Arrowhead detection in biomedical images

***K.C. Santosh*[√]*; Naved Alam*[†]*, Partha Pratim Roy*[†]*, Laurent Wendling*[‡]*, Sameer Antani*[♯]*, George R. Thoma*[♯]

[√]**Dept. of Computer Science, The University of South Dakota, Vermillion, SD 57069, USA**
[†]**Dept. of Computer Sci., Indian Institute of Technology, Roorkee, Uttarakhand 247667, INDIA**
[‡]**LIPADE, Université Paris Descartes (Paris V), 75270 Paris Cedex 06, FRANCE**
[√,♯]**National Library of Medicine, National Institutes of Health, Bethesda, MD 20894, USA**

[√] **Corresponding author – santosh.kc@usd.edu**

Abstract

Medical images in biomedical documents tend to be complex by nature and often contain several regions that are annotated using arrows. Arrowhead detection is a critical precursor to region-of-interest (ROI) labeling and image content analysis. To detect arrowheads, images are first binarized using fuzzy binarization technique to segment a set of candidates based on connected component principle. To select arrow candidates, we use convexity defect-based filtering, which is followed by template matching via dynamic programming. The similarity score via dynamic time warping (DTW) confirms the presence of arrows in the image. Our test on biomedical images from imageCLEF 2010 collection shows the interest of the technique.

Introduction

Motivation

Essential information is often conveyed succinctly through graphical illustrations and figures/images in biomedical publications. Medical images tend to be complex by nature, and are often annotated with graphical overlay pointers, such as arrows and asterisk. Medical researchers often use these pointers to highlight meaningful regions-of-interest (ROIs) (see Fig. 1), while minimizing distractions from other less relevant regions. Additionally, they are often referred to in figure captions and mentioned in the article text. Therefore, detecting arrows could help identify meaningful ROIs and annotate them with the concepts appearing in the biomedical text [1, 2]. This paper improves on prior work in arrow detection toward meeting this goal in image content analysis.

Related work

We find that there are few techniques reported in the literature to detect overlaid arrows. Existing methods rely on sparse pixel vectorization, segmenting text-like and symbol-like objects, and global or local thresholding.

In [3], Dori et al, proposed a technique to detect arrows based on sparse pixel vectorization [4]. The concept relies on the cross sectional runs (or width runs) of black image regions (assuming arrow in black). These runs represent the line at intervals along the tracking direction and records the middle points of these sections. The points are then used to construct vectors. The 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. Features such as eccentricity, convex area and solidity has been used to detect arrows, but the current techniques are limited to regular arrows (i.e., straight arrows showing left, right, top and bottom) [5]. 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 [6]. From the binary image, arrow-like object separation employs a fixed sized mask (after removing the small objects and noise as in [5]), 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 [7], which is followed by edge detection techniques and fixed thresholds as reported in [8, 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 [5–7]. No matter how robust the arrow detection techniques are, hard thresholding (either global or local) is one of the primary reasons for failure. 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. Template-based methods are limited since they require new templates to train new images. Also, it may be necessary to re-evaluate the threshold values when new images are used. Edge-based techniques are still considered since sampling points can be remarkably compact compared to solid regions, especially when broken boundaries are recoverable. In biomedical images, one of the major issues for a broken boundary is non-homogeneous intensity distribution, where pointers overlap with content. According to Hori and Doermann (1995), broken lines can be recovered when gaps are small but, in practice, they are often inaccurate [10].

Contribution outline

Our method can be summarized as shown in Fig. 2. It relies on a grayscale fuzzy binarization process at different levels. Similar to previously reported work [11], candidates are segmented based on connected component (CC) principle. These candidates are filtered using hull convexity defect-based technique. This step helps prune artefacts (or unwanted noisy connected components) and store arrowhead-like candidates. Next we perform template

Figure 1. *Three examples showing different types of arrows pointing specific image regions. These are taken from published biomedical articles.*

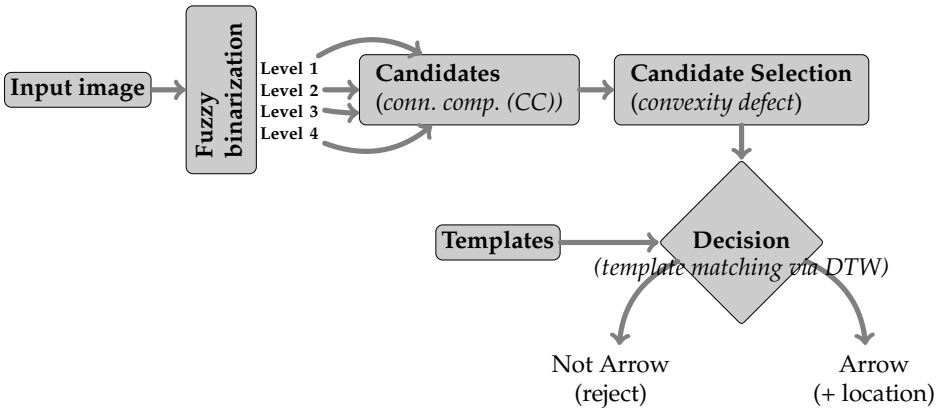

Figure 2. *Overall system workflow in block format. Block-wise explanation can be found in section 'method'.*

matching using dynamic programming to confirm whether the candidate is an arrowhead. In our assessment, An arrow is said to be detected if their matching score exceeds an empirically set threshold.

Unlike the common state-of-the-art methods, our method uses four different levels of fuzzy binarization. This ensures that overlaid arrow candidates are not missed. however, it may result in repetitions. We note that the primary variation in an arrow appearance is due to its tail (shape and size). Therefore, our method limits itself to just detecting arrowheads.

The remainder of the paper is organized as follows. In section 'method', we explain our concept in detail, where it mainly includes binarization process and candidate selection. Results are reported in experiments section, including a comprehensive state-of-the-art comparison. In the conclusion section, we state conclusions and provide next-steps.

Method

Fuzzy binarization

In biomedical images (see Fig. 1), arrows appear with 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 are unable to perfectly extract candidate regions. 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 [12]. 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. 3. In Fig. 3 (a), arrow candidates are encircled in both red and black (with respect to the background color). The main idea of using four different levels of binarization is not to miss the overlaid arrows. Furthermore, deformed and/or distorted arrows can be discarded since the arrows are repeated in other levels of binarization. In Fig. 3 (b), some of the arrows are repeatedly segmented.

From a pool of several candidates (see Fig. 3 (b)), we are required to select arrow-like candidates. In what follows, we describe a complete candidate selection process in detail.

Candidate selection

Our candidate selection process is based on the characteristics of the arrowhead, which can typically be represented by a triangle. Unlike the previously reported work [11], we do not take tail information into account. One of the primary reasons is that it may vary geometric signatures computed from extreme points of a triangle (i.e., triplet) because tail structures tend to vary from time to time. Such a change will affect overall appearance of the arrow (Fig. 4). After we detect arrowhead, we will take the corresponding tail into account since both came from the same CC.

To detect an arrowhead, the following steps are carried out: 1) convexity defect-based arrowhead candidate cropping; and

Figure 3. *Fuzzy binarization (of Fig. 1 (c)): (a) four different levels (level 1 to level 4), where the segmented arrows are encircled both in red and black with respect to the background color; and (b) a collection of all segmented CCs including arrows (encircled in red).*

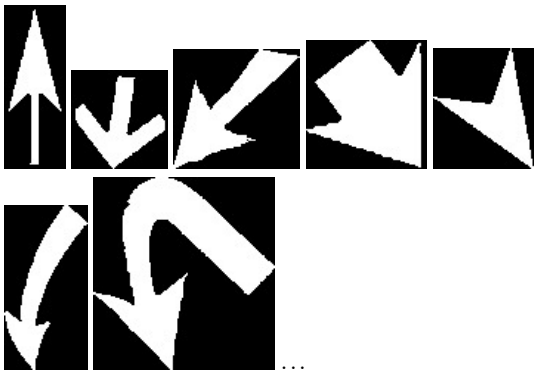

Figure 4. *Examples showing the changes in tail structure (plus its absence).*

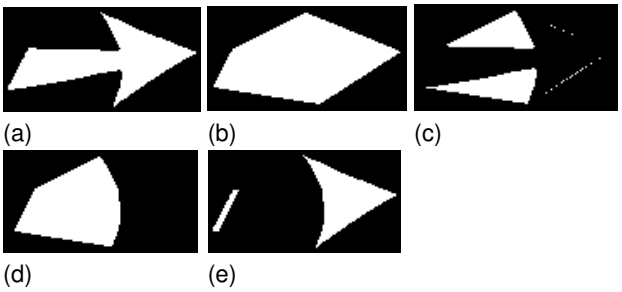

Figure 5. *Arrowhead candidate cropping: (a) an arrow, (b) convex hull, (c) convexity defect, (d) a complete convexity defect region, and (e) arrowhead candidates.*

2) arrowhead candidate matching with the templates.

Convexity defect-based arrowhead candidate cropping

To select arrow-like candidates, we apply hull convexity defect concept (see Fig. 5). A set of points along the contour of the binary CC are defined to be convex if it contains the line segments connecting each pair of its points. In a convex combination, each point x_i in the set S is assigned a weight or coefficient w_i in such a way that the coefficients are all non-negative and sum to one, and these weights are used to compute a weighted average of the points. For each choice of coefficients, the resulting convex combination is a point in the convex hull, and the whole convex hull can be formed by choosing coefficients in all possible ways. Expressing this as a single formula, the convex hull is the set: $\left\{\sum_{i=1}^{|S|} w_i x_i | (\forall i : w_i \geq 0) \wedge \sum_{i=1}^{|S|} w_i = 1\right\}$. This means that the convex hull of a finite point set $S \in \mathbb{R}^n$ forms a convex polygon when n = 2. In Fig. 5 (b), an example is shown. Using such a convex hull, we attempt to remove tail since their exists convex shaped silhouettes in both sides (see Fig. 5 (c)), which is computed by subtracting an original candidate from the convex hull. In Fig. 5 (d), the convexity defect region is shown, which is just a convex hull of both convex shaped silhouettes. At the end, in Fig. 5 (e), arrowhead candidate(s) is(are) selected by subtracting an original image with the convexity defect region.

Arrowhead candidate matching with template

To confirm arrowhead candidates (see Fig. 5), we apply a template matching technique. We extract a feature along the contour and match with the predefined templates using dynamic time warping (DTW) technique. The arrowhead candidate is confirmed when the similarity score crosses the empirically designed threshold.

Feature extraction. Along the contour, we have a set of coordinate points, $P = \{p_i\}_{i=1,\ldots,n}$. To extract feature vector (f), we compute the change in angle with respect to x-axis from any consecutive pair, $f = \{\alpha_i\}_{i=1,\ldots,n}$, where $\alpha_i = \arctan\left(\frac{y_l - y_{l-1}}{x_l - x_{l-1}}\right)$. This goes in a cyclic order either clock wise or anti-clock wise. In our feature vector, continuous redundancy of α_i can be possible, $\alpha_i = \alpha_{i+j}, j = 1,\ldots,m$, where $m \leq n$. Therefore, it is desired to express the contours of shapes with a few representative pixels (called the dominant points). Through polygonal approximation [13–15], we represent a digital curve using fewer points such that the properties of the curvature of the digital curve are retained. Next the geometrical properties like inflexion points or concavities can be evaluated. In our implementation, to make it simple and effective, we compute the difference between the angles and check whether it crosses the threshold, ε. The choice of ε is usually user-defined. This means we take α_i if $|\alpha_i - \alpha i + 1| \leq \varepsilon$. Like most line fitting/polygonal approximation (or dominant point detection) methods, it can be made non-parametric by using the error bound due to digitization as a termination condition. Fig. 6 shows three examples, where the changes in angles are shown at all dominant points. To make the feature vector rotation invariant, one needs to follow either clockwise or anti clock-wise to compute changes in angles.

Dynamic time warping (DTW). DTW allows to find the dissimilarity between two non-linear sequences potentially having different lengths [16, 17]. In Fig. 6, one can notice the variations in feature vector from one arrowhead to another. Let us consider two feature sequences: $f_1 = \{\alpha_i\}_{1=1,\ldots,n}$ and $f_2 = \{\beta_j\}_{j=1,\ldots,m}$ of size n and m, respectively. The aim of the algorithm is to provide the optimal alignment between both sequences. At first, a matrix of size $n \times m$ is constructed. Then for each element, local distance metric $\delta(i,j)$ between the events e_i and e_j is computed i.e., $\delta(i,j) = (e_i - e_j)^2$. Let $D(i,j)$ be the global distance up to (i,j),

$$D(k,l) = \min \begin{bmatrix} D(k-1,l-1), \\ D(k-1,l), \\ D(k,l-1) \end{bmatrix} + \delta(k,l)$$

with an initial condition $D(1,1) = \delta(1,1)$ such that it allows warping path going diagonally from starting node $(1,1)$ to end (n,m). The main aim is to find the path for which the least cost is associated. The warping path therefore provides the difference cost between the compared signatures. Formally, the warping path is, $W = \{w_k\}_{k=1\ldots l}$, where $\max(i,j) \leq l < i + j - 1$ and k^{th} element of W is $w(i,j)_k \in [1:n] \times [1:m]$ for $k \in [1:l]$. The optimised warping path W satisfies the following three conditions: boundary condition, monotonocity condition and continuity condition. We then define the global distance between f_1 and f_2 as,

$$\Delta(f_1, f_2) = \frac{D(n,m)}{l}.$$

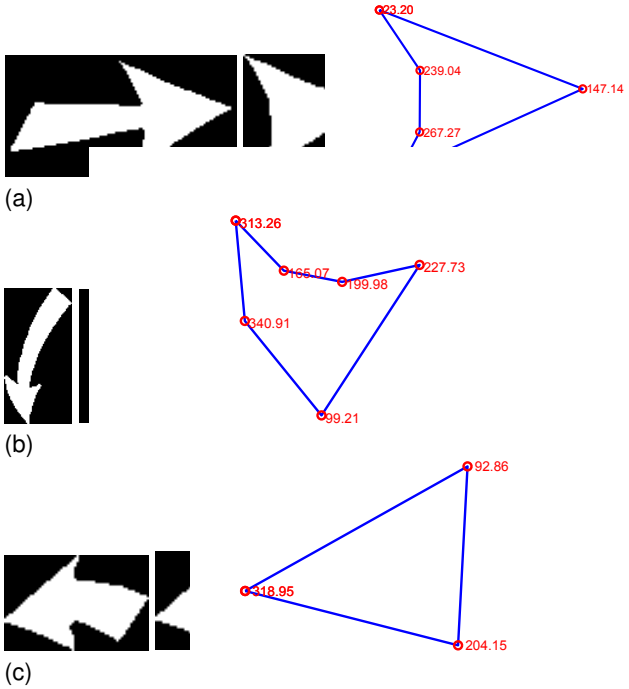

Figure 6. *Three examples showing a complete process (from left to right) starting from an original candidate (resulting from fuzzy binarization - see Fig. 3), arrowhead cropping (see Fig. 5) to feature extraction after polygonal approximation.*

The last element of the $n \times m$ matrix gives the DTW-distance between f_1 and f_2, which is normalised by l i.e., the number of discrete warping steps along the diagonal DTW-matrix. Overall, DTW measures the similarity between two sequences, and can be summarized as follows.

1) If the cropped candidate is not actually an arrowhead, DTW results in high cost. The results are opposite to those arrowhead candidates.

2) Thanks to DTW, noise in arrowhead (along the contour) does not let the cost to go beyond the threshold. This means that some of the arrowheads with noisy artefacts connected to them are still detected.

3) Feature extraction and DTW matching techniques provide robustness to rotation and scale changes. As an example, $\Delta($  $) = 0.00$ and $\Delta($  $) = 0.00$

Experiments

Datasets, ground-truth and evaluation protocol

The imageCLEF dataset[23] is used for testing. It is composed of 298 chest CT images. Each image is expected to have at least one arrow, and there are 1049 pointers, in total. For all images in the dataset, ground-truth 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,

Table 1: Performance comparison (in %).

Metric	Our method	Previously reported methods				Template-based methods				
		M1 [6]	M2 [8]	M3 [7]	M4 [11]	GFD [18]	SC [19]	ZM [20]	RT [21]	D-Radon [22]
Precision	88.50	81.10	22.80	84.20	93.14	75.10	68.30	55.20	59.50	62.10
Recall	93.80	74.10	77.80	81.60	86.92	78.33	71.40	57.70	63.60	65.30
F_1-score	91.09	77.00	35.00	83.00	89.94	76.68	69.82	56.40	61.48	63.65

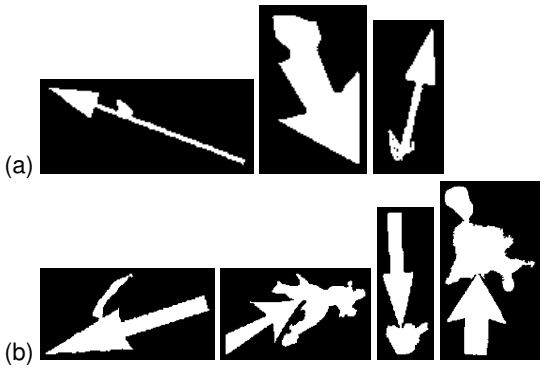

Figure 7. *Examples of when the proposed method succeeds and fails: (a) noisy artefacts connected along the tail does not affect the method, and (b) they do largely affect when connected with arrowhead.*

our performance evaluation criteria are precision, recall and F1-score,

$$\text{precision} = \frac{m_1}{M}, \quad \text{recall} = \frac{m_1}{N} \quad \text{and}$$

$$F_1 \text{ score} = 2\left(\frac{(m_1/M)\times(m_1/N)}{(m_1/M)+(m_1/N)}\right),$$

where m_1 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.

Result and analysis

Our results

Table 1 shows the performance evaluation scores in terms of precision, recall and F1-score. In the reported results, we prioritize the recall measure since we do not like to miss arrow candidates. The method achieved F_1 score of 91.09%.

Our method is able to detect arrowheads regardless their tail structure. But, if the shape of the arrowhead is affected by noisy artefacts, the proposed method fails. Fig. 7 shows both examples: noisy artefacts that are connected along the tail, and noisy artefacts that are connected with arrowhead. Also, the method does not detect highly curved arrows since convexity defect-based arrowhead cropping does not yield expected arrowhead candidates (i.e., almost all curved arrows are missed).

Comparative study

Further, the comparative study with state-of-the-art methods has been made. In this comparison, our benchmarking methods are categorized into two groups: 1) state-of-the-art methods that are specially designed for arrow detection; and 2) common template-based method by using well-known state-of-the-art shape descriptors.

Recent arrow detection methods. Four well-known methods from the state-of-the-art that are specially designed for arrow detection are used: 1) global thresholding-based method (M1) [6], 2) two edge-based methods (M2:M3) [7, 8], and 3) a template-free geometric signature-based method [11]. The results are provided in Table 1, where method 4 (M4) performs the best with precision, recall and F_1 score values 93.14%, 86.93% and 89.94%, respectively.

Template-based methods. In case of template-based method, we created 11 templates (arrows) having different shapes (including sizes). The template size can further be extended in accordance with the dataset. To extract shape features, we took the most frequently used shape descriptors (in computer vision) from the state-of-the-art. They are 1) generic Fourier descriptor (GFD) [18], 2) shape context (SC) [19], 3) Zernike moment (ZM) [20], 4) R-transform (RT) [21] and 5) DTW-Radon [22]. As before, results (precision, recall and F_1-score) are provided in Table 1. Among all shape descriptors, GFD provides the best performance.

On the whole, considering such a dataset, the proposed method outperforms the best state-of-the-start arrow detection method by more than 1% F_1 score, and the template-based (shape descriptor) method by more than 16% F_1 score, at the cost of low precision and high recall.

Conclusion & future work

In this paper, we have presented a new method to detect overlaid arrows in biomedical images. Images are first binarized via fuzzy binarization tool to segment a set of candidates. To select arrow candidates, we use a hull convexity defect-based arrowhead cropping, which is followed by template matching via dynamic programming. In our assessment, (using imageCLEF 2010 collection), our results outperforms the state-of-the-art methods.

To the best of our knowledge, this is the first time arrow detection has been done without using tail information since variations in the shape and size of the tail change an overall shape of the complete arrow. As our next steps, we plan to integrate previously reported techniques (state-of-the-art methods) that can be used as pre- or post-processing steps. Further, use of machine learning instead of using template-based approaches, would be our immediate concern.

Acknowledgements

This research was supported [in part] 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).

References

[1] D. Demner-Fushman, Sammer Antani, Matthew Simpson, and MM Rahman. Combining text and visual features for biomedical information retrieval and ontologies. Technical report, LHNCBC Board of Scientific Counselors, National Institutes of Health, Bethesda, MD, September 2010.

[2] Dina Demner-Fushman, Sameer Antani, Matthew S. Simpson, and George R. Thoma. Design and development of a multimodal biomedical information retrieval system. *Journal of Computing Science and Engineering*, 6(2):168–177, 2012.

[3] Dov Dori and Liu Wenyin. Automated cad conversion with the machine drawing understanding system: Concepts, algorithms, and performance. *IEEE Transactions on System, Man, and Cybernetics-part A: System and Humans*, 29:411–416, 1999.

[4] Dov Dori, Senior Member, and Wenyin Liu. Sparse pixel vectorization: An algorithm and its performance evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:202–215, 1999.

[5] Jongan Park, Waqas Rasheed, and Junguk Beak. Robot navigation using camera by identifying arrow signs. In *International Conference on Grid and Pervasive Computing - Workshops*, pages 382–386. IEEE Computer Society, 2008.

[6] Beibei Cheng, R. Joe Stanley, Soumya De, Sameer Antani, and George R. Thoma. Automatic detection of arrow annotation overlays in biomedical images. *Int. J. Healthc. Inf. Syst. Inform.*, 6(4):23–41, October 2011.

[7] Daekeun You, Matthew S. Simpson, Sameer Antani, Dina Demner-Fushman, and George R. Thoma. A robust pointer segmentation in biomedical images toward building a visual ontology for biomedical article retrieval. In Richard Zanibbi and Bertrand Coüasnon, editors, *Document Recognition and Retrieval*, volume 8658 of *SPIE Proceedings*. SPIE, 2013.

[8] Daekeun You, Emilia Apostolova, Sameer Antani, Dina Demner-Fushman, and George R. Thoma. Figure content analysis for improved biomedical article retrieval. In Kathrin Berkner and Laurence Likforman-Sulem, editors, *Document Recognition and Retrieval*, volume 7247 of *SPIE Proceedings*, pages 1–10. SPIE, 2009.

[9] Daekeun You, Sameer Antani, Dina Demner-Fushman, Md. Mahmudur Rahman, Venu Govindaraju, and George R. Thoma. Biomedical article retrieval using multimodal features and image annotations in region-based cbir. In Laurence Likforman-Sulem and Gady Agam, editors, *Document Recognition and Retrieval*, volume 7534 of *SPIE Proceedings*, pages 1–10. SPIE, 2010.

[10] O. Hori and D. S. Doermann. Robust table-form structure analysis based on box-driven reasoning. In *International Conference on Document Analysis and Recognition - Volume 1*, pages 218–, Washington, DC, USA, 1995. IEEE Computer Society.

[11] K. C. Santosh, Laurent Wendling, Sameer Antani, and George Thoma. Scalable arrow detection in biomedical images. In *Proceedings of IAPR International Conference on Pattern Recognition*, pages 3257–3262, Stockholm (Sweden), August 2014. IEEE Computer Society.

[12] H.D. Cheng and Y.-H Chen. Fuzzy partition of two dimensional histogram and its application to thresholding. *Pattern Recognition*, 32:825–843, 1999.

[13] Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3):244 – 256, 1972.

[14] David H. Douglas and Thomas K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122, 1973.

[15] Dilip K. Prasad, Maylor K.H. Leung, Chai Quek, and Siu-Yeung Cho. A novel framework for making dominant point detection methods non-parametric. *Image and Vision Computing*, 30(11):843 – 859, 2012.

[16] Hiroaki Sakoe. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26:43–49, 1978.

[17] Eamonn J. Keogh and Michael J. Pazzani. Scaling up dynamic time warping to massive dataset. In *European PKDD*, pages 1–11, 1999.

[18] D. Zhang and G. Lu. Shape-based image retrieval using generic fourier descriptor. *Signal Processing: Image Communication*, 17:825–848, 2002.

[19] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.

[20] Whoi-Yul Kim and Yong-Sung Kim. A region-based shape descriptor using zernike moments. *Signal Processing: Image Communication*, 16(1-2):95 – 102, 2000.

[21] Thai V. Hoang and Salvatore Tabbone. The generalization of the r-transform for invariant pattern representation. *Pattern Recognition*, 45(6):2145–2163, 2012.

[22] K. C. Santosh, Bart Lamiroy, and Laurent Wendling. Dtw-radon-based shape descriptor for pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 27(3), 2013.

[23] Henning Müller, Jayashree Kalpathy-Cramer, Ivan Eggel, Steven Bedrick, Saïd Radhouani, Brian Bakke, Charles E Kahn Jr, and William Hersh. Overview of the clef 2009 medical image retrieval track. In *Multilingual Information Access Evaluation II. Multimedia Experiments*, pages 72–84. Springer, 2010.

Author Biography

K.C. Santosh, Ph.D., is an Assistant Professor for the department of computer science at the University of South Dakota (USD). Before joining the USD, he worked as a research fellow at the U.S. National Library of Medicine (NLM), National Institutes of Health (NIH). He has demonstrated expertise in pattern recognition, image processing, computer vision and machine learning with various applications such as graphics recognition, document information exploitation, medical image analysis and biometrics.

Naved Alam is currently a master student in computer science at Indian Institute of Technology (IIT), Roorkee. His research areas include image analysis, pattern recognition and machine learning.

Dr. Partha Pratim Roy received his Ph.D. degree in computer science in 2010 from Universitat Autònoma de Barcelona, Barcelona. He worked as postdoctoral research fellow in the Computer Science Laboratory (LI, RFAI group), France (2010-2012) and in Synchromedia Lab, Canada (2013). Presently, Dr. Roy is working as an Assistant Professor at Indian Institute of Technology (IIT), Roorkee. His main research area is pattern recognition.

Laurent Wendling received the Ph.D. degree in Computer Science in 1997 from the Université Paul Sabatier, Toulouse, France. He received the habilitation degree in 2006. From 1999 to 2009, he was an assistant professor at the ESIALNancy, and a member of LORIA. He is currently a full professor at the Université Paris Descartes. He is also a member of the SIP team, LIPADE. His current research topics are spatial relations, feature selection, and image segmentation.

Dr. Sameer Antani received his B.Eng. in Computer Engineering from the University of Pune, India (1994), and M.Eng., and Ph.D. both in Computer Science and Engineering from the Pennsylvania State University, USA (1998, 2001). He is a scientist at the National Library of Medicine (NLM), part of the National Institutes of Health (NIH), USA. His R&D includes various topics in biomedical imaging, information retrieval, and global health. He is a senior member of SPIE.

George R. Thoma is Chief of the Communications Engineering Branch of the Lister Hill National Center for Biomedical Communications, a research and development division of the U.S. National Library of Medicine at the National Institutes of Health. He earned a B.S. from Swarthmore College, and the M.S. and Ph.D. from the University of Pennsylvania, all in Electrical Engineering. Dr. Thoma is a Fellow of the SPIE, the International Society for Optical Engineering.