1D representation of a contour in the means of an orientation function $\alpha(t)$ as follows: $\alpha(t) = tan^{-1}((y_{t+q} - y_{t-q})/(x_{t+q} - x_{t-q}))$ where q is fixed at 3. Using this fixed threshold is very sensitive to noises, particularly in the cases of saw-toothed contours. Following our experimentations with a number of different curvature functions as presented in [8], we derived that the following one provides with the most stable results since it avoids the problems caused by discrete derivations and small changes in slope.

$$\vec{a}_{tq} = (x_t - x_{t+q}, y_t - y_{t+q})$$

$$\vec{b}_{tq} = (x_t - x_{t-q}, y_t - y_{t-q})$$

$$f(t) = cos_{tq} = \frac{\vec{a}_{tq} \cdot \vec{b}_{tq}}{|\vec{a}_{tq}||\vec{b}_{tq}|}$$

In our implementation, q is an adaptive threshold which is set based on the line thickness at the skeleton point (x_t, y_t).

The function $f(t)$ is $2D$ translation- and 2D rotation-invariant in image plane, but not shift-invariant (i.e. $1D$ translation: $f(t)$ is dependent on the selection of the starting point). Even though there exists several techniques in wavelet analysis supporting the shift invariant, they are not really shift invariant. In practices, we could miss some true interest points depending on the selection of the starting point. This issue was not explicitly presented in many former approaches ([14], [15]). We present a solution of selecting the starting point to gain the shift-invariant as follows. If the path P contains at least one ended-node, it is simple to show that the starting point could be such a node. On contrary, suppose that P contains several high curvature points and if the starting point is selected as one of them, due to the border distortion effect of the wavelet analysis, we could miss one true positive. We therefore should try to avoid this situation. It is noticed that the function $f(t)$ tends to be a constant (or near constant) at some partition $[t_a, t_b]$ $(1 \leq t_a < t_b < N)$, if and only if, there is no high curvature point in this partition. One such partition presents a straight line segment of P and could be found by employing some basic method. Consequently, the starting point is selected as a middle point in $[t_a, t_b]$.

D. Dominant point detection

Our local interest point detection algorithm is applied to the 1D representation of each path by using multi-resolution discrete stationary wavelet transform (SWT) [3]. We employ SWT to detect the dominant points from 1D signals because it provides many important advantages compared to other traditional high curvature point detection approaches [8]. Firstly, one of the most important applications of wavelet analysis is to detect the exact instant when a signal changes. Secondly, wavelet transform supports multi-resolution analysis in which the number of scales needed to completely decompose the signal is rather small because we only work at *dyadic* scales (i.e. $2^0, 2^1, \ldots, 2^n$), not continues scales as presented in traditional scale-space techniques. Further more, the number of scales could be definitely determined based on the wavelet mother and the length of the signal without using any threshold at all [4]. Finally, there exists many fast wavelet transform algorithms which could be efficiently implemented supporting for our all processes in a linear time complexity.

Let $Wf(s,t)$ is the wavelet transform (using SWT) of $f(t)$ at the scale $s = 2^j$ ($j = 0, 1, \ldots, J-1$) where J is the maximum number of analyzed scales. Since we are interested in detecting high curvature points, *Haar* mother wavelet is selected to decompose the signal. The key idea of the dominant point detection process is that, at the dominant point, (s_0, t_0), the $Wf(s_0, t_0)$ reaches a local maximum over time space and this maximum tends to propagates to another maximum at larger scale ($s = s_0 * 2$), but this property is not hold at finer scale ($s = s_0 * 2^{-1}$). In contrast, the number of false maxima (e.g. noises) tends to be greatly reduced when the scale increases. Based on these characteristics, the steps of this procedure are presents as follows:

- Compute the $Wf(s,t)$ of $f(t)$ using SWT at different scales $s = 2^j$ with $j = 0, \ldots, J-1$.
- Detect the local maxima of $Wf(s,t)$ at each scale $s = 2^j$ with $j = 0, \ldots, J-1$.
- Discard every local maximum $Wf(s_\rho, t_\rho)$ at the scale s_ρ in which there is no local maximum $Wf(s_{\rho'}, t_\rho)$ at the scale $s_{\rho'} = 2 * s_\rho$.

The remaining maxima are regarded as the candidate keypoints. Figure 4a presents the junction points (i.e. small red dots) detected from the input image in Figure 2.

E. Correcting and refining

Let $O = \{o_0, o_1, \ldots, o_K\}$ are the set of junction points detected from some path P in the previous stage. Due to the disturbance of noises, it is inevitable to detect some false

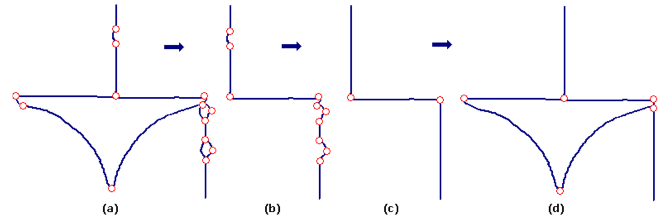

Figure 4. (a) The junction points (small circles) detected from an image in Figure 2, (b) a path P of (a) with the detected keypoints, (c) after correcting and refining, (d) the final skeleton graph with the final keypoints.

junctions. In order to improve the results, we propose a post-process of refining the detected points at two levels: *single-path* and *inter-paths*.

At the single-path level, we consider that the path P is partitioned into M segments by the detected points $o_0, o_1, \ldots, o_K$. Each segment is assigned a property of either *long* or *short* one in the same way as the definitions of long and short branches. Then, we make a reasonable assumption as presented in [9] that the short segments are likely to be the candidates of false skeletons due to noises. Thus, we try to delete all consecutive short segments which are bounded between two long ones. Let $S_{k_0}, S_{k_1}, \ldots, S_{k_L}$ are the consecutive segments obtained from the path P in which all these segments are the short ones except S_{k_0} and S_{k_L} ($k_L < M$), the segments $S_{k_1}, \ldots, S_{k_{L-1}}$ are removed and two the long ones (S_{k_0}, S_{k_L}) are reconnected by extending these line segments as far as they meet at one common point, if and only if, all points of the extented segments are foreground pixels. The keypoints linked to the deleted segments are discarded and the common point is treated as a new keypoint. Each new keypoint is then removed if these exists only two long segments associated with this keypoint such that they nearly construct a straight line. Figure 4 illustrates this process for one path of a *SCG*.

At the inter-paths level, we exploit information of the keypoints detected from different paths to discard false alarms and identify new keypoints as crossing-points based on a voting scheme. The crossing-points could be easily detected from the skeleton but it is not reliable to identify them as keypoints because of noises. Let $P_1, P_2, \ldots, P_m$ are the different paths passing one common crossing-point p, if there exists several keypoints, $q_1, q_2, \ldots, q_h$, detected from these paths such that they are identical to p, we could believe that the crossing-point p is a desirable keypoint. Nevertheless, the keypoints $\{q_1, q_2, \ldots, q_h\}$ in practice are located in a neighborhood of p. To address this issue, each keypoint $q_i \in P_j$ ($1 \leq i \leq h, 1 \leq j \leq m$) is supposed to support for all crossing-points of P_j located in a neighborhood of q_i with the size w_{q_i} (e.g. w_{q_i} is set as the line thickness at q_i in our implementation). If one crossing-point gains much support from the keypoints nearby, it is regarded as new keypoint and all the keypoints supporting for this crossing-point are removed.

III. Experiments

In this section, we evaluate our junction detection method based on *repeatability* criteria as this criteria is a standard one for performance characterization of keypoint detection methods in literature [7]. This criteria works as follows. Given a reference image I_{ref} and a test image I_{test} taken under different transformations (e.g. noising, rotation, scaling) from I_{ref}, the repeatability signifies that local features detected in I_{ref} should be repeated in I_{test} with some small error ϵ in location. Particularly, we denote $D(I_{ref}, I_{test}, \epsilon)$ as the set of points in I_{ref} which are successfully detected in I_{test} in the senses that for each point $p \in D(I_{ref}, I_{test}, \epsilon)$, there exists at least one corresponding point $q \in I_{test}$ such that $distance(p, q) \leq \epsilon$. Let n_r and n_t are the number of keypoints detected by one detector from I_{ref} and I_{test}, respectively. The repeatability score of this detector applied for the pair (I_{ref}, I_{test}) is computed as follows:

$$r(I_{ref}, I_{test}, \epsilon) = \frac{|D(I_{ref}, I_{test}, \epsilon)|}{Mean(n_r, n_t)}$$

In order to obtain a comparative evaluation, we carry out two experiments with a standard Harris-Corner detector [7] and a classical junction detection technique of Rutovitz [10]. We have selected here these two detectors since the Rutovitz technqiue is commonly used in the problem of *fork-point* detection in DIA and the Harris technique detects *corner* points which have a large overlapping with junction points.

In the first experiment, our method and Harris-Corner method are able to detect both types of junctions: *connective*-junction (e.g. *Z*-junction, *L*-junction, *N*-junction, etc.) and *crossing*-junction (e.g. *T*-junction, *K*-junction, *X*-junction, etc.) which are defined as junctions that have only two or at least three 8-connected neighbors, respectively. We refer this evaluation as *full-junction* experiment. In the second experiment, we compare our method with classical junction detector of Rutovitz which is able to detect only crossing-junctions. To make a fair evaluation, a pre-process of noise and artifact removing is carried out before applying the Rutovitz method. In addition, to adapt our method for this comparison, we discard all connective-junctions detected from our method remaining only crossing-junctions. We refer this evaluation as *crossing-junction* experiment.

In both experiments, all detectors are applied on the training datasets #3 and #5 from GREC2011 [1]. These datasets used the same model dataset of 150 clean symbols to generate test images. In the dataset #3, each model symbol is used to generate more 10 test symbols undergone different rotation and scaling resulting in 1500 test symbols. In the dataset #5, each model symbol is used to generate more 25 symbols with various levels of noises resulting in 3750 test symbols. The parameters of rotation, scaling and noise applied for each model symbol in both the datasets are known in groundtruth.

Our strategy to perform evaluation in two experiments is described in Figure 5. This strategy follows the general characterization protocol for keypoint detection as detailed in [7]. We first apply each detector to the model symbols and the test symbols in each dataset to obtain ideal junctions (S_m) and detected junctions (S_t), respectively. Then, we use parameters in groundtruth to compute repeatability scores of this detector from two sets of junctions: S_m and S_t. The overall repeatability score of each detector in each experiment is computed as an average score from the repeatability scores of the detector applied for all model symbols and test symbols in each dataset. We vary the value of parameter ϵ to obtain a *ROC-like* curve of the repeatability score.

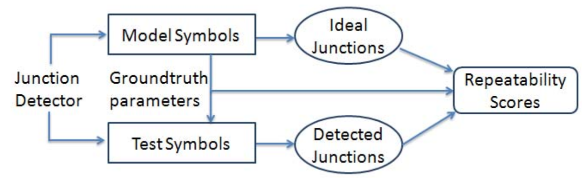

Figure 5. The evaluation strategy applied for each detector.

We present in Figure 6 and Figure 7 the results of each method applied for the datasets #3 and #5 in the full-junction and crossing-junction experiments. As we can see, our method provides interesting results compared to Harris-Corner detector and Rutovitz's crossing-junction detector in both experiments. The mean time of our method, Harris-Corner and Rutovitz's method applied for each symbol are 19.3 ms, 21.5 ms and 16.2 ms, respectively.

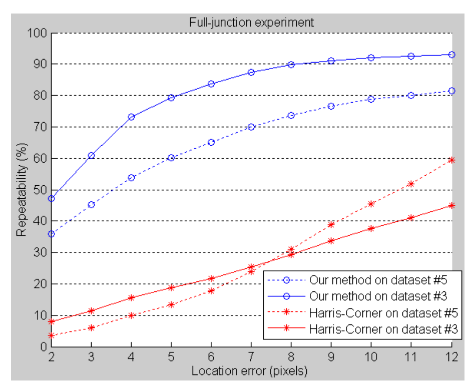

Figure 6. Repeatability score of our method and Harris-Corner detector on datasets #3 and #5 in the full-junction experiment.

IV. Conclusions and Perspectives

In this paper, an approach for junction detection in line-drawing image is proposed. Several major aspects are addressed in our approach resulting in a robust method. At first, we extract skeleton of image and then construct

[1] http://iapr-tc10.univ-lr.fr/index.php/symbol-contest-2011

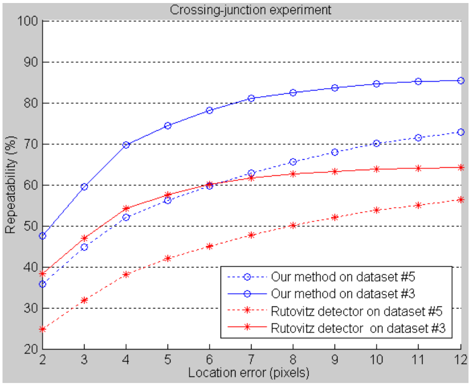

Figure 7. Repeatability score of our method and Rutovitz's junction detector on datasets #3 and #5 in the crossing-junction experiment.

the Skeleton Connective Graph with the expectation that it provides a good way to detect junctions from images. Next, we define a path of the graph taking different branches into account and present how to extract the paths from the graph. The paths are then represented in the forms of 1D signals following a process of dominant point detection by employing the multi-resolution wavelet analysis. A last post-process of pruning and connecting the skeleton segments is performed to discard false detected points and to refine the skeleton. We present in experiments interesting results compared to the Harris-Corner detector and the classical Rutovitz's junction detector.

Following this work, several perspective works are planned. Firstly, more experiments taking recent and advanced techniques of junction detection into account are performed to exactly evaluate the performance of different methods. Secondly, a study on adaptive multi-representation (e.g. a combination of contour and skeleton) rather than single representation (i.e. either skeleton or contour) could significantly improve our results. Finally, some specific applications of the proposed method, vectorization and symbol localization for examples, are likely to be done.

Naturally, vectorization is the very first application of our method with an assumption that input images are supposed to include two kinds of shapes: straight lines and circular arcs. By reconstructing the *SCG* graph from the detected local interest points (i.e. junctions and ended-points), it is noticed that each edge of the SCG is already a smooth segment (otherwise, the proposed method is likely to detect one or more high curvature points on this edge). Therefore, each edge of SCG is vectorized by using some technique such as *linear least squares regression* for fitting the model of line or arc to the edge [9].

In the context of symbol localization, a local primitive is first extracted at each detected junction point. Each local primitive are then described by a very compact and distinctive descriptor at two levels. The level-one descriptor includes basic information such as the number of arms at each junction, the type of each arm, the difference in degree between two consecutive arms, etc. The level-two descriptor could be contructed as a 1D profile of each junction as presented in [13]. The former descriptors are used to do preliminary matching to discard many false candidate matches. The later descriptors are then used to refine the matching results. One such application could be found in [12].

Acknowledgment

This work has been supported by the Viet Nam International Education Development project, numbered 322-VIED.

References

[1] Marco Block et al., *SITT - A Simple Robust Scale invariant Text Feature Detector for Document Mosaicing*. In Proc. of ICIA2009, pp. 400-403.

[2] G.S. di Baja, *Well-Shaped, Stable, and Reversible Skeletons from the (3,4)-Distance Transform*. J. Visual Comm. and Image Representation, vol. 5, no. 1, pp. 107-115, 1994.

[3] Nason G.P et al., *The stationary wavelet transform and some statistical applications*. Lecture Notes in Statistics, pp.281-299.

[4] S. Mallat, *A Wavelet Tour of Signal Processing*. Academic Press, New York, 1999.

[5] M.T. Quang et al., *Recognition of 2D object contours using the wavelet transform zero-crossing representation*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 8, 1997.

[6] Black W. et al., *A general purpose follower for line structured data*. Pattern Recognition 198, vol 14, pp. 33-42.

[7] K. Tuytelaars et al., *Local invariant feature detectors: A survey*. Computer Graphics and Vision, 3:177-280, 2007.

[8] Cho-huak Teh et al., *On the detection of dominant points on digital curves*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 11, no. 8, 1989.

[9] X. Hilaire and K. Tombre, *Robust and Accurate Vectorization of Line Drawings*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 6, 2006.

[10] Rutovitz, *Pattern Recognition*, J.Roy. Stat. Soc., vol 129. Series A, pp. 504-533, 1966.

[11] P. Mordohai et al. *Junction inference and classification for figure completion using tensor voting*. In Proc. of CVPRW, vol. 4, pp. 56-64, 2004.

[12] Naeem A. Bhatti et al., *Detection and classification of local primitives in line drawings*. In 35th Annual workshop of the Austrian Association for Pattern Recognition, 2011.

[13] Andrzej Sluzek. *A local algorithm for real-time junction detection in contour images*. In CAIP 01, pp. 465-472, 2001.

[14] X. You et al., *Wavelet-based Approach for Skeleton Extraction*. In Proc. of the 7^{th} IEEE Workshop on Appl. of CV.

[15] X. Gao et al., *Multiscale corner detection of contour images using wavelet transform*. Journal of Electronic Imaging.