2014 22nd International Conference on Pattern Recognition

Scalable Arrow Detection in Biomedical Images

K.C. Santosh†, Laurent Wendling‡, Sameer K. Antani†, George R. Thoma†
†Communications Engineering Branch, US National Library of Medicine (NLM)
National Institutes of Health (NIH)
8600 Rockville Pike, Bethesda, MD 20894, USA

‡SIP–LIPADE, Université Paris Descartes (Paris V)
45, rue des Saints-Pres, 75270 Paris Cedex 06, France
Email: {Santosh.KC, Sameer.Antani, George.Thoma}@nih.gov, Laurent.Wendling@parisdescartes.fr

***Abstract*—In this paper, we present a scalable arrow detection technique for biomedical images to support information retrieval systems under the purview of content-based image retrieval (CBIR) and text information retrieval (TIR). The idea primarily follows the criteria based on the geometric properties of the arrow, where we introduce signatures from key points associated with it. To handle this, images are first binarized via a fuzzy binarization tool and several regions of interest are labeled accordingly. Each region is used to generate signatures and then compared with the theoretical ones to check their similarity. Our validation over biomedical images shows the advantage of the technique over the most prominent state-of-the-art methods.**

***Keywords*—Arrow detection, biomedical images, content-based image retrieval and text information retrieval.**

I. Introduction

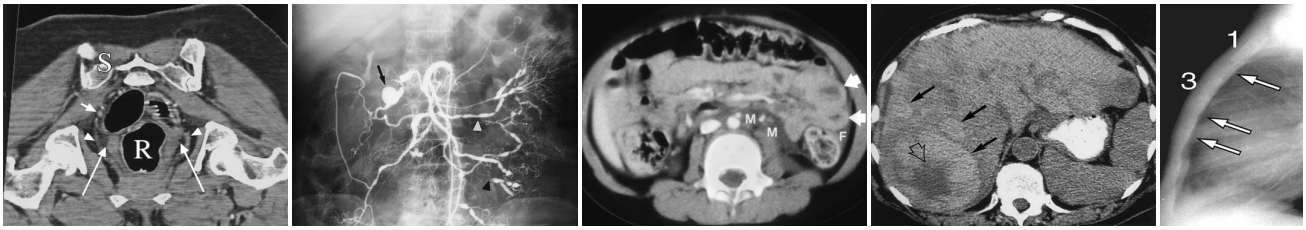
Biomedical images are valuable not only for educational and medical research purposes, but also for establishing clinical decision support system (CDSS) that can benefit from content-based image retrieval (CBIR). In turn, CBIR technologies can be advanced by automatically annotating the regions of interest (ROIs) on images. For clinicians and medical researchers, essential information is often conveyed through graphical illustrations in biomedical publications found in online databases such as the U.S. National Library of Medicine's PubMed Central® system. In order to exploit meaningful content (i.e., ROIs) from figures in these scientific publications, one needs either to index the whole image or provide access directly to its relevant parts. Medical images tend to be complex by nature and often contain several regions that need to be differentiated. Therefore, processing an entire image at one time is not trivial, and can make analysis systems cumbersome and inefficient. Medical researchers often use pointers (such as arrows and symbols) to highlight meaningful regions. Also the pointers minimize distractions from other less relevant regions in images. Additionally, these pointers are often referred to in figure captions and mentioned in the text of the associated document. Detecting arrows could help identify meaningful ROIs and improve CBIR performance. However, in the CBIR literature, there are few techniques to detect overlaid arrows. Existing methods rely on sparse pixel vectorization, segmenting text-like and symbol-like objects, and thresholding (either global or local). The techniques tend to vary for different types of images. In this paper, we present a scalable arrow detection technique that overcomes challenges posed by the visual complexity of the arrows in medical images (see Fig. 1), and advances the technology compared to previous text information retrieval frameworks [1], [2], [3].

In [4], Dori and Wenyin propose a technique to detect arrows based on sparse pixel vectorisation [5]. The concept relies on the cross sectional runs (or width runs) of the black area. This represents the line at intervals along the tracking direction and records the middle points of these sections. These points are used to construct vectors. Such a vectorization process results in many thick short bars from the arrow heads that are then used to make a decision. The technique utilizes an interesting application but is limited to machine printed line images. Measuring features such as eccentricity, convex area and solidity can be considered to detect arrows, but the current techniques are limited to regular arrows (i.e., straight arrows showing left, right, top and bottom) [6]. Additionally, the method uses pre-defined threshold to avoid small objects and noise. Cheng et al use text-like and arrow-like objects separation, assuming that arrows are shown in either black or white color with respect to the background [8]. From the binary image, arrow-like object separation employs a fixed sized mask (after removing the small objects and noise as in [6]), which are then used for feature computation such as major and minor axis lengths, axis ratio, area, solidity and Euler number. A recent study uses a pointer region and boundary detection to handle distorted arrows [10], which is followed by edge detection techniques and fixed thresholds as reported in [7], [9]. These candidates are used to compute overlapping regions, which are then binarized to extract the boundary of the expected pointers.

Fundamentally, edge-based arrow detection techniques are limited by the weak-edge problem [8], [6], [10], no matter how robust the binarization tools are. Hard thresholding (either global or local) is one of the primary reasons. This means that a hard threshold cue often weakens the decision in pointer detection. For edge detection in binary or grayscale images, most state-of-the-art methods use classical algorithms like Roberts, Sobel and Canny edge detection. Further, template-based methods are limited since they require new templates to train new images. Moreover, one needs to re-evaluate the threshold values when new images are used.

Edge-based techniques are worth considering since sampling points can be remarkably compact compared to solid regions, especially when a broken boundary does not affect the performance. In biomedical images, one of the major issues for a broken boundary is non-homogeneous intensity distribution, where pointers overlap with content. Broken lines can be

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.561



(a) sample 1 (b) sample 2 (d) sample 3 (c) sample 4 (e) sample 5

Fig. 1. A few examples of arrows in biomedical images. Considering background color, to make them visible, both filled and non-filled (black and white) arrows are employed, in addition to large variations in style.

recovered when gaps are small, but this may be inaccurate [11]. In this context, techniques that use key points hold promise and form the basis of our proposed method. Following the basic concept presented in [12], in this paper, we consider the geometrical definition of an arrow. To efficiently handle arrow detection, discrete signatures are computed along the contour from every candidate region and are compared with the theoretical signatures. Such a comparison can handle several different types of arrows, not just the regular ones.

The rest of the paper is organized as follows. We define arrow in Section II. This includes an explanation about discrete and theoretical arrows. Based on it, in Section III, we describe a complete system. It details the binarization process where lists of key points from every labeled connected component (CC) are used to perform discrete arrow assessment. We evaluate the the proposed technique using the imageCLEF dataset in Section IV, and then compare it with recent state-of-the-art methods. Section V concludes the paper.

II. Arrow definition

A. Arrow description

As shown in Fig. 2, an arrow is modeled as an isosceles triangle, $\mathbf{T}$, that is linked to a rectangle, $\mathbf{R}$,

$$\mathbf{T} = \left(A_{(x_a,y_a)}, B_{(x_b,y_b)}, C_{(x_c,y_c)}\right) \text{ and}$$
$$\mathbf{R} = \left(E_{(x_e,y_e)} F_{(x_h,y_h)} G_{(x_g,y_g)} H_{(x_f,y_f)}\right), \quad (1)$$

where we set $c = d(A,B)$ and $a = d(A,C) = d(B,C)$, and d refers to a Euclidean distance metric. The aim is to model both discrete and theoretical signatures of arrows as in [12] and to integrate them into a broad (scalable) arrow recognition model.

B. Discrete arrow signature

Let $Sec(a,c)$ be the sector defined from the segments $[A,C]$ and $[A,B]$ which includes T from A. We can then set the angle between (A,B) and (O,x) as,

$$\theta_B^A = \text{atan}\left(\frac{y_B - y_A}{x_B - x_A}\right) + m\pi, \quad (2)$$

where $x_B - x_A \neq 0$ and $m \in \mathbb{N}$, and similarly the angle θ_C^A between (A,C) and (O,x). A triangle is by definition a convex polygon. For any segment joining two points in $\mathbf{T}$, every point on the segment must also lie within $\mathbf{T}$. From A, take the pencil of lines $\mathcal{L}$ and the corresponding set V defined by $\mathbf{T}$ from A, in the sector $Sec(a,c)$,

$$\mathcal{L}^A = \left\{D_\theta^A\right\}_{\theta\in[0,\pi]} \text{ and } V_A = \{I_A^\theta\}_{\theta\in[\theta_B^A,\theta_C^A]}, \quad (3)$$

where I_A^θ is the segment beared by pencil of lines contained in $Sec(a,c)$. Note C_A on the circle centered in A of radius $r = \max(a,c)$ and all the segments I_A^θ, are included in C_A. Repeat the same process to define V_B. In addition, V_C includes the definition of $\mathbf{R}$ (that is using C_C centered in C of radius $r = \max(a, d(C,F))$). In this case, $\mathbf{T}$ and $\mathbf{R}$ are completely described (where $(CEFGH)$ is convex). For efficiency reasons, Bresenham's algorithm [14] is used since it is fast to minimize errors in drawing lines on integer grid points.

C. Theoretical arrow signature

In this section, we define the theoretical signature, S_X, associated with the discrete ones, V_X. To generalize, we consider any triangle, $\mathbf{T}'$, which consists of three unaligned points X_1, X_2 and X_3. X_1 is assumed to be the origin of the orthogonal frame and θ' and θ'' the angles described by the segments $[X_1,X_2]$ and $[X_1,X_3]$ in this frame. We also set $x = d(X_1,X_2)$ and $y = d(X_1,X_3)$. Let f be the function defining the new representation of $[X_2,X_3]$ from X_1 is given by: $S_{X_1}(\theta) : [\theta',\theta''] \to \mathbb{R}_*^+$

$$f(\theta, \langle x,y,\theta',\theta''\rangle) = \frac{x\cdot y\cdot \sin(\theta'+\theta'')}{x\cdot\sin(\theta-\theta') - y\cdot\sin(\theta-\theta'')}. \quad (4)$$

Based on Eq. (4), for the points A, B and C associated with the triangle $\mathbf{T}$, we have

$$\begin{aligned}
&S_A(\theta) : [\theta_B^A,\theta_C^A] \to \mathbb{R}_*^+,\ S_A(\theta) = f\left(\theta, \langle a,c,\theta_B^A,\theta_C^A\rangle\right),\\
&S_B(\theta) : [\theta_C^B,\theta_A^B] \to \mathbb{R}_*^+,\ S_B(\theta) = f\left(\theta, \langle b,c,\theta_C^B,\theta_A^B\rangle\right),\\
&S_C(\theta) : [\theta_A^C,\theta_B^C] \to \mathbb{R}_*^+,\ S_C(\theta) = f\left(\theta, \langle a,a,\theta_A^C,\theta_B^C\rangle\right), \quad (5)
\end{aligned}$$

where $\theta_A^C \leq \theta \leq \theta_B^C$, for $\mathbf{R} = \emptyset$ in case of $S_C(\theta)$. This does not hold true when $\mathbf{R} \neq \emptyset$. In the latter case, five different triangles are processed to express $S_C(\theta)$ (see Fig. 2),

$$S_C(\theta) \to \begin{cases}
\theta_A^C \leq \theta < \theta_E^C & f(\theta, \langle a, a', \theta_A^C, \theta_E^C\rangle)\\
\theta_E^C < \theta \leq \theta_F^C & f(\theta, \langle a', a'', \theta_E^C, \theta_F^C\rangle)\\
\theta_F^C < \theta \leq \theta_G^C & f(\theta, \langle a'', a'', \theta_F^C, \theta_G^C\rangle)\\
\theta_G^C < \theta \leq \theta_H^C & f(\theta, \langle a'', a', \theta_G^C, \theta_H^C\rangle)\\
\theta_H^C < \theta \leq \theta_B^C & f(\theta, \langle a', a, \theta_H^C, \theta_B^C\rangle)
\end{cases}$$

where $a' = d(C,E) = d(C,H)$ and $a'' = d(C,F) = d(C,G)$. On the whole, two different cases defined in the context of $\mathbf{R}$ must be used to compute $S_C(\theta)$. A more detailed idea about the signature's unicity can be found in [12], which was first introduced to recognize similar arrows often linked with area networks in line drawings.

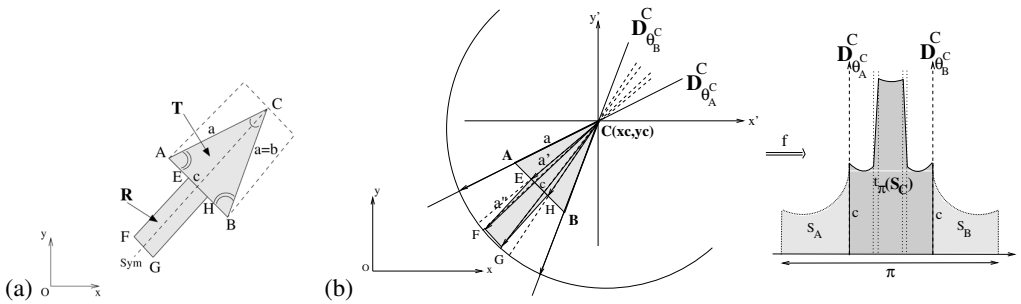


Fig. 2. An example illustrating (a) a discrete arrow and (b) a signature from point C i.e., $S_C(\theta)$ of an arrow using pencils of lines in that particular sector.

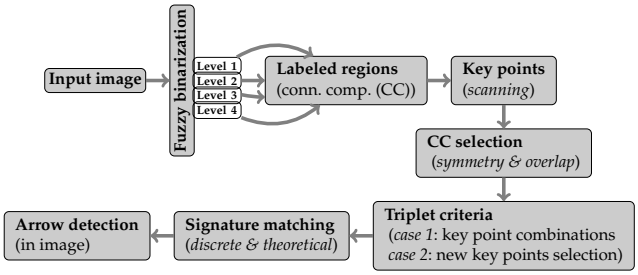


Fig. 3. Overall system workflow in block format.

III. Arrow detection

As shown in Fig. 3, our testbed system can be summarized as follows. It relies on a binarization process on grayscale images. To extract ROIs, we employ an adaptive binarization tool to produce four different levels of binarized images. We take CC from every level of binarization and select extreme points representing arrow heads. Since we consider the geometrical convex properties of an arrow, we detect arrowhead locations by considering a maximum of four scans. At the same point, two major criteria, viz., symmetry and overlap, are used to select the potential candidates (i.e., CCs). This step avoids noisy CCs. After that, we handle mainly two different cases: 1) three head points are checked whether it characterizes the inscribed circle; 2) two head points are studied in order to find a missing point by estimating via geometrical arrow description. Broadly speaking, both cases allow us to check a triplet of points that on the whole, characterize an arrow head. To make a decision, the sets of triplets are then estimated from the comparison of theoretical and discrete models. In our assessment, the corresponding CCs are said to be detected as arrows if their similarity ratio crosses the empirically set threshold.

A. Fuzzy binarization

In our dataset, arrows appear at either high or low intensity to enhance their visibility in the image. In addition, in many cases arrows are blurred, overlapped or surrounded by textured areas. In such contexts, typical binarization tools that are based on fixed threshold values do not extract ROIs perfectly. Therefore, we focus on an adaptive binarization tool, which is based on a fuzzy partition of a 2D histogram of the image, taking into account the gray level intensities and local variations [13]. 2D Z-function criteria based on the optimization of fuzzy entropy are then computed from this histogram to automatically set the threshold. Z-function employs two kernels: low level and high level cuts, in addition to direct inversions. The latter issue (image inversion) takes opposite image intensities into account. Altogether, four different binarized levels are processed, as illustrated in Fig. 4. Finally, CCs are extracted and labeled to identify the shapes as arrow-like objects.

B. Key points

The aim here is to find arrowhead points (*cf.* Fig. 2) by studying four scans over the orthogonal frame. Consider a scan s and a set of K CCs $\{\mathbf{G}_k\}_{k=1,\dots,K}$, where $k = \mathbf{I}_{ij}$ is the value of the point at (i,j), associated with CCs $\mathbf{G}_k$ in an image $\mathbf{I}$ to be evaluated. While evaluating, a point is added to a list $L_s^{\mathbf{G}_k}$

$$L_s^{\mathbf{G}_k} = L_s^{\mathbf{G}_k} \cup \{(i,j)\} \text{ iff } \mathbf{I}_{ij} = k \wedge \forall t \neq s, (i,j) \notin L_s^{\mathbf{G}_k}. \quad (6)$$

Fig. 5 shows a list of key points associated with all CCs. Based on the convex property of arrow heads, these four scans are sufficient to catch point C (as shown in Fig. 2(a)). Other arrowhead points can be issued from the scan or they can be estimated (see Section III-C and Fig. 6) from the location of point C.

C. Discrete arrow assessment

For any connected component k, as described previously, the discrete signature relies on the location of the arrow head key points A, B and C. During the recognition process, we compute geometric properties of an arrow considering any combination of points.

Point C first, can be estimated from the symmetric axis of the arrow $Sym(C)$. By definition, four scans of the image are enough to include point C. If several points are found, any point of the arrow can be reached by a segment starting from C, that eventually provides the size of the arrow. Consider the set of n points X_i included in all $\mathcal{L}_s^{C_k}$ and p the number of bins of V_i,

$$Sym(X_i) = \sup_{t \in [1,\dots,\frac{p}{2}]} \left\{ \frac{\sum_{i=1}^{p} \min\left(V_{X_i}^{t+i}, V_{X_i}^{t-i}\right)}{\sum_{i=1}^{p} \max\left(V_{X_i}^{t+i}, V_{X_i}^{t-i}\right)} \right\}, \text{ then}$$
$$Sym(C) = \arg\max_{i=1,\dots,n} \{Sym(X_i)\}. \quad (7)$$

The aim is now to assess the location of points A and B from C. Sym is maximum when angle t corresponds to the axis of symmetry of the signature. The symmetry axis $\Delta(C)^t$

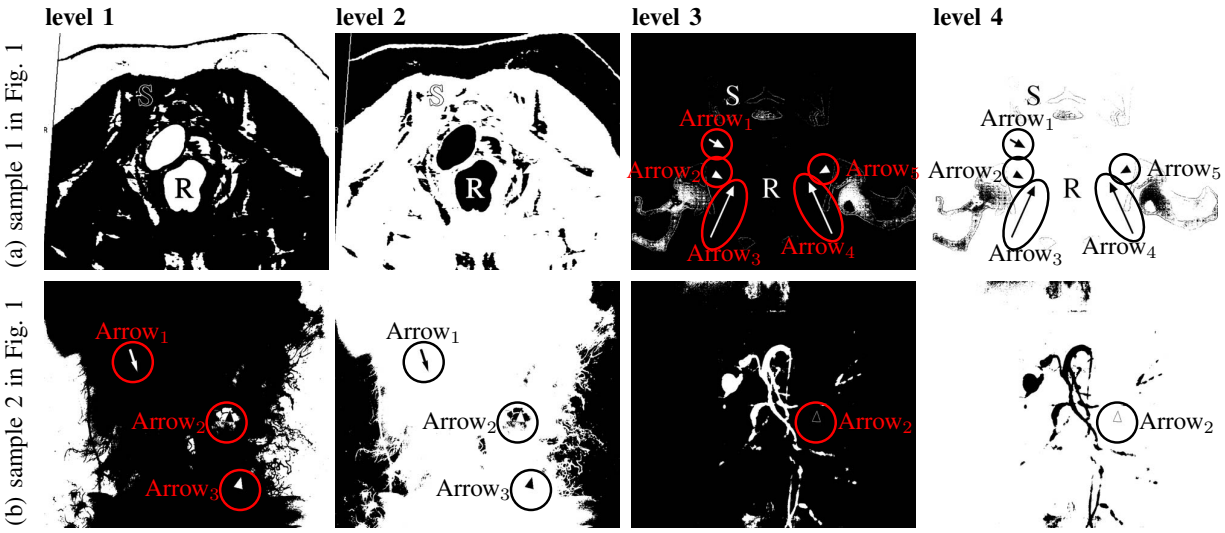


Fig. 4. Fuzzy binarization at four different levels using the samples from Fig. 1. The overall idea is to illustrate that no arrows are missed during binarization. Arrows are encircled in both red and black (with respect to the background color). Note that in sample 2, arrow 2 is fully surrounded as well as overlapped by noisy textures at the first two levels, while it is visible at the other two levels.

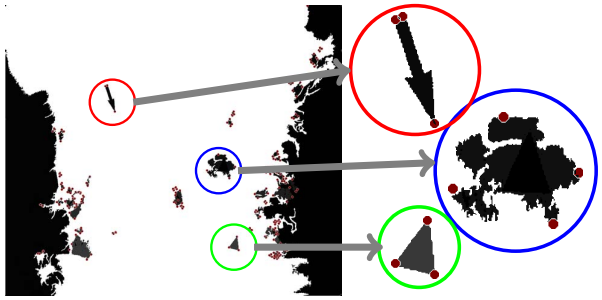


Fig. 5. An example (sample 2, level 2 in Fig. 4) illustrating the CCs and their corresponding key points, resulting from four orthogonal scanning. In this example, a few regions of interest are magnified to make key points visible.

following the angle t is computed to split the remaining points into two lists. Since the base of the triangle largely covers the rectangle, we can consider the farthest points A_* and B_* belonging to list such that $\Delta(C)^t \perp (A_*, B_*)$ with

$$Cardinality = \frac{\min(|V_A|, |V_B|)}{\max(|V_A|, |V_B|)}, \{V_A^i, V_B^i, t_\pi(V_C^i)\} \simeq 1, \quad (8)$$

where $|V_X|$ refers to the area of the signature. Additionally, we also consider another criterion i.e., an area for assessment,

$$Area = 1 - \left(\frac{\mathbf{K} - \mathbf{H}}{\mathbf{H}}\right), \ \mathbf{H} = \sqrt{(s(s-a)(s-b)(s-c))}, \quad (9)$$

where $\mathbf{K}$ is the common area identified from the scans performed from A, B and C, and $s = \frac{1}{2}(a+b+c)$ is the semi-perimeter to check the well-defined isosceles triangle. By definition, if the triplet describes a triangle in the image, the value of $\mathbf{K}$ must be close to the calculation of the Herons formula $\mathbf{H}$.

Considering the whole shape of the CC, both symmetric and overlapping criteria based on $\mathbf{H}$ are first employed to limit the number of possible candidates. Two main cases can then

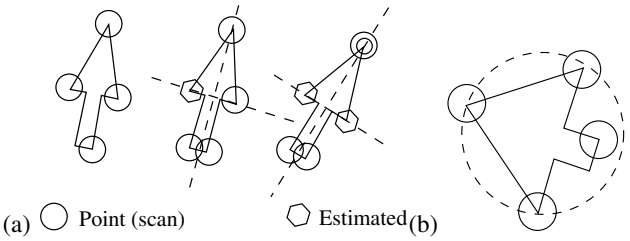


Fig. 6. Examples of arrows: (a) regular and (b) degenerated. Arrows that do not possess rectangle at the base, can be one of the degenerated examples.

be processed for regular and degenerated arrows (see Fig. 6). In case of regular arrows, arrow head points can be directly captured from scanning, which is the case when an arrow belongs to a surrounding circle or follows the location of arrows. If not, the same set of points is used again to estimate the remaining points with respect to C.

D. Decision

In the state-of-the-art, authors use several metrics to compare two distributions. In our test, a basic similarity ratio (SR) based on 'Tanimoto' index (the minimum over maximum) is calculated between the theoretical and discrete signatures. In Fig. 7, we demonstrate an example of a discrete signature from a sample arrow to see how similar it is to the theoretical one. Magnitude of the signatures can be decreased to limit the impact of the size of the rectangle by focusing essentially on head arrow. But, an overlapping is appeared when we compute a signature from an arrow with noise and/or from other pointer-like objects. Therefore, SR is weighted by an overlapping assessment in order to compute recognition rate,

$$Reco = \text{SR}\left(1 - \frac{\sum_{i=1}^{p} \inf\{V_A^i, V_B^i, t_\pi(V_C^i)\} - 2 \cdot d(A,B)}{\sum_{i=1}^{p} \sup\{V_A^i, V_B^i, t_\pi(V_C^i)\}}\right). \quad (10)$$

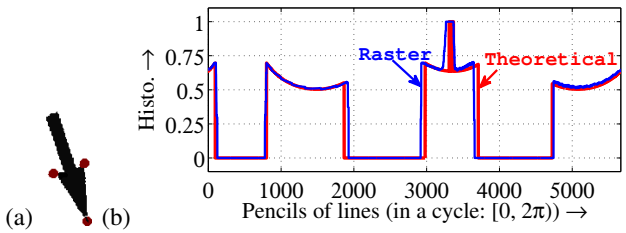


Fig. 7. Discrete and theoretical signatures, for a global comparison. It includes both signatures from triplets and rectangles.

It is not surprising that the more the overlapping, the lower the recognition rate. To avoid noisy or distorted areas, we employ a threshold λ (empirically designed) to support the decision process whether the shape (i.e., any labeled CC from binarization in Section III-A) is recognized as an arrow.

IV. Evaluation

A. Datasets, ground-truth and evaluation protocol

A very well-known imageCLEF dataset [15] is used for testing, which is composed of 298 chest CT images. Each image is expected to have at least a single arrow. There are 1049 pointers, on the whole. For all images in the dataset, ground-truths of the pointers were created and each ground-truth includes information like arrow type, color, location, and direction. For validation, for any given image in the dataset, our performance evaluation criteria are precision, recall and F_1-score.

$$Metrics = \begin{cases} m1/M & \text{precision} \\ m1/N & \text{recall} \\ 2\left(\frac{(m1/M)\times(m1/N)}{(m1/M)+(m1/N)}\right) & F_1\text{-score}, \end{cases} \quad (11)$$

where $m1$ is the number of correct matches from the detected set M and N is the total number of pointers (in the ground-truth) that are expected to be detected.

B. Results and comparison study

Before providing an overall performance of the proposed method, we first provide a visual demonstration. It aims to provide an intuitive feeling about how our method works. Following Figs. 4 and 5, we provide outputs in Fig. 8 based on the decision (defined by $\lambda = 0.85$). Outputs follow the samples presented in Fig. 1. As an example, considering signature matching, the arrow detection scores from 'sample 1' are as follows (see Fig. 4 for arrow labeling):

	$Arrow_1$	$Arrow_2$	$Arrow_3$	$Arrow_4$	$Arrow_5$
score (%)	0.94	0.87	0.94	0.96	0.89

Moreover, in this illustration, one can take a closer look into sample 2 and 4, where two different binarized levels have been used to detect arrows. Therefore, their composition will provide a complete solution, no matter which binarized level is used. Note that we avoid redundancies in arrow detection if any CC is found to be repeated in two or more different binarized levels. More output samples are provided in Fig. 9, where test images are taken from radiology image dataset (outside the imageCLEF dataset). These output samples attest to the extensibility of the proposed method.

TABLE I. Arrow detection performance (in %).

Methods	Precision	Recall	F_1-score
Method 1 [7]	22.80	77.80	35.00
Method 2 [8]	81.10	74.10	77.00
Method 3 [10]	84.20	81.60	83.00
Our method	93.14	86.92	89.94

In our test dataset, we have white- and black-filled arrows and therefore category-wise performance evaluation is organized as follows.

	White	Black	Overall
recognition rates (%)	84.51	88.43	86.92

Using the whole dataset and the evaluation protocol defined in Eq. (11), experimental results are provided in Table I, where false positives are also taken into account. Further, the comparison study with recent state-of-the-art methods 1) global thresholding-based method [8] and 2) two edge-based methods [7], [10] has been made. Based on the reported results, we have observed the following. Higher precision suggests that the binarization tool does not miss significant number of arrows and similarly the robustness of the arrow detection algorithm that can handle up to 90% in average. But on the other side, binarization tool still carries some artifacts resembling the arrow-like objects. Theoretically speaking, our algorithm cannot reject those triangle shaped CCs. The latter issue however, does not significantly affect the system. Besides, big arrows provide higher recognition rates since their signatures are robust to noise, compared to small ones. Smaller ones do not offer clear arrow heads (see test 2 in Fig. 9). In addition, our system does not offer correct detection of triangle-shaped arrows with concave base, in few cases. In this test, we do not consider other pointers such as ✽ and ♣. Even though, the overall performance provides difference with the benchmarking methods.

V. Conclusion and Future Work

In this paper, we have presented a scalable arrow detection technique for biomedical images. The idea is based on the geometric properties of the arrow, where we introduce signatures from key points associated with it. After image binarization, the similarity between the signatures from every labeled region and theoretical ones leads to a decision. As shown in this paper, our validation using biomedical images (from imageCLEF dataset) outperforms the state-of-the-art methods for straight arrows.

Based on the dataset used, a few cases of occulted arrows will be improved by considering the impact of junction points as arrowhead candidates. Besides, since the current work does not consider curved arrows, and fixed graphical shaped pointers like ✽ and ♣, we plan to integrate shape description (DTW-Radon [16], for instance) to detect similar CCs based on their shape similarity. The proposed method is found to be applicable to a broader set of images, which is not just limited to biomedical images.

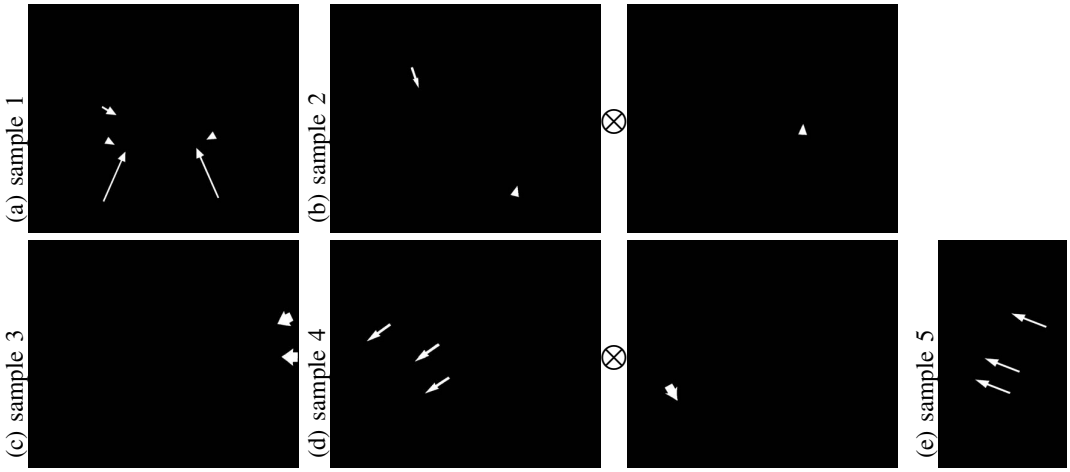


Fig. 8. Output examples, which are correctly matched with ground-truths. The outputs are associated with the samples provided in Fig. 1. Among all, sample 2 and 4 require two different binarization levels to detect arrows and thus two outputs are superimposed.

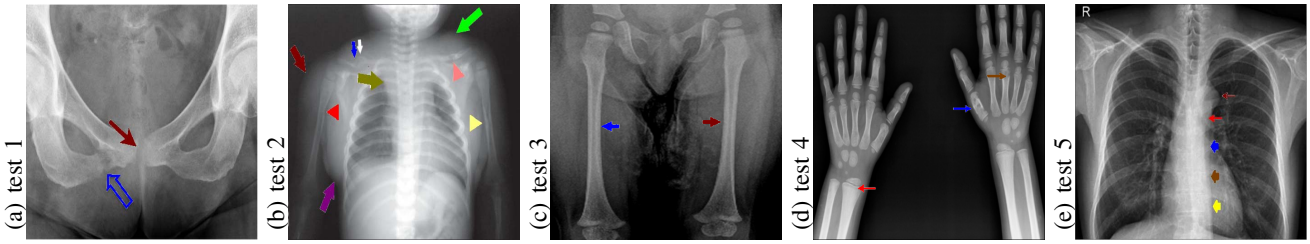


Fig. 9. Arrow detection output samples using radiology images. Both filled and non-filled arrows are detected. Arrows without color are missed (see test 2).

ACKNOWLEDGEMENTS

This research was supported by the Intramural Research Program of the National Institutes of Health (NIH), National Library of Medicine (NLM), and Lister Hill National Center for Biomedical Communications (LHNCBC). The authors would like to thank the NIH Fellows Editorial Board for their editorial assistance.

REFERENCES

[1] T. M. Deserno, S. Antani, and L. R. Long, "Ontology of gaps in content-based image retrieval," *J. Digit. Imag.*, vol. 22, no. 2, pp. 202–215, 2009.

[2] D. Demner-Fushman, S. Antani, M. Simpson, and M. Rahman, "Combining text and visual features for biomedical information retrieval and ontologies," LHNCBC Board of Scientific Counselors, NIH, Bethesda, MD, Tech. Rep., September 2010.

[3] M. Simpson, D. You, S. Antani, G. Thoma, and D. Demner-Fushman, "Towards the creation of a visual ontology of biomedical imaging entities," in *AMIA Annu Symp Proc.*, 2012.

[4] D. Dori and L. Wenyin, "Automated cad conversion with the machine drawing understanding system: Concepts, algorithms, and performance," *IEEE SMC-part A: Syst. and Humans*, vol. 29, pp. 411–416, 1999.

[5] D. Dori, S. Member, and W. Liu, "Sparse pixel vectorization: An algorithm and its performance evaluation," *IEEE PAMI*, vol. 21, pp. 202–215, 1999.

[6] J. Park, W. Rasheed, and J. Beak, "Robot navigation using camera by identifying arrow signs," in *Int. Conf. on Grid and Pervasive Computing*. 2008, pp. 382–386.

[7] D. You, E. Apostolova, S. Antani, D. Demner-Fushman, and G. R. Thoma, "Figure content analysis for improved biomedical article retrieval," in *DRR*, vol. 7247. SPIE, 2009, pp. 1–10.

[8] B. Cheng, R. J. Stanley, S. De, S. Antani, and G. R. Thoma, "Automatic detection of arrow annotation overlays in biomedical images," *Int. J. Healthc. Inf. Syst. Inform.*, vol. 6, no. 4, pp. 23–41, Oct. 2011.

[9] D. You, S. Antani, D. Demner-Fushman, M. M. Rahman, V. Govindaraju, and G. R. Thoma, "Biomedical article retrieval using multimodal features and image annotations in region-based CBIR," in *DRR*, vol.7534. SPIE, 2010, pp. 1–10.

[10] D. You, M. S. Simpson, S. Antani, D. Demner-Fushman, and G. R. Thoma, "A robust pointer segmentation in biomedical images toward building a visual ontology for biomedical article retrieval," in *DRR*, vol. 8658. SPIE, 2013.

[11] O. Hori and D. S. Doermann, "Robust table-form structure analysis based on box-driven reasoning," in *ICDAR*. 1995, pp. 218–.

[12] L. Wendling and S. Tabbone, "A new way to detect arrows in line drawings," *IEEE PAMI*, vol. 26, no. 7, pp. 935–941, 2004.

[13] H. Cheng and Y.-H. Chen, "Fuzzy partition of twodimensional histogram and its application to thresholding," *Pattern Recogn.*, vol. 32, pp. 825–843, 1999.

[14] B. J., "Algorithm for computer control of a digital plotter," *IBM Syst. Journal*, vol. 4, no. 1, pp. 25–30, 1965.

[15] H. Müller, J. Kalpathy-Cramer, I. Eggel, S. Bedrick, J. Reisetter, C. E. K. Jr., and W. R. Hersh, "Overview of the clef 2010 medical image retrieval track," in *Working Notes of CLEF*, 2010.

[16] K. C. Santosh, B. Lamiroy, and L. Wendling, "Dtw-radon-based shape descriptor for pattern recognition," *IJPRAI*, vol. 27, no. 3, 2013.