Towards Real-time Guidewire Detection and Tracking in the Field of Neuroradiology

Martin Spiegel [a,d], Marcus Pfister [b], Dieter Hahn [a], Volker Daum [a], Joachim Hornegger [a,d], Tobias Struffert [c], Arnd Dörfler[c]

[a]Friedrich-Alexander University Erlangen-Nuremberg(FAU), Department of Computer Science, Chair of Pattern Recognition, Martensstr. 3, 91058 Erlangen, Germany
[b]Siemens AG, Sector Healthcare AX, Siemensstr. 1, 91301 Forchheim, Germany
[c]FAU, Department of Neuroradiology, Schwabachanlage 6, 91054 Erlangen, Germany
[d]Erlangen Graduate School in Advanced Optical Technologies (SAOT), Paul-Gordan-Str. 6, 91052 Erlangen, Germany

ABSTRACT

Two-dimensional roadmapping is considered state-of-the-art in guidewire navigation during endovascular interventions. This paper presents a methodology for extracting the guidewire from a sequence of 2-D roadmap images in almost real time. The detected guidewire can be used to improve its visibility on noisy fluoroscopic images or to do a back projection of the guidewire into a registered 3-D vessel tree. A lineness filter based on the Hessian matrix is used to detect only those line structures in the image that lie within the vessel tree. Loose wire fragments are properly linked by a novel connection method fulfilling clinical processing requirements. We show that Dijkstra's algorithm can be applied to efficiently compute the optimal connection path. The entire guidewire is finally approximated by a B-spline curve in a least-squares manner. The proposed method is both integrated into a commercial clinical prototype and evaluated on five different patient data sets containing up to 249 frames per image series.

Keywords: Localization and Tracking Technologies, Guidewire Tracking, B-splines, Dijkstra algorithm, Lineness filter

1. INTRODUCTION

Minimally invasive treatments of cerebrovascular pathologies, i.e. aneurysms, stenoses or fistulae, have become an alternative to invasive vascular surgery in certain cases. During these treatments, both a guidewire and a microcatheter are inserted into the patient's vessel system for diagnosis and treatment. The navigation and positioning of the guidewire all the way up to the cerebrovascular pathology is accomplished by x-ray guidance and 2-D roadmapping techniques. To the best of our knowledge, no commercial product is available so far, that covers guidewire detection given a fluoroscopic image series in a sufficient manner regarding efficiency(processing speed), usability(minimal user interaction) and robustness. A reason for this might be the poor image quality of the fluoroscopic images which exhibit a signal-to-noise ratio of approximately one. However, a detected guidewire together with its B-spline approximation would deliver further diagnostic information to the physician due to its improved visibility in noisy images. The possibility to do a back projection into a registered 3-D vessel tree would lead to a complete 3-D navigation tool.

Palti-Wassermann et. al.[1] introduced a guidewire extraction method working on a single frame without time considerations. Lin[2] considered the guidewire extraction in x-ray images as a minimum-cost path using Salience Distance Transform, introduced by Rosin and West.[3] There is an abundance of literature concerning the enhancement of line-like structure such as center lining and vessel extraction.[4–8] Barbu et al.[9] proposed a method for learning a curve model for detection and segmentation of guidewires. Baert et. al.[10] introduced a semiautomatic guidewire tracking method using template matching and a spline energy optimization approach based

---

Further author information: (Send correspondence to M. Spiegel)
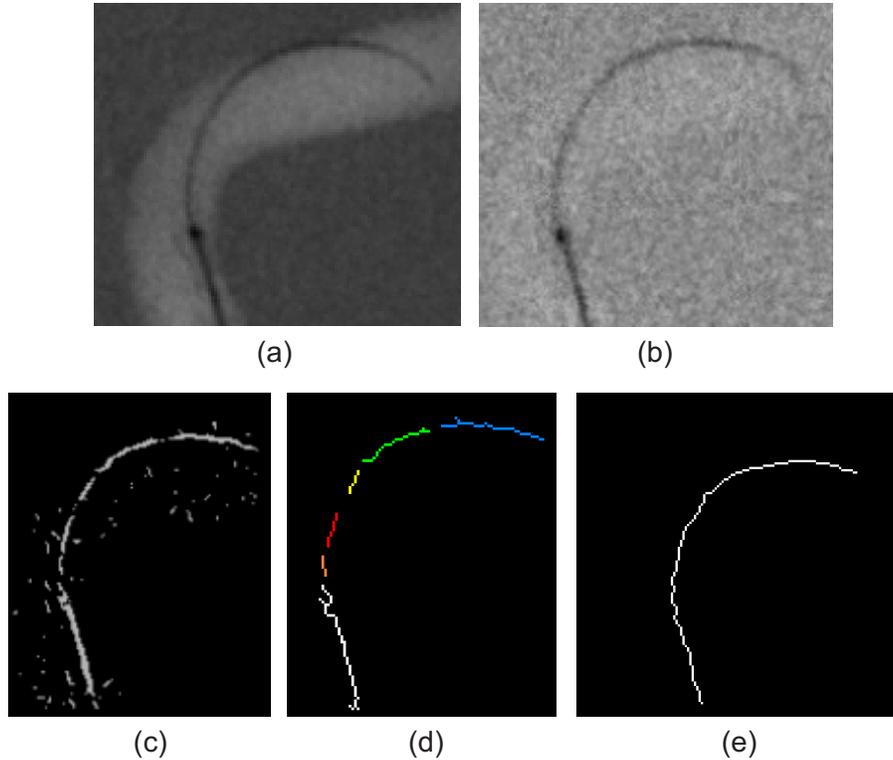M. Spiegel: E-mail: martin.spiegel@informatik.uni-erlangen.de, Telephone: +49 (0) 9131 85 27826

Figure 1. (a) Original 2-D roadmap image depicting the guidewire, cropped for clarity. (b) Subtraction image. (c) Response of the lineness filter, scaled for contrast. (d) Labeled image after skeletonization and region growing. (e) B-spline approximates the guidewire

on a feature image where line-like structures are enhanced. However, this approach does not fulfill the efficiency requirements for clinical applications e.g. coiling of aneurysms or stenting of stenoses.

Our goal was to develop a guidewire detection system that statisfies the requirements of the clinical workflow i.e. efficiency, accuracy and usability. During endovascular interventions, neuroradiologists usually require a processing speed of 7-8 frames per second (FPS). Furthermore, user interaction should be avoided as much as possible in order to achieve a high acceptance by doctors. This paper formulates a fast automated guidewire detection method that is already integrated into a clinical prototype. We incorporate a 2-D lineness filter for guidewire detection and the vessel information into a framework that is related to 'Live-Wire'.[11] For efficiency, the detected line segments are properly linked together with Dijkstra's algorithm.[12] Finally, the entire guidewire is approximated with B-splines. We experimentally evaluated the method on image series from five different patients with a total of 769 frames showing the clinical applicability of our approach.

2. METHODS

Given a sequence of 2-D roadmap images, the first few images are used for both the mask image generation and the vessel tree extraction. The mask image in this context is composed of temporally filtered 2-D roadmap images, where no guidewire is present. Subsequent 2-D roadmap images are subtracted from this mask image so that the resulting subtraction images ideally feature only background noise and a black line representing the guidewire (see Figure 1 (b)). The subtraction images are then normalized to have a zero mean and a standard deviation of one. Since the guidewires always remain within the vessel system during an endovascular intervention, one can further reduce the search space for guidewire detection by extracting the vessel tree and only examining the vessel part of the image. Thus, our next task is to extract the vessel tree. Since an ordinary 2-D roadmap image as in Figure 1 (a) is composed of two classes of pixel intensities, Otsu's threshold segmentation algorithm[13] is applied to separate bright pixels being vessel structure and dark pixels denoting background noise. The vessel

tree extraction takes less than 0.5 seconds and has to be repeated for every working X-ray view. However, a change of the X-ray view involves the reset of the entire 2-D roadmap process, i.e. (1) removal of the guidewire from the patients vessel system, (2) new contrast agent injection, (3) mask image generation and (4) vessel tree extraction which is negligible compared to the time needed for this reset. Since the guidewire tip position will not be known for the first few iterations, the detection system looks only at the image boundary (15% of the image dim.) so as to fulfill the real time constraints.

2.1 2-D Lineness Filter

The guidewire appears as a curve within subtraction images (see Figure 1 (b)). Those elongated structures are extracted by a line filter[4, 14] which makes use of the eigenvalues of the Hessian matrix at scale σ. The Hessian matrix $\mathbf{H}_\sigma$ is defined as

$$\mathbf{H}_\sigma = \left(\begin{array}{cc} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{array} \right) \tag{1}$$

where f_{xy} denotes the convolution of the subtraction image with the second derivative of the 2-D isotropic Gaussian $G(x, y, \sigma)$

$$f_{xy} = f * \frac{\partial^2}{\partial x \partial y} G(x, y, \sigma) \tag{2}$$

where σ is the standard deviation and x and y are the pixel coordinates. The eigenvalue decomposition of $\mathbf{H}_\sigma$ delivers two sorted eigenvalues, i.e. $|\lambda_1| \leq |\lambda_2|$. Only those pixels are considered as curve candidates whose magnitude of the first eigenvalue λ_1 is very small since its eigenvector points in the direction of the guidewire exhibiting low intensity variations. The magnitude of the second eigenvalue λ_2 has to be rather high because the second eigenvector points across the guidewire resulting in high intensity variation. This leads to the following constraints for curve segments that are detected by the lineness filter:

$$\begin{aligned} |\lambda_1| &\approx 0 \\ |\lambda_1| &\ll |\lambda_2| \\ |\lambda_2| &> \epsilon \end{aligned} \tag{3}$$

where ϵ denotes a user-specified threshold. One needs to keep in mind that guidewires appear in different sizes in the image, depending on their thickness and the applied zoom factor of the angiography system. Hence, a multi-scale filtering approach as introduced by Frangi et. al.[4] or Klein et. al.[5] is used to overcome these issues. For the remainder of the paper we will refer to as a 'lineness image'.

2.2 Noise Reduction

Depending on the image quality, the lineness filter ideally detects the entire guidewire in one or more curve fragments. However, it also enhances other linear structures which correspond to noise, as depicted in Figure 1 (c). For this reason the 'lineness image' is skeletonized and labeled to obtain further information about the detected regions, e.g. the number of pixels per region and endpoints of curve fragments which have only a single neighbor within an 8-pixel neighborhood. Each detected region has to exhibit a minimum number of pixels otherwise it is considered an artifact and is eliminated. After this noise reduction, the remaining curve segments together with respective number of pixels and endpoints are considered as potential guidewire fragments (see Figure 1 (d)).

2.3 B-spline Generation and Connection Algorithm

At the end, the final guidewire curve should be approximated with a single B-spline requiring a proper connection of the curve fragments to one final curve. Therefore, this section describes a novel application of Dijkstra's algorithm for solving two issues: (a) B-spline generation and selection and (b) incremental B-spline connection to approximate the entire guidewire with one B-spline. Moreover, Dijkstra's algorithm with a Fibonacci heap structure is applied to statisfy the efficiency constraints of a clinical guidewire detection and tracking method. We used third order B-splines where the number of control points are one third of the number of pixels along the curve fragment.

B-spline Generation and Selection

Before Dijkstra's algorithm can be applied, the noise reduced image (see Figure 1 (d)) with $M \times N$ pixels is mapped to a graph with $M \times N$ vertices that are associated with costs. The vertex cost function $c(k)$ assigns a weight to the vertex k that depends on the location of k in the image:

$$c(k) = \left\{ \begin{array}{rcl} 1 & : & \text{if } k \in L \\ 100 & : & \text{if } k \notin L \end{array} \right. \tag{4}$$

where L denotes the set of all curve segments. The edge weight associated with the directed edge between vertex k_{a} and an adjacent vertex k_{b} is considered to be the vertex cost of $ec(k_{\mathrm{a}}, k_{\mathrm{b}}) = c(k_{\mathrm{b}})$. So far, L exhibits curve segments denoting an unordered set of pixels with at least two endpoints. More than two endpoints per curve segment occur due to skeletonization effects. Now, we catch all curves from L with more than two endpoints and put them into a new set, called E. E is defined as follows: $E \subseteq L$ with $E = \{e | e \in L | n > 2\}$ where n is the number of endpoints of the curve segment e. Thus, given $x \in E$ all endpoints of x are connected to each other via Dijkstra's algorithm to eliminate all endpoints of x except for two that represent the longest part of the curve segment x. Hence, $\forall x \in E : x$ can be decomposed into $\binom{n}{2}$ curves. Thus, Dijkstra's algorithm is used to search for a connection path between the endpoints of a curve segment. To compute all combinatorial paths, the total number of Dijkstra calls in this step evaluates to $|E| * \binom{n}{2} + |L \backslash E|$. The resulting paths denote an ordered sequence of pixels per curve segment that can be approximated with B-splines in real time using a commonly known least-squares approach.[15] $\forall x \in E$ result in $\binom{n}{2}$ B-splines where the B-spline with the largest length is taken to represent the curve segment of x.

Incremental B-spline Connection

Now, let S be the set of B-splines whose elements have to be linked to one final B-spline representing the entire guidewire. Again, Dijkstra's algorithm is used to compute the connection paths between the remaining B-splines. To ensure that the connection path between two B-splines proceeds on line structures and within the vessel tree, the 'lineness image' and the vessel information is incorporated into the modified cost function $c_{\mathrm{mod}}(k)$ (see equation 4):

$$c_{\mathrm{mod}}(k) = \left\{ \begin{array}{rcl} 1 & : & \text{if } k \text{ is an element of curve-like structures} \\ 20 & : & \text{if } k \text{ lies inside the vessel tree} \\ 100 & : & \text{if } k \text{ lies outside the vessel tree} \end{array} \right. \tag{5}$$

Let $s \in S$ be the B-spline with the largest length, the incremental B-spline connection method uses s as the starting B-spline to be successively linked to the remaining B-splines in S. During the composition of the final B-spline, there may be several connection possibilities to other B-splines available. In this case, the length of the connection B-spline together with its curvature is taken into account to select that connection B-spline yielding the smoothest and shortest link.

2.4 Treatment of Motion Artifacts

During an intervention, the physician puts tension stress on the guidewire in order to navigate through the vessel system. This may result in fast guidewire movements. These movements generate motion artifacts on the roadmap image due to the temporal filtering of the angiographic system and will be detected by the lineness filter (see Figure 2). To eliminate these artifacts, all B-splines are traversed at equidistant positions to search along the normal vector for parallel B-splines. If two splines are parallel, the shorter one is removed.

2.5 Region of Interest - ROI

As long as no guidewire is present in the first roadmap images, the region of interest (ROI) is defined as the image boundary as mentioned at the beginning of section 2. When the guidewire first becomes visible on the image, the ROI changes to cover the entire length of the guidewire. Once the guidewire is firstly approximated with a B-spline, the ROI changes again to a rectangle centered at the guidewire tip.
From that point on, all image processing methods, i.e. 2.1, 2.2 and 2.3 are only computed within the ROI to ensure real-time processing. During the intervention, the guidewire tip moves through the image and as a result
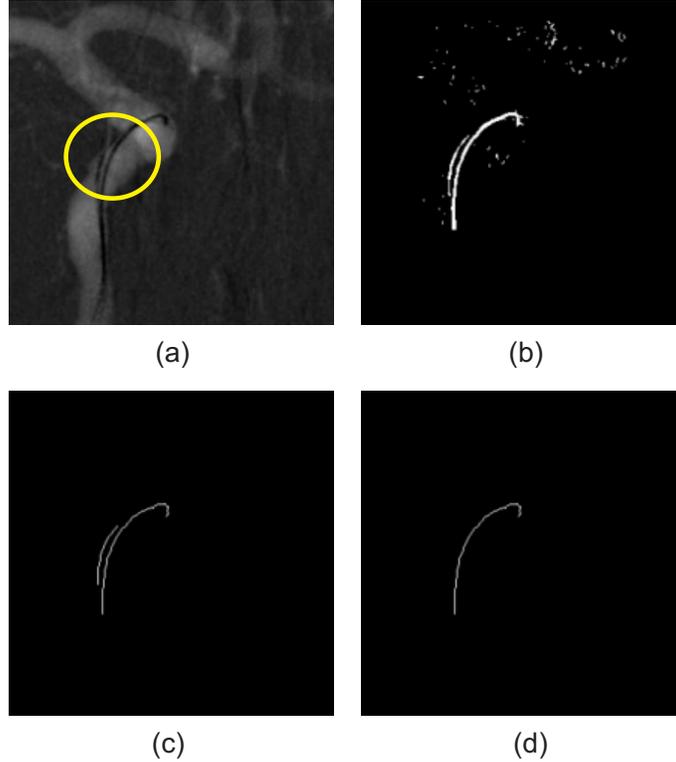
Figure 2. (a) A roadmap image depicting a motion artifact (yellow circle) due to quick movements of the guidewire. (b) Image result from the lineness filter. (c) After noise reduction and B-spline generation, two B-splines are parallel to each other. (d) The shorter B-spline is removed.

the ROI may not cover the entire guidewire anymore, as can be seen in Figure 3. The first time the ROI does not cover the entire guidewire length, two specific points of the guidewire, called Q1 and Q2, are identified for further processing in the next upcoming image in the image series. Q1 is the entry point of the guidewire in the image and Q2 is the entry point of the guidewire into the ROI. In Figure 3, Q1 and Q2 are depicted as orange circles. The position of Q1 and Q2 gets updated within each iteration and Dijkstra's algorithm links them to get a B-spline called $s_{\mathrm{init}} \in S$.

In this case a 'lineness image' is not required to link Q1 and Q2 because Dijkstra's algorithm runs only on the vessel image where dark pixels denoting the guidewire are associated with low costs and white pixels get high costs. As long as the ROI does not cover the entire guidewire, s_{init} will be taken as the starting B-spline for the incremental B-spline connection method (see section 2.3).

3. RESULTS

The evaluation of our proposed method is done on the bases of five patient image series with a sequence length between 32 and 249 frames acquired from a Siemens AXIOM Artis dBA TWIN x-ray fluoroscopy system during clinical interventions. The image sizes vary between $512{\times}512$ and $776{\times}776$ and the pixel spacing in x/y is $0.15/0.15$ in mm. A qualitative image result of the guidewire detection and tracking is illustrated in Figure 3. The method is quantitatively evaluated through the construction of a gold standard segmentation, where the guidewire is manually outlined in all 769 images. To quantify the performance of our method, four measurements are applied: efficiency, tracking, mean distance, and tip distance. Table 1 summerizes the results of the measurements.

The *efficiency* of a guidewire detection method describes a crucial measurement in terms of clinical applicability. Physicians usually require 7-8 frames per second during an endovascular intervention. Our proposed technique is detecting the guidewire well within 1-2 frames per second where the entire framework is not optimized.
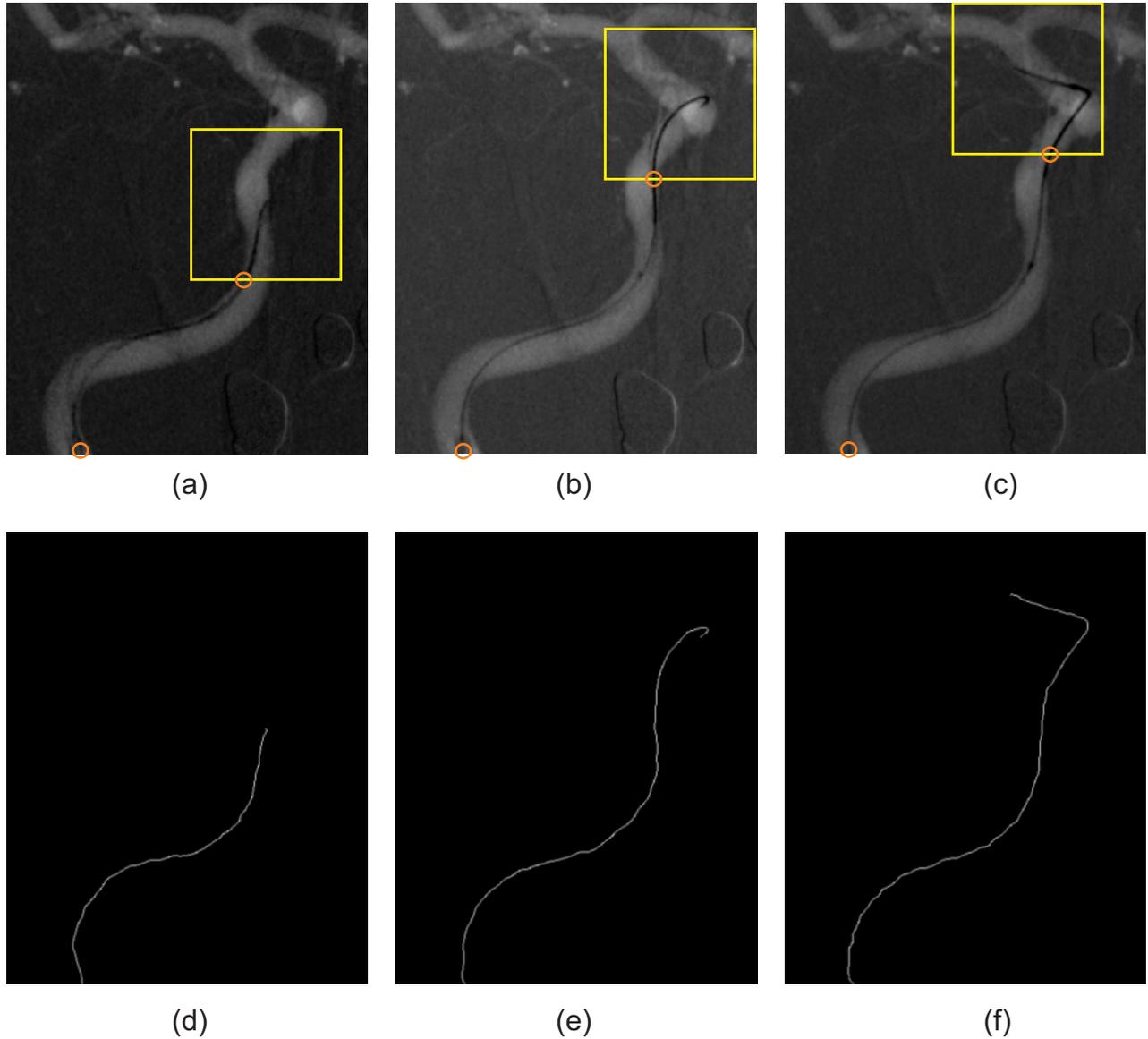
Figure 3. Tracking examples out of the image series patient 1 (see Table 1).

The *tracking* success depends on the number of failed detected guidewire that are easily identified by visual inspection of the detection results. In this context, a failed detection is considered either that the entire guidewire was well detected but the tip part was not due to ghosting effects or the B-spline connection algorithm (see section 2.3) linked the curve fragments in a wrong manner. We only considered successful tracking results for the computation of the mean and tip distance. The accuracy of a segmentation is usually computed by the percentage of false positives and false negatives of a segmentation compared to the gold standard. However, the exact guidewire position is often unclear and different observers may outline the guidewire differently.

Hence, the *mean distance* is calculated for evaluating the entire fit of the detected guidewire compared to the gold standard segmentation, as done in Baert et. al.[10] For each position $\mathbf{x}_i \in \mathbb{R}^2$ on the final B-spline curve C_1, its closest point to the gold standard segmentation C_2 is determined and vice versa. Thus, the mean distance

Measurements	patient 1	patient 2	patient 3	patient 4	patient 5
# Frames	132	249	223	133	32
Dimension x/y	720/720	512/512	720/720	720/720	776/776
Efficiency in FPS	1.1-1.5	1.9-2.1	1.2-1.4	1.2-1.4	1.2-1.4
Tracking	78%	98%	85%	95%	88%
avg. d_{tip} ($\sigma_{d_{tip}}$)	0.79(0.77)	1.05(0.90)	1.28(1.08)	0.89(0.93)	1.20(0.74)
avg. d_{mean} ($\sigma_{d_{mean}}$)	0.28(0.13)	0.25(0.20)	0.32(0.16)	1.02(0.33)	0.52(0.20)

Table 1. Quantitative detection results: minimum and maximum efficiency in frames per second (FPS), tracking rate, average mean distance and average tip distance. The σ denotes the standard deviation. The distances are given in mm.

yields a symmetric distance measure being defined as:

$$d_{mean}(C_1, C_2) = \frac{1}{2N}\left(\sum_{i=1}^{N} d_{min}(C_1(\mathbf{x}_i), C_2) + \sum_{i=1}^{N} d_{min}(C_1, C_2(\mathbf{x}_i))\right) \qquad (6)$$

$$d_{min}(C_1(\mathbf{x}_i), C_2) = \min d(C_1(\mathbf{x}_i), C_2(\mathbf{x}_j)) \;\; j \in [1, N] \qquad (7)$$

where d_{min} denotes the minimum distance of point $C_1(\mathbf{x}_i)$ on curve C_1 to curve C_2 and N indicates the number of sample points on the final B-spline and gold standard curve.

The *tip distance* (as in Baert et. al.[10]) measures the distance between the endpoint of the B-spline of the gold standard and the endpoint of the automatically determined B-spline.

4. DISCUSSION

Our methodology proposes a novel automated solution to the guidewire detection and tracking problem. The successful implementation and integration into a clinical prototype together with the new Dijkstra-based connection algorithm states the major contribution of this work. Nonetheless, when implementing this method, due to the employment of the lineness filter, the sigma range and the threshold of the second eigenvalue ϵ have to be carefully selected. Currently, the sigma range $1.2 \leq \sigma \leq 1.6$ and the value of $\epsilon = 1.2$ are obtained more or less empirically. This could potentially be automated to completely avoid user interaction.

The results obtained during the evaluation phase are comparable and better than those published for guidewire detection.[10, 16] Considering the experimental result in Baert et. al.[10] using subtration images, the mean distance lies between 1.15 and 1.44 and the mean tip distance lies between 4.92 and 5.04. Especially the efficiency results look promising for real-time guidewire detection and tracking applications.

The efficiency (processing speed) of our method is between one and two frames per second, which looks promising for a real clinical application compared to the far slower spline-energy based approach from Baert et. al.[10] Due to the temporal filtering of the angiography system, fast guidewire movements result in motion artifacts such that the actual guidewire position is unclear. Hence, the average tip distance exhibits higher values. However, the guidewire could be tracked up to 98 % within an image series. In Baert et. al.,[10] the tracking success ranges between 54% and 73% for subtraction images.

5. CONCLUSION

This paper introduces a novel and fast guidewire detection method for sequential 2-D roadmap images which is already integrated into a clinical prototype. The method incorporates multi-scale lineness filtering and vessel information into a framework which employs Dijksta's algorithm to efficiently connect guidewire fragments. So far the proposed guidewire tracking system is a non-optimized framework. It can be expected that using multi-threading/core techniques or hardware solutions can accelerate the system by factors.

REFERENCES

[1] Palti-Wassermann, D., Brukstein, A. M., and Beyar, R. P., "Identifying and tracking a guide wire in the coronary arteriesduring angioplasty from x-ray images," *IEEE Trans. on Biomedical Engineering* **44**, 152–164 (February 1997).

[2] Lin, Q., *Enhancement, Extraction, and Visualization of 3D Volume Data*, PhD thesis, Linkoeping University, Sweden (2003).

[3] Rosin, P. L. and West, G. A. W., "Salience distance transform," *Graphical Models and Image Processing* **57**, 483–521 (November 1995).

[4] Frangi, A. F., Niessen, W. J., Vincken, K. L., and Viergever, M. A., "Multiscale vessel enhancement filtering," *Lecture Notes in Computer Science* **1496**, 130–137 (1998).

[5] Klein, A., Lee, F., and Amini, A., "Quantitative coronary angiography with deformable spline models," *IEEE Trans. on Medical Imaging* **16**, 468–482 (October 1997).

[6] Poli, R. and Valli, G., "An algorithm for real-time vessel enhancement and detection," *Comput. Meth. and Programs Biomed* **52**, 1–22 (1997).

[7] Kutka, R. and Stier, S., "Extraction of line properties based on direction fields," *IEEE Trans. on Medical Imaging* **15**, 51–58 (January 1996).

[8] Sato, Y., Nakajima, S., Atsumi, H., Yoshida, S., Koller, T., Gerig, G., and Kikinis, R., "Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images," *Medical Image Analysis* **2**(2), 143–168 (1998).

[9] Barbu, A., Athitsos, V., Georgescu, B. Boehm, S., Durlak, P., and Comaniciu, D., "Hierarchical leraning of curves: Application to guidewire localization in fluoroscopy.," in [*IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], (2007).

[10] Baert, S., Viergever, M., and Niessen, W. J., "Guide-wire tracking during endovascular interventions," *IEEE Trans. on Medical Imaging* **22**, 965–972 (August 2003).

[11] Mortensen, E., Morse, B., Barrett, W., and Udupa, J., "Adaptive boundary detection using 'live-wire' two-dimensionaldynamic programming.," in [*Proceedings Computers in Cardiology*], 635–638 (1992).

[12] Dijkstra, E. W., "A note on two problems in connexion with graphs," *Numerische Mathematik* **1**, 269–271 (1959).

[13] Otsu, N., "A threshold selection method from grey level histograms," *IEEE Trans. Systems, Man and Cybernetics* **9**(1), 377–393 (1979).

[14] Koller, T., Gerig, G., Szekely, G., and Dettwiler, D., "Multiscale detection of curvilinear structures in 2d and 3d image data," in [*ICCV*], 864–869 (1995).

[15] Hoschek, J. and Lasser, D., [*Fundamentals of Computer Aided Geometric Design*], Teubner, Stuttgart (1993).

[16] Takemura, A., Hoffmann, K., Suzuki, M., Wang, Z., Rangwala, H., Harauchi, H., Rudin, S., and Umeda, T., "An algorithm for tracking microcatheters in fluoroscopy," *Journal of Digital Imaging* **21**(1), 99–108 (2008).