Hierarchical Learning of Curves
Application to Guidewire Localization in Fluoroscopy

Adrian Barbu[1], Vassilis Athitsos[2], Bogdan Georgescu[1], Stefan Boehm[3], Peter Durlak[3], Dorin Comaniciu[1]
[1]Siemens Corporate Research, 755 College Rd. E, Princeton, NJ 08540, USA
[2]Boston University, Boston, MA 02215, USA
[3]Siemens MED-AX, Siemensstr. 1, Forchheim, Germany

Abstract

In this paper we present a method for learning a curve model for detection and segmentation by closely integrating a hierarchical curve representation using generative and discriminative models with a hierarchical inference algorithm. We apply this method to the problem of automatic localization of the guidewire in fluoroscopic sequences. In fluoroscopic sequences, the guidewire appears as a hardly visible, non-rigid one-dimensional curve. Our paper has three main contributions. Firstly, we present a novel method to learn the complex shape and appearance of a free-form curve using a hierarchical model of curves of increasing degrees of complexity and a database of manual annotations. Secondly, we present a novel computational paradigm in the context of Marginal Space Learning, in which the algorithm is closely integrated with the hierarchical representation to obtain fast parameter inference. Thirdly, to our knowledge this is the first full system which robustly localizes the whole guidewire and has extensive validation on hundreds of frames. We present very good quantitative and qualitative results on real fluoroscopic video sequences, obtained in just one second per frame.

1. Introduction

Detection and segmentation of wire-like structures is a challenging problem with many practical applications in both medical imaging and computer vision. Our main interest is the detection and segmentation of the guidewire from fluoroscopy images used during coronary angioplasty, a medical procedure used to restore blood flow through clogged coronary arteries. During this minimally-invasive procedure, a catheter containing a guidewire is inserted through an artery in the thigh, and guided by the cardiologist until it reaches the blocked coronary artery. Then, a catheter with a deflated balloon is inserted along the wire and guided so that the balloon reaches the blockage. At that point, the balloon is inflated and deflated several times so as to unblock the artery. A device called a stent is often placed at that position in order to keep the artery from getting blocked again. Throughout this procedure, the cardiologist uses fluoroscopic images to monitor the position of the catheter, guidewire, balloon and stent. Fluoroscopic images are x-ray images collected at a rate of several frames per second. In order to reduce the patient's exposure to x-ray radiation, the x-ray dosage is kept low and as a result, the images tend to have low contrast and include a large amount of noise.
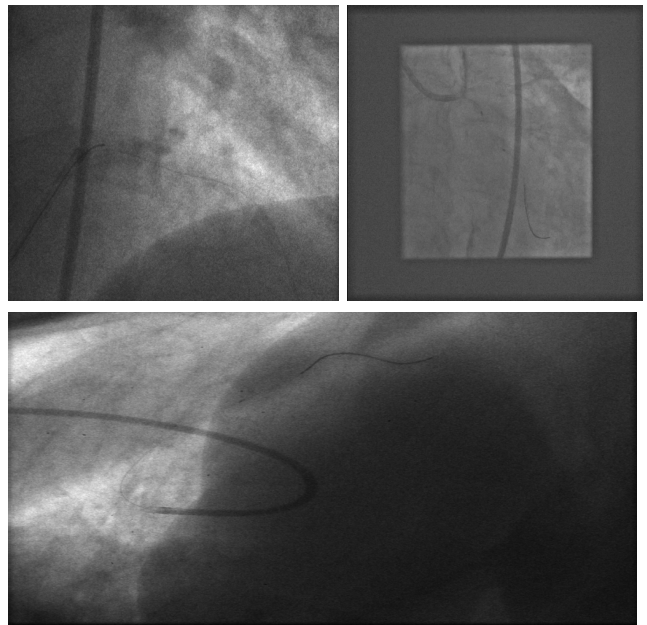

Figure 1. Example frames from fluoroscopic video sequences displaying the catheter, guidewire and wire tip.

As Figure 1 illustrates, it is often very hard to distinguish the objects of interest, in particular the catheter and guidewire, in such images. Automatic detection and tracking of the guidewire can greatly aid in enhancing the visualization quality of fluoroscopic data, while minimizing the exposure of the patient to x-ray radiation. In addition,

accurate localization of the guidewire can provide useful information for inferring 3D structure in bi-plane systems, allowing precise navigation of the guidewire tip through the arterial system.

The physical shape of the guidewire can be represented as a one-dimensional curve in 3D space. The projection of this shape onto the image plane can be represented as a one-dimensional curve in two dimensions. The shape of this curve is highly non-rigid, and a representation of the shape would require a large number of parameters. Detecting such a curve automatically is a challenging problem because of the complexity of finding optimal parameters in a high-dimensional space.

Previous work for guidewire detection [2, 12] used filtering techniques to enhance the guidewire. The results obtained in [12] were in the form of a set of pixels, which could sometimes be disconnected, while [2] used splines to track the wire, but only concentrated on the wire tip, which has much better visibility than the guidewire. The guidewire was also detected in [14], as a set of pixels, using a Hessian filter, with the purpose of adaptive filtering for image quality enhancement. Another approach to guidewire detection [11] treats the problem as a minimum cost path and uses a Fast Marching Algorithm for inference, but is only validated on 5 images and only detects the guidewire tip.

There is also a large amount of work in the field of curve modeling using differential geometry [13, 7], with advanced and generic curve models but without efficient inference algorithms and no robustness evaluation on a large dataset.

In comparison, our method is specialized for localizing the guidewire in Fluoroscopic images, and takes into consideration many specific elements that constrain the problem (noise patterns, shape models, scale, etc). Moreover, by using a large annotated database, a hierarchical representation and a hierarchical computational model, we can obtain robust results with great computational efficiency.

From an energy minimization perspective, our algorithm can be considered as an energy based learning method [10] for the full guidewire model, but the search space for the optimal parameters is largely restricted by all the previous levels of the hierarchy, increasing speed by many degrees of magnitude. Moreover, since the search space is restricted using the training data, it is unlikely for the global optimum to be missed. Other approaches to curve localization using energy minimization [6] use a global additive energy function and Dynamic Programming, this way being restricted in the form of the energy function and therefore in the system performance.

The diagram of our hierarchical model is illustrated in Figure 2. There are conceptually three levels, the first level being the low level of ridge (segment) detection, the intermediate level modeling curves with a range of parameters and the highest level representing the whole guidewire. In

our database-guided approach, we maintain a database of more than 700 frames in which the guidewire, catheter, wire tip and stent have been manually annotated. Example of such annotations are in Figure 3. We divided the database into two disjoint sets, one for training and one for testing.
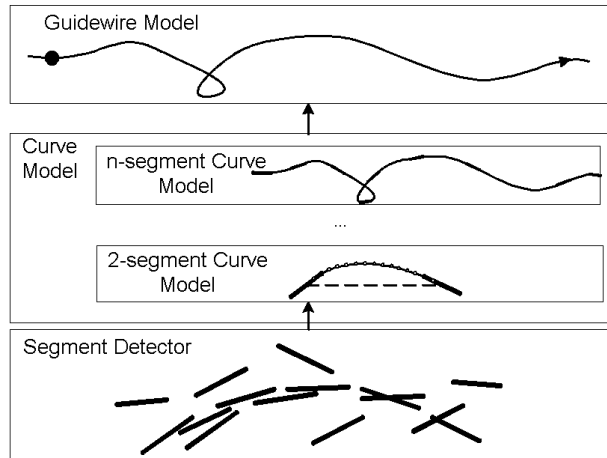

Figure 2. The diagram of our hierarchical approach.

We make no claims of optimality, but instead we verify our approach on more than 500 real fluoroscopic images and obtain a statistical measure of the localization error.
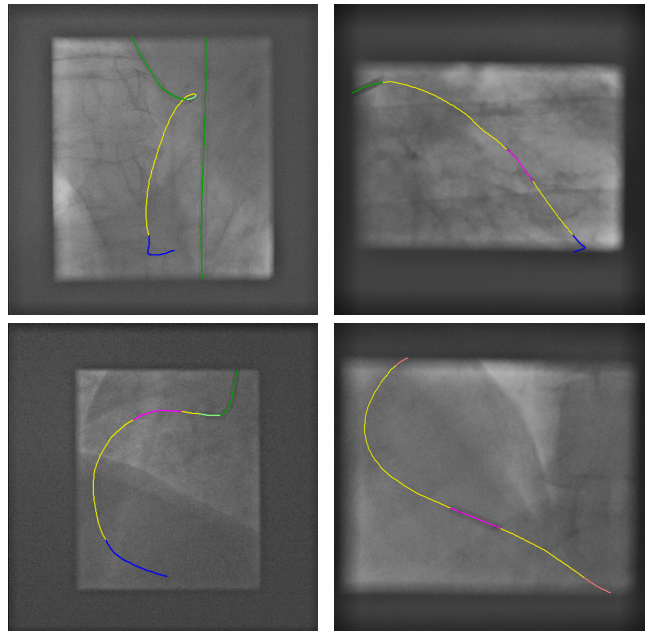

Figure 3. Example of annotations containing the guidewire (yellow), barely visible guidewire (brown), catheter (green), wire tip (blue) and stent (pink).

2. Marginal Space Learning

Many problems require the fast estimation of a large number of parameters. In this paper, the full guidewire model is controlled by about one hundred parameters, making any kind of full search practically impossible.

Most approaches [6, 11] handle so many parameters through a descriptive model (e.g. Markov Random Field) and restrict the energy function to have specific forms to apply fast inference algorithms. For example [6] uses an additive form of energy in order to use Dynamic Programming while [11] uses a specific cost function to apply a Fast Marching Algorithm variant. This limits performance because the restricted type of energy function cannot handle all the variability existent in natural images.

Learning based methods can model the images more accurately and usually handle the large number of parameters using a coarse-to-fine strategy [1]. At all the steps, the dimensionality of the search space is the same, but the space granularity varies. This approach cannot be used in our method, since the guidewire is not visible when the image is reduced in size and can be easily missed if large steps are used in the grid search.

In *Marginal Space Learning*, we propose a novel approach in which the dimensionality of the search space is gradually increased. Let Ω be the space where the solution to the given problem exists and let P_Ω be the true probability that needs to be learned. The learning and computation are performed in a sequence of marginal spaces

$$\Omega_1 \subset \Omega_2 \subset ... \subset \Omega_n = \Omega \tag{1}$$

such that Ω_1 is a low dimensional space (e.g. 3-dimensional in our guidewire application), and for each k, $\dim(\Omega_k) - \dim(\Omega_{k-1})$ is small. The marginal spaces are chosen in such a way that the marginal probabilities

$$P_{\Omega_k}(\theta) = \int_{X \perp \Omega_k} P_\Omega(\theta, x)dx \tag{2}$$

have small entropies, which is reflected in the fact that the learning tasks are easy. A search in the marginal space Ω_1 using the learned probability model finds a subspace $\Pi_1 \subset \Omega_1$ containing the most probable values and discards the rest of the space. The restricted marginal space Π_1 is then extended to $\Pi_1^e = \Pi_1 \times X_1 \subset \Omega_2$. Another stage of learning and detection is performed on Π_1^e obtaining a restricted marginal space $\Pi_2 \subset \Omega_2$ and the procedure is repeated until the full space Ω is reached.

At each step, the restricted space Π_k is one or two degrees of magnitude smaller than $\Pi_{k-1} \times X_k$, thus obtaining a restricted space n to $2n$ degrees of magnitude smaller than Ω. This reflects in a very efficient algorithm with minimal loss in performance.

For our guidewire localization problem, we use a joint hierarchical model for the curve shape and appearance, closely following the hierarchy of subspaces (1). The initial space Ω_1 is the 3-dimensional space of short segments with position and orientation while for each $k > 1$, Ω_k models a longer curve than Ω_{k-1} by extending it with 7 dimensions.

There is a difference between a model for the whole guidewire and a model for a (potentially long) part of a guidewire. This is because a full guidewire model uses the contextual information that the guidewire usually starts from a catheter and ends in a guidewire tip, both structures being very visible. In the Marginal Space Learning perspective, the models for partial guidewires can be regarded as the path to reach the full guidewire model.

3. Hierarchical Guidewire Model

The detectors at all levels of the hierarchical model are trained using the Probabilistic Boosting Tree (PBT), previously developed in our lab for other projects [15]. The PBT is a method to learn a binary tree from positive and negative samples and to assign a probability to any given sample by integrating the responses from the tree nodes. Each node of the tree is a strong classifier boosted from a number of weak classifiers (features). The PBT is a very powerful and flexible learning method, easy to train and to control against overfitting. Please follow [15] for more details.

To improve speed, in our hierarchical model each of the three levels only communicates with the previous level. This way, the information reaching each level comes in a condensed form through a vocabulary which becomes smaller as the level of the hierarchy increases.

3.1. Low Level Segment Detector

The first level of our system is a ridge detector aimed at detecting the simplest types of curves, namely short line segments of constant length. Such a curve has three parameters (x, y, θ), where (x, y) the segment center location and $\theta \in [-90, 90]$ is the segment orientation. The space of the orientations is discretized into 30 values.
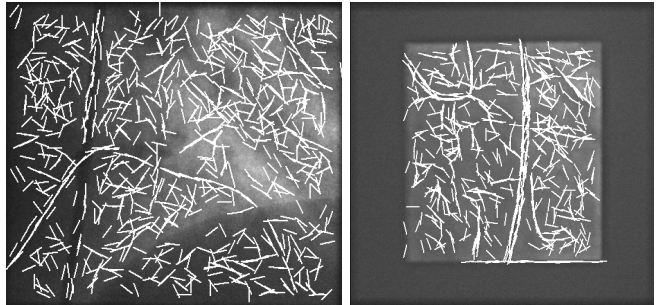

Figure 4. Low level detection for the upper images of Fig. 1.

The segment detector uses Haar features and integral images computed for all the 30 possible discrete image rotations. There are 22 types of Haar features, chosen appropriately for the task of detecting 1 dimensional structures. The Haar features are restricted to a window of size $W_x \times W_y$ (see Table 2) centered around the segment sample. There are about 100,000 features for this level.

The positive samples for the segment detector are segments on the visible guidewire (shown in yellow in Figure 3). The negative samples are chosen to be at distance at

least D_{neg} from the annotation. This way we obtain about 166k positives and 3.6 million negatives.

The detector is a PBT with five levels, of which the first three are enforced as cascade. In Figure 4 is shown the output of the segment detector.

In [4], the authors also use PBT and different types of features including Haar features to detect edges and ridges. The difference is that they did not align the edge orientations, and therefore the learning is much harder, the PBT is much larger and prone to overfitting.

We performed a comparative evaluation of our learning based ridge detection method and detection by Steerable Filters [5], tuned for guidewire detection. The results are summarized in Table 1. The error measures are described in section 5.

Detection Method	Missed Detection	False Detection
Steerable filters	0.11	0.90
Learning based	0.07	0.86

Table 1. Comparison of ridge detection using Steerable Filters and our learning based approach.

It is clear that our learning based approach gives much smaller missed and false detections than the Steerable Filters. This is because the guidewire appearance is more complex than the ridge model in [5].

3.2. Hierarchical Curve Model for Shape and Appearance

The Curve Model is designed to handle increasingly longer curves which can ultimately contain the whole guidewire. The curve shape is controlled by a number of segments, obtained from the Segment Detector 3.1, as shown in Figure 5. This was called discrete trace in [13] and can be considered an assembly of parts [3], but without an additive total cost.

Because the guidewire can have a wide range of lengths, the number of control segments is not fixed and we train specific models for each such number.
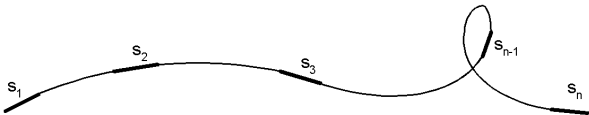

Figure 5. The Curve Model is controlled by a number of segments from the first level of detection.

To obtain a balance between the degree of generality obtained using descriptive models (Markov Random Fields) and the capacity to adequately constrain the shape space by generative models (PCA), we model the curve shape $C(s_1, ..., s_n)$ deterministically from the control segments $s_1, ..., s_n$ as described in 3.2.1 and we verify the obtained curve using a discriminating model based on shape and appearance described in 3.2.3.

The Curve Detection algorithm starts by constructing 2-segment curves using the detected segments from the Seg-

ment Detector 3.1 as control points, as described in Section 3.2.1. Then for each 2-segment curve, its probability is computed as in Section 3.2.2. Based on their probabilities, the most promising 2-segment curves are extended to 3-curves using again segments from Segment Detector 3.1 as control points, and the PCA shape model from Section 3.2.1. For each 3-segment curve, its probability is computed as in Section 3.2.3. The process of extending the most promisisng curves and computing their probabilities is repeated for a fixed number of steps.

In a fashion similar to Dynamic Programming, at each level we keep at most one curve between any given pair of line segments. This simplification largely limits the number of detections at each level and increases computational efficiency with minimal performance loss.

3.2.1 The PCA curve shape inference

The shape of the 2-segment curves is modeled using a PCA model. These curves are divided into $N - 1$ (see Table 2) equally distant segments, and thus approximated with N equally distant points. The collected samples from the training annotations are subsampled to N equidistant points. The samples were aligned by rotating and translating them so that the endpoints have coordinates $(-2(N-1), 0)$ and $(2(N-1), 0)$. Then Principal Component Analysis was performed and an evaluation revealed that 99.9% of the samples cou   s.
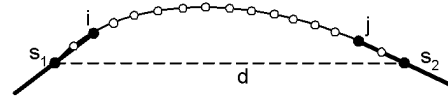

Figure 6. The 2-segment PCA curves are constructed deterministically from pairs of segments s_1, s_2 using two control points, shown in black.

To infer the shape of a 2-segment curve, the PCA coefficients are obtained deterministically from the two control segments s_1, s_2 as illustrated in Figure 6. The segments are simultaneously rotated, translated and scaled by the same transformation R to place their centers at locations $(-2(N-1), 0)$ and $(2(N-1), 0)$. Then the positions (x_1, y_1) and (x_2, y_2) of the segment points at distance D_{seg} from the centers are matched to the intermediate points with index $i = i_0$ and $j = N - i_0$, as shown in Fig. 6.

Let V^x, V^y, M^x, M^y be the x and y-eigenvector matrix (of size $Nx4$) and the mean shapes. Denoting by $A(k)$ the k-th line of matrix A, the PCA coefficients X of the curve are obtained by solving the linear system:

$$\begin{pmatrix} V^x(i) \\ V^y(i) \\ V^x(j) \\ V^y(j) \end{pmatrix} X = \begin{pmatrix} x_1 - M^x(i) \\ y_1 - M^y(i) \\ x_2 - M^x(j) \\ y_2 - M^y(j) \end{pmatrix} \quad (3)$$

Then all the N points of the obtained curve $C = M +$

VX are moved to the true location by the inverse transformation R^{-1}, obtaining the curve $C(s_{-}, s_{-})$
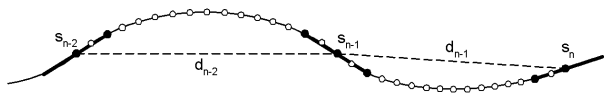

Figure 7. The curves are extended by concatenation of the 2-segment PCA curves.

The shape of a n-segment curve $C(s_1, ..., s_n), n \geq 3$ is constructed by concatenating the 2-segment curves $C(s_{k-1}, s_k)$ for all $k \leq n$, as illustrated in Figure 7.

3.2.2 Trained 2-segment Curve Classifier

After the 2-segment PCA curves have been constructed, a discriminative joint shape and appearance model is trained using the PBT.

To gain computational efficiency, we construct the appearance model using only information from the Segment Detector level 3.1, instead of going back to the original data.

For the 2-segment curve level, the information from the Segment Detector comes in the form of a 2 dimensional map of the computed segment probabilities. Based on this map, the features for training the PBT classifier are:

1. The PCA parameters of the 2-segment curve.

2. The probability of the best segment at different relative locations (along and perpendicular) to the curve.

3. The dot product of the orientation of the best segment at any of the locations above and the curve orientation at the projection location.

4. The product of the two corresponding quantities from 2 and 3 above.

5. The size of the largest gaps of the thresholded probability map along the curve, sorted in decreasing order.
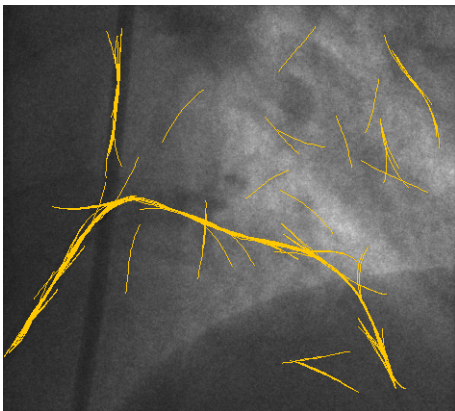

Figure 8. Example of the best 1000 2-segment PCA curves.

As one can observe, the feature pool contains features for both shape and appearance, and uses the probability map

obtained from the Low Level Segment Detector as a condensed form of the appearance.

Using these features, we trained a PBT with 5 levels, of which the first two enforced as cascade, and starting with 10 weak classifiers per node. The positive and negative samples for training are short PCA curves constructed as described in 3.2.1. The positives are the curves with maximum distance D_{pos} from annotation, while the negatives have distance at least D_{neg} from annotation. This way we obtained 26,000 positives and 3 million negatives.

We also trained a model in which we added to the feature pool Haar features at many locations along the curve. We observed that very few Haar features were picked by the training algorithm, and that the performance gain was insignificant. In Figure 8 we show the 1000 2-segment curves with the highest probability for the left image of Figure 1.

3.2.3 Trained n-segment Curve Classifier

For each $2 < n \leq n_{max}$, we construct a classifier designed to model the shape and appearance of n-segment curves. The classifiers are constructed recursively, the n-segment curve classifier depending on all the k-segment curve classifiers with $k < n$.
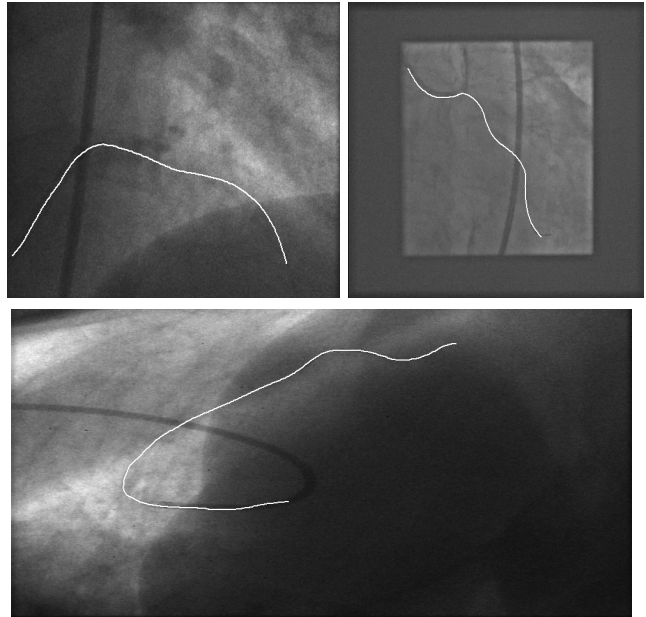

Figure 9. Results of hierarchical curve localization for the images in Figure 1.

For computational efficiency, the n-segment curve classifier is trained using PBT based on the following features:

1. The features from 3.2.2 of all the $n-1$ curve segments from which the curve is composed.

2. The differences $c_i^j - c_i^k, i \in \{1, ..., 4\}, j, k \in 1, ..., n-1$ between the corresponding PCA parameters of any two curve segments j, k.

3. The probabilities of all the $2, 3, ..., n-1$-segment sub-curves.

4. Products of probabilities of disjoint subcurves that when concatenated give the whole curve.

The positives and negatives at each level are obtained by extending the detection results from the previous level, and then keeping as positives samples those sufficiently close to the annotation and as negatives samples those sufficiently far from annotation.

In Figure 9 we show the curve with the highest probability for each of the images in Figure 1.

3.3. Full Guidewire Model

The guidewire model extends the curve model with two parameters, the position x_B, x_E of the guidewire starting and end points on the curve segments $C(s_1, s_2)$ and $C(s_{n-1}, s_n)$. Thus the guidewire is a curve $G(s_1, ..., s_n, x_B, x_E)$, fully specified by the control segments $s_1, ..., s_n$ and the endpoints x_B, x_E. We train an endpoint detector, using the same technique as in Section 3.1.
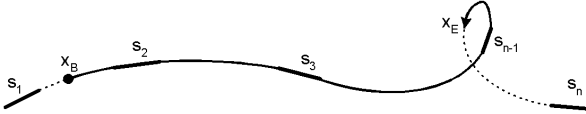

Figure 10. The guidewire model enhances the curve model with a classifier trained to recognize the starting point x_B (dot) and ending point x_E (arrow) of the guidewire.

This way we obtain a probability $P_E(x)$, trained to recognize the guidewire endpoints. The whole guidewire probability is then:

$$P(G(s_1, ..., s_n, x_B, x_E)) = P(C(s_1, ..., s_n))P_E(x_B)P_E(x_E)$$
(4)

From each level of the curve hierarchy, the curve with highest probability is augmented to the guidewire model and the parameters x_B, x_E are searched on the first and last curve segments. The guidewire with the highest probability is reported as the final localization result.

4. Efficient Implementation

There are a few implementation details to make our system faster. We present them in this section. In the Low Level Detector 3.1, we use non-maximal suppression to keep only 1000 segments as candidate control points.

At all the levels of the Hierarchical Curve Model, we perform a fast initial screening for potentially good candidates for curve construction and extension. The fast screening is performed using a 3-dimensional probability model $P(s_1, s_2) = P^{\text{on}}(s1, s2)/P^{\text{off}}(s_1, s_2)$ based on pairs $(s1, s_2)$ of segments obtained from the Low Level Segment Detector from 3.1. We construct unnormalized histograms

H^{on} and H^{off} where H^{on} collects the statistics of all segment pairs on the guidewire or stent and H^{off} collects the statistics of the background. The three dimensions of the histograms are the distance d between segment centers and the two angles a_1, a_2 between the segments' orientations and the orientation of the segment connecting the centers. This is illustrated in Figure 11.
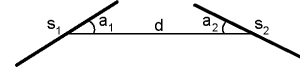

Figure 11. The 3-dimensional histograms measure the distance d between the segment centers and the two relative angles a_1, a_2.

Then the marginal probability $P(s_1, s_2)$ for fast screening is

$$P(s_1, s_2) = \frac{H^{\text{on}}(s1, s2)}{H^{\text{on}}(s1, s2) + H^{\text{off}}(s1, s2)}$$
(5)

In Table 2, we collect the many parameters that are used at different levels of the algorithm.

Parameter Name	Symbol	Value
Level 0 window size	$W_x \times W_y$	41×15
Dist. negatives from annotation	D_{neg}	4
Dist. positives from annotation	D_{pos}	2
Number of PCA intermediary pts.	N	17
Distance on segment for PCA	D_{seg}	8
Index for PCA shape inference	i_0	2

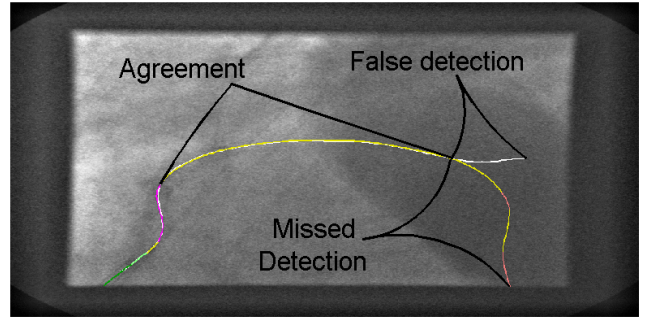Table 2. Parameters of our algorithm.

5. Results

Figure 12. Illustration of the error measures used for evaluation of the system's performance.

We present qualitative and quantitative results of our method. To present quantitative results, we need an error measure of the detection result compared to the annotation. However, we cannot measure the error in terms of detection rate and false alarm, because it can happen that parts of the detection result are correct while some other parts are erroneous. To measure how much of the wire is correct and how much is erroneous, we compute two quantities:

1. *Missed detection* - the percentage of guidewire pixels (strong or weak) of the annotation that were at distance at least 3 pixels (0.6mm) from the detection result.

2. *False detection* - the percentage of the detection result pixels that were at distance at least 3 pixels (0.6mm) from the annotation.

These two error measures are illustrated in Figure 12.

Using these error measures we obtained the results summarized in Table 3.

Set (No. sequences)	Missed	False Detection
Training (38)	0.24	0.12
Unseen (15)	0.17	0.05
Overall, (53)	0.22	0.10

Table 3. Evaluation results on the 38 training sequences and 15 unseen sequences totaling 535 images.

For comparison, Table 4 shows an evaluation using the same error measures on the result obtained using Steerable Filters [5].

Set (No. sequences)	Missed	False Detection
Training (38)	0.22	0.79
Unseen (15)	0.25	0.86
Overall, (53)	0.23	0.81

Table 4. Evaluation results using Steerable Filters.

We see that for approximately the same missed detection, our method has a considerably smaller false detection.

Qualitative results are shown in Figure 13 and 14. The average computation time is one second per frame on a 3.4GHz desktop PC with 2Gb of RAM. The software has good potential for further optimization.

6. Conclusion

In this paper we presented a hierarchical representational and computational model for the localization of the guidewire in fluoroscopic images. The hierarchical representational model offers advantages in the ability to enforce strong generative and discriminative priors, which together with our learning based approach is capable to obtain results even where the guidewire is invisible in large areas. The hierarchical computational model based on Marginal Space Learning allows to quickly discard large parts of the search space long before going to the full guidewire model, obtaining great computational speed.

To our knowledge, this is the first system to localize the whole guidewire and have validation on more than 500 frames. In [2], only the guidewire tip is tracked, a much more visible and easier to detect and track structure. Other methods for navigation [9] use a magnetic method for 3D tracking, with an error of about 6.5mm. Our image based method has an error of less than 1mm.

In the future, we plan to incorporate the motion coherence into our hierarchical framework to obtain an even more robust system. The main challenge is that parts of the guidewire can move hundreds of pixels between consecutive frames, making motion coherence a hard-to-define concept.

References

[1] Y. Amit, D. Geman and X. Fan, A coarse-to-fine strategy for multi-class shape detection, *IEEE Trans. PAMI* (28), 1606-1621, 2004

[2] S. Baert, M. Viergever and W. Niessen. Guide Wire Tracking during Endovascular Interventions *IEEE Trans Med Img* (22) 965-972, 2003.

[3] C. Batten. Algorithms for Optimal Assembly, *SSBC* 2004

[4] P. Dollr, Z. Tu, and S. Belongie, Supervised Learning of Edges and Object Boundaries, *CVPR*, 2006

[5] W. T. Freeman and E. H. Adelson. The Design and Use of Steerable Filters, *PAMI* 1991

[6] D. Geiger, A. Gupta, L. Costa and J. Vlontzos. Dynamic Programming for Detecting, Tracking, and Matching Deformable Contours *PAMI* 1995

[7] L. Iverson and S.W. Zucker, Logical/linear operators for image curves, *IEEE Trans. PAMI*, (17), 982-996, 1995.

[8] S. Konishi, A. L. Yuille, J. M. Coughlan, and S. C. Zhu,. Statistical Edge Detection: Learning and Evaluating Edge Cues, *IEEE PAMI*, **25** (1): 57-74, 2003

[9] S. Krueger, H. Timinger, R. Grewer and J Borgert. Modality-integrated magnetic catheter tracking for x-ray vascular interventions *Phys. Med. Biol.*(50), 581597, 2005.

[10] Y. LeCun, S. Chopra, R. Hadsell, F.-J. Huang and M.-A. Ranzato, A Tutorial on Energy-Based Learning, *Predicting Structured Outputs*, Bakir et al. (eds), MIT Press 2006

[11] Q. Lin, Enhancement, Extraction, and Visualization of 3D Volume Data. PhD Thesis, pp 125-132, Linkping University, Sweden

[12] D. Palti-Wasserman, A. M. Brukstein, and R. P. Beyar, Identifying and tracking a guide wire in the coronary arteries during angioplasty from X-ray images, *IEEE Trans. Biomed. Eng.* (44) 152164, 1997.

[13] P. Parent and S.W. Zucker, Trace Inference, Curvature Consistency, and Curve Detection, IEEE Trans. PAMI, (11), 823-839, 1989.

[14] G. Schoonenberg, M. Schrijver, Q. Duan, R. Kemkers and A. Laine - Adaptive spatial-temporal filtering applied to x-ray fluoroscopy angiography *SPIE* 2005

[15] Z. Tu, Probabilistic Boosting-Tree: Learning Discriminative Models for Classification, Recognition, and Clustering, *ICCV*, 2005.
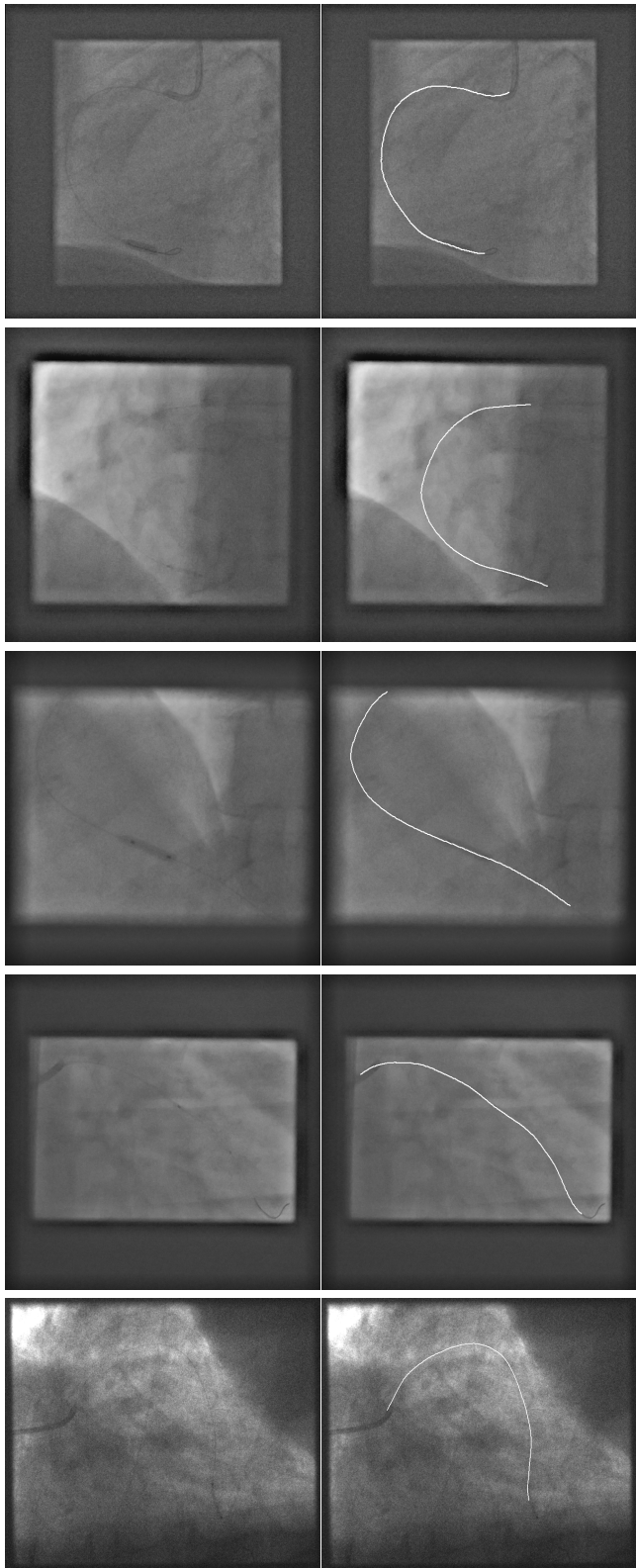
Figure 13. Results obtained using our hierarchical model. Left: original image, right: guidewire localization result.
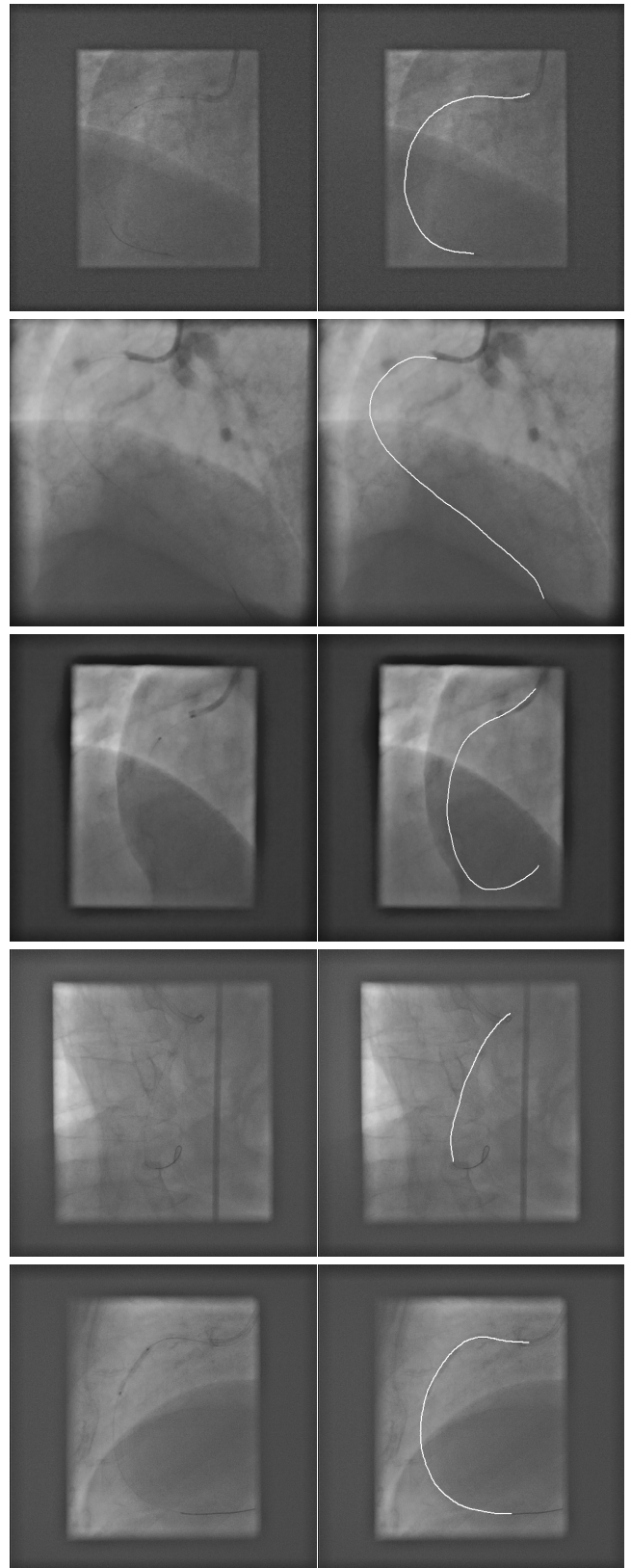

Figure 14. More results obtained using our hierarchical model. Left: original image, right: guidewire localization result.