Motion Estimation of Skeletonized Angiographic Images Using Elastic Registration

Brian C. S. Tom, Student Member, IEEE, Serafim N. Efstratiadis, Member, IEEE, and Aggelos K. Katsaggelos, Senior Member, IEEE

Abstract—In this paper an approach for estimating the motion of arteries in digital angiographic image sequences is proposed. Binary skeleton images are registered using an elastic registration algorithm in order to estimate the motion of the corresponding arteries. This algorithm operates recursively on the skeleton images by considering an autoregressive (AR) model of the deformation in conjunction with a dynamic programming (DP) algorithm. The AR model is used at the pixel level and provides a suitable cost function to DP through the innovation process. In addition, a moving average (MA) model for the motion of the entire skeleton is used in combination with the local AR model for improved registration results. The performance of this motion estimation method is demonstrated on simulated and real digital angiographic image sequences. It is shown that motion estimation using elastic registration of skeletons is very successful especially with low contrast and noisy angiographic images.

I. INTRODUCTION

In MEDICAL IMAGING, much attention has been focused on estimating the motion of the heart from angiographic image sequences. Based on knowledge of the regional wall stress and local wall deformation, for example, cardiologists can better evaluate the presence of any abnormal cardiac functions. The majority of the motion estimation schemes used in angiographic images track some form of landmark, whether it be natural, such as bifurcation points [10], or artificial, such as radiopaque markers that have been injected into the system [7], or lead markers that have been sutured to the epicardium [16].

On the other hand, if no landmarks are used, displacement vector field (DVF) estimation techniques, which have been developed for processing video sequences, can be used. In such DVF estimation techniques a motion vector is assigned to each and every pixel or block of pixels [1], [2], [4], [5], [8], [18]. The application of gradient-based DVF techniques to angiographic image sequences has been reported in [17], [21], while the combination of landmark tracking and DVF estimation approaches has been proposed in [13].

The motion estimates resulting from the application of DVF estimation techniques to angiographic image sequences are not very accurate, for the following reasons [11], [12], [20]:

Manuscript received September 24, 1992; revised November 28, 1993. This paper is based on work supported in part by Siemens. This paper was recommended by Associate Editor N. Ayache.

B. Tom and A. Katsaggelos are at Northwestern University, Department of Electrical Engineering and Computer Science, McCormick School of Engineering, Evanston IL 60208 USA; e-mail agk@delta.eecs.nwu.edu.
S. Efstratiadis is at the Aristotle University, Department of Electrical Engineering, Thessaloniki, Greece.
IEEE Log Number 9403429.

(1) one of the basic assumptions in the derivation of the DVF estimation algorithms, that the moving objects are rigid, is clearly violated, since the motion of the heart and the coronary arteries is elastic; (2) angiographic image sequences are usually of low contrast with no sharp edges, which make the matching criterion utilized by block matching techniques easier to fail and the spatial gradients required by the gradient-based pel-recursive approaches more sensitive to errors; and (3) there is usually noise present in the angiographic images which is signal-dependent due to the physics of generating such images. Most DVF estimation algorithms do not take the presence of even additive noise into account. Therefore, this is another source of error in the final DVF estimates.

For the above reasons a different approach is followed in our work in estimating the motion of the coronary arteries or the ventricle boundaries. In this paper, it is assumed that the angiographic sequences have been segmented and thinned to obtain the skeleton of the objects of interest. The problem of estimating the motion or the DVF of the angiograms reduces to elastically matching the skeletons of two successive frames. In previous work [12], registration of skeletonized frames was attempted by using a block matching algorithm that was applied directly to the skeleton pixels. However, the results obtained were not very satisfactory, due to the fact that the objects (skeletons) to be matched were elastically deformed.

Another approach, which provides the framework for the method proposed here, was developed by Maitre and Wu [15]. This method used dynamic programming (DP) to obtain an optimal match between elastically deformed contours. Since DP is an optimization technique, a cost function needs to be chosen. By modeling the estimated displacement vector as an auto-regressive (AR) model, the cost is taken as a function of the difference (error) between this estimated vector and the actual, real displacement vector found. The advantage of this AR-DP algorithm is that, unlike other existing methods, the computational cost does not depend upon the complexity of the geometric deformation. In addition, occluded regions can also be accounted for by storing these virtual nodes in a "refrigerator" [14]. However, its main limitation is its prerequisite that the working (or reference) image be expressible as an ordered sequence of primitives (pixels), which may be difficult to achieve.

Geiger and Vlontzos recently proposed an algorithm that also matches elastically-deformed contours [6]. Similar to Maitre's algorithm, it is also based on dynamic programming (DP), and can be viewed as a simplified version of [15]. Instead
of choosing a cost function based on the prediction errors from AR modeling, they formulate the cost based upon the following two constraints: 1) that the vector field be smooth (the difference between two successive displacement vectors is minimized), and 2) that the size of the vectors themselves is minimized. However, like [15], there is an inherent assumption that the contours be expressible as an ordered sequence of pixels.

Both of these approaches cannot match skeletons whose contours cannot be easily ordered, unless a (possibly costly) preprocessing step of decomposing the skeletons into its branches is used. In the general case of ordering angiograms (or, any other image) by raster scanning, it is clear that the resulting sequence of pixels does not necessarily preserve its connectedness. For example, in raster scanning a "Y", a pixel in the upper right branch would be scanned immediately after a pixel in the upper left branch (on the same row). Therefore, the two data points, while they are consecutive after raster scanning, are not actually connected together in the original image. The two algorithms listed above would encounter great difficulty in matching contours of this type.

In this paper, Maitre's algorithm is extended in order to register multibranched skeletonized images (such as the "Y"). In addition, we address the issue of bifurcation points, where two or more branches meet. Several experiments were performed to see how well the algorithm registers in these difficult cases.

The paper is outlined as follows. In Section II the problem of registering skeletons is mathematically formulated, and notation is introduced. The first two subsections of the next section, Section III, briefly describe Maitre's original AR-DP algorithm, with the remainder of the section covering the necessary extensions to this algorithm. In Section IV, experimental results are presented for registration of skeletons with both synthetic deformations and with real data. Finally, conclusions drawn from the research presented in this paper are covered in Section V.

II. PROBLEM FORMULATION

In estimating the motion of the objects of interest in angiographic image sequences, the following general approach is adopted. It is assumed that the image sequence has been thinned and segmented into their corresponding skeletons. The image sequences used in this paper to demonstrate the performance of the proposed method are digital angiograms courtesy of Evanston Hospital, Department of Cardiology. Three frames of such a sequence, each of size 256 x 256, can be seen in Figs. 1, 2, and 3. The white boxes outline three different regions of interest, each of size 64 x 64. Each of these smaller images was segmented, and then thinned using the width-independent thinning algorithm of [3]. The segmented and thinned images can be seen in Figs. 4, 5, and 6 for each of the three different regions for each frame. Note that each image contains at least one bifurcation point, and at least three different branches. In the first 64 x 64 image sequence (Fig. 4), the "Y"-shaped branch represents two arteries, with a catheter running across it. Fig. 5 depicts a wire hook (the oval) along with an artery, and Fig. 6 depicts a branching artery (only the first and second frames are shown). After segmenting and thinning the images, the motion of the skeleton pattern is then estimated by finding the correspondences between the pixels of skeletons of successive frames.
Let the pixels (or primitives) of the binary skeleton images be described by their spatial location. Let $M$ and $N$ be the number of pixels in the current and previous skeleton images, respectively, after raster scanning. Furthermore, let
\begin{align*}
z_1(m) &= [x_1(m), y_1(m)], m = 1, \ldots, M \\
z_2(n) &= [x_2(n), y_2(n)], n = 1, \ldots, N
\end{align*}
where $z_k(m)$ is the $m$-th pixel in skeleton $S_k$, and $x_k(m)$ and $y_k(m)$ describe its vertical and horizontal positions, respectively. The goal of registration is to find for every pixel in the first frame, $z_1(m)$, $m = 1, 2, \ldots, M$, a corresponding pixel, $z_2(n)$, $n = 1, 2, \ldots, N$ in the next frame,
\[z_1(m) \rightarrow z_2(n)\]
where $n$ is given by
\[n = u(m)\]
where $u$ is the mapping function that must be determined. Once a correspondence has been found, estimates for the displacement vectors between the corresponding pixels $m$ and $n$ of successive frames can be found by
\[d(m, n) = z_1(m) - z_2(n).\]

It should be mentioned that image registration can be performed either on the pixel level or on the primitive level, where each primitive is a block of pixels, usually arranged in some ordered fashion [14]. In this paper, registration is done on the pixel level for simplicity; therefore, primitives and pixels will henceforth be used interchangeably.

### III. ELASTIC REGISTRATION OF MULTIBRANCHED SKELETONS

#### A. Improved Dynamic Programming

Dynamic Programming (DP) is a general optimization tool that solves an $N$-variable problem by breaking it into $N$-cascaded monovariable problems under the assumption of optimality. It is useful to view DP as a graph, with the two axes being the ordered sequence of pixels of $S_1$ and $S_2$. Fig. 7 shows a typical DP graph where each grid point represents a matched pair or node, \([z_1(m), z_2(n)]\), where $z_2(n)$ is matched to $z_1(m)$. Note that due to deformation, possible occlusions and the presence of noise in the skeletonization, it is likely that $M \neq N$.

At each stage, that is, for each pixel along the horizontal axis, $z_1(m)$, $m = 1, 2, \ldots, M$, there are $N$ different paths, or possible solutions, and DP finds the best path up to that stage by taking the path with the minimum cumulative cost up to that stage, $m$. The overall solution is found by taking the path with the minimum cost at the last stage, $m = M$. The costs in linking nodes to a path are fully described in Section III.B.

The algorithm operates as follows. Starting at step $m = 1$, the cost for every possible pairing \([z_1(1), z_2(n)]\), $n = 1, 2, \ldots, N$, is computed. At every subsequent step $m (m \geq 2)$ the path of minimum cumulative cost (or best path) ending at each node \([z_1(m), z_2(n)]\), $n = 1, 2, \ldots, N$, is determined by finding the cost of linking that node to every node of the previous stage \([z_1(m-1), z_2(\ell)]\), $\ell = 1, 2, \ldots, N$, and taking the path that yields the minimum cumulative cost. Let $c(m, n, m-1, u(m-1))$ be the elementary (or step) cost found by linking node \([z_1(m), z_2(n)]\) with node \([z_1(m-1), z_2(u(m-1))])$ of the previous stage, and $C(m, n)$ the minimum cumulative cost of the path ending up to that node \([z_1(m), z_2(n)]\). The step-cost is added to $C(m-1, u(m-1))$ to give the cumulative cost $Q(m, n, m-1, u(m-1))$. For every node \([z_1(m), z_2(n)]\) at stage $m$, we have $N$ cumulative costs, $Q(m, n, m-1, u(m-1))$, where $u(m-1) = 1, 2, \ldots, N$, and $C(m, n)$ is defined by
\begin{align*}
C(m, n) &= \min_{u(m-1)} \{Q(m, n, m-1, u(m-1))\} \\
&= \min_{u(m-1)} \{C(m-1, u(m-1)) + c(m, n, m-1, u(m-1))\}.
\end{align*}
The node \([z_1(m-1), z_2(u(m-1))]\) that yields the minimum cumulative cost, \(C(m, n)\), becomes the predecessor of node \([z_1(m), z_2(n)]\). This entire process is repeated for the \((m+1)\)-th pixel and the best path at the last stage \(m = M\) gives the overall optimum path.

Finally, let \(\{z_2(u_n(m-1)), z_2(u_n(m-2)), \ldots, z_2(u_n(1))\}\) be the pixels associated with the path that ends at node \([z_1(m), z_2(n)]\). Thus, all the nodes that belong to the path \(n\) are given by \(\{[z_1(1), z_2(u_n(1))], [z_1(2), z_2(u_n(2))], \ldots, [z_1(m), z_2(n)]\}\).

The skeletonized images frequently contain discontinued sections that are due to data occlusions, missing points or isolated section. These pixels arise from various obstructions in the original image, such as noise, X-ray imaging effects, etc. In such cases, the DP described above does not perform well and an improved DP method has been proposed [14]. The improvement comes from the idea of a “refrigerator,” which acts as an extension of the “real” nodes into “virtual” nodes. In other words, when no correspondence can be found, a virtual node is created and stored in the refrigerator. This virtual node contains all the information about the broken path; namely, the cumulative cost, the path length, and the AR model. During registration in the gap region, the virtual node is used as a signal to estimate the next correspondence. While there are no real pixels in the occluded region, the predicted node is used as a predecessor for future nodes, in place of the real node as the predecessor. The cost function for this bridging technique is more tolerant than the cost function described in (4).

Computationally, DP requires three nested loops. Two of these nested loops correspond to moving along the vertical axis of the DP graph, while the other loop represents movement along the horizontal axis. At every node \([z_1(m), z_2(n)]\), DP needs to find the cost of associating that node with every node at the previous stage, \([z_1(m-1), z_2(k)]\), \(k = 1, 2, \ldots, N\). Thus, the three nested loops would go from \(m = 1, 2, \ldots, M\), \(n = 1, 2, \ldots, N\), and \(k = 1, 2, \ldots, N\), which means that DP is on the order of \(O(M^2N^3)\).

B. AR Modeling of Displacement Vectors

In applying the DP algorithm in the registration of skeletons, a suitable step cost function \(c(m, n, m-1, u(m-1))\) to evaluate (4) needs to be determined. In skeleton registration, a correspondence or displacement vector is used to represent the change in the pixel location from one frame to the next. Let \(d(m, n)\) be the displacement vector between pixels \(z_1(m)\) and \(z_2(n)\), repeated here for convenience,

\[
d(m, n) = z_1(m) - z_2(n) = d_x(m, n) + jd_y(m, n)
\]

(5)

where \(j = \sqrt{-1}\). The real and imaginary parts of \(d(m, n)\) correspond to the displacement in the vertical and horizontal directions, respectively, or

\[
d_x(m, n) = x_1(m) - x_2(n) \quad \text{and} \quad d_y(m, n) = y_1(m) - y_2(n).
\]

(6)

DP provides several sequences of displacement vectors which belong to the paths and are candidates for an AR model.
At the same time, AR modeling of the displacement vectors along each path furnishes a convenient cost function to DP, through the process innovations. Auto-regressive modeling of the displacement field was also used in [5] in deriving a spatio-temporal gradient recursive DVF estimator.

By using an AR model to the correspondence vectors that belong to a path ending at node \( z_1(m), z_2(n) \), we can predict its displacement vector by

\[
\hat{d}(m, n) = - \sum_{\ell=1}^p a_\ell \hat{d}(m - \ell, u_n(m - \ell))
\]

(7)

where \( a_\ell \) are the AR coefficients. Then, the predicted location of the matching pixel is given by

\[
\hat{z}_2(m, n) = z_1(m) + \hat{d}(m, n)
\]

(8)

and the prediction error, \( e(m, n) \), is

\[
e(m, n) = z_2(n) - \hat{z}_2(m, n).
\]

(9)

By comparing the innovations to a prespecified threshold \( \tau \), a suitable similarity measure is

\[
\|e(m, n)\| \begin{cases} = 0 & \text{exact matching} \\ \leq \tau & \text{elastic matching} \\ > \tau & \text{no matching.} \end{cases}
\]

(10)

Finally, the step cost in (4) is evaluated as a function of the magnitude of the prediction error (or innovations), that is,

\[
e(m, n, m - 1, u_n(m - 1)) = \psi(\|e(m, n)\|).
\]

(11)

The error function \( \psi(\cdot) \) should be a quadratic or higher order function of the error norm, or can be determined experimentally as given in Table I. In case no correspondence is found a virtual node is created and stored in the refrigerator. For the virtual nodes a more tolerant error function is used, given in Table II.

Let us assume that at the final registration result \( m = M \) the \( n^\text{th} \) path had the minimum cumulative cost associated with it. Then, the registration result is given by the nodes \([z_1(m), z_2(u_n(m))])\), where \( m = 1, 2, \ldots, M \). Finally, the motion vectors \( \hat{d}(m) \) that correspond to the pixels of the first skeleton \( z_1(m) \) and \( z_2(u_n(m)) \) are given by

\[
\hat{d}(m) = z_1(m) - z_2(u_n(m)).
\]

(12)

In estimating the AR parameters, \( a_\ell \), the Burg (maximum entropy) algorithm is used. The advantage of this algorithm, compared to other algorithms, is that the coefficients \( a_\ell \) are determined directly from the observation vectors. These observation vectors, \( d(m, n) \), are found by DP alone, where for the first \( L \) pixels \( (m = 1, 2, \ldots, L) \), or the AR learning stage, the cost function is simply a nondecreasing function of the norm of the difference between two successive vectors. Then, Burg’s algorithm is applied on this set of \( L \) displacement vectors. In this paper, the order of the AR model was chosen as \( p = 3 \) and the size of the Burg input sequence was \( L = 10 \).

C. Causal Masking

A severe limitation of the algorithms presented in [6, 15] is that it can be applied only to ordered sequences of pixels or primitives. However, typical skeletonized angiographic images contain a large number of branches that depict a rather complicated pattern of blood vessels on the heart surface. One way that either Maitre’s or Geiger’s algorithms could register such a complicated image would be to first, decompose the skeleton images into its skeleton branches, next, register each branch separately, and then finally obtain the overall DVF by piecing together the vector fields of each individual branch.

While this is a logical extension of registering single-branch contours, it is not the approach taken in this paper, due to the following reasons. First, the automatic separation of the various branches may be a complicated preprocessing step, especially if a large number of branches exist. Second, if a large number of branches do exist, then their independent registration will be suboptimal, since the motion of the entire skeleton is not taken into account.

From Section III.B, a displacement vector for a given pixel is predicted based on previous vectors which correspond to the predecessors of that particular pixel, implying that accurate registration depends heavily upon each pixel’s predecessors. Without an intelligent searching strategy, the closest \( p \) predecessors of \( z_1(m) \) will be \( z_1(m - p), z_1(m - p + 1), \ldots, z_1(m - 1) \). Since these primitives might belong to a different branch from that of \( z_1(m) \), and might therefore move in a completely different direction, the registration results will not be very accurate. Therefore, it is desired that the predicted displacement vector, \( d(m, n) \), be strongly correlated to its predecessors, \( d(m - \ell, u_n(m - \ell)) \) in (7).

For the above reasons, causal masking is introduced in this section as an extension of Maitre’s algorithm in order to register multi-branch skeletons. Essentially, causal masking is a policy used in assigning a pixel’s predecessors. As shown in Fig. 8, the searching scheme is carried out within the pixels’ causal support such that only those pixels lying very close to the primitive are considered in finding its predecessors. Such a policy ensures that the predecessors are highly correlated to the current pixel, which results in improved registration. If \( p \) predecessors cannot be found, where \( p \) is the order of the
where $b_k$ is a complex coefficient and $v_k(m)$ is a complex vector indicating the error in the model. In other words, skeleton $\{z_k(m)\}$ is assumed to be the output of a simple complex system, which is described by a constant $b_k$, and the uncertainty in the location of the primitives represents the error term $v_k(m)$. By identifying or estimating $b_k$, we can get a prediction of point $z_2(m)$, denoted by $\hat{z}_2(m)$, based on the entire skeleton, where

$$\hat{z}_2(m) = b \cdot z_1(m)$$ (14)

and where the subscript $k$ has been dropped from the complex parameter, $b$. This predicted correspondent is less accurate than the $z_2(m)$ that was predicted by the AR model of the displacement vectors, but offers the advantage that it includes information about the deformation of the entire skeleton as a whole. Therefore, using the prediction of (14) large registration errors can be avoided due to the assumption of rigidity for the predicted skeletons.

In estimating $b_k$ the following formula is used

$$b_k = \frac{\sum_{m=1}^{\min(M,N)} z_k^* (m) z_{k-1}(m)}{\sum_{m=1}^{\min(M,N)} z_k^* (m) z_k(m)}$$ (15)

where $z_k^* (m)$ denotes the complex conjugate of $z_k(m)$. Finally, the error from using the global matching is given by

$$e_k(m, n) = \hat{z}_k(m) - z_k(n).$$ (16)

Since (16) works on a global level, it can be used in conjunction with the AR-DP algorithm covered in Sections III.A-B, which operates on a local level. Therefore, in determining corresponding primitives, we combine the two prediction models to obtain an improved cost function

$$c(m, n, m-1, u(m-1)) = \psi(|e_k(m, n)| + \lambda_{gm} \|e_k(m, n)\|),$$ (17)

where $\lambda_{gm}$ represents the weighting of the error due to global matching, and $e_k(m, n)$ and $\lambda_{gm}$ are given by (9) and (16), respectively. It should be noted that the cost function in Geiger’s algorithm is similar to (17), except that the two errors measure the smoothness of the displacement field and the size of the displacement vectors [6]. Note that, for $\lambda_{gm} = 0$ only the AR displacement vector model is used in the DP algorithm and the algorithm reduces to that of Sections III.A-B. When $\lambda_{gm} > 0$ both errors are affecting the registration. Also, since $e_k(m, n)$ can be significantly greater than $e_k(m, n)$, its contribution could dominate the total error in (17), resulting in a large step cost $c(m, n, m-1, u(m-1))$. By using this modified cost function the registration results are smoother than if using $\|e_k(m, n)\|$ alone in the above cost function. This is to be expected, since the motion of the entire skeleton is being modeled by just one parameter, $b_k$. Global matching by itself gives very rough matching but when combined with AR-DP gives better results than AR-DP alone.

The actual values of $\lambda_{gm}$ can be determined by using results from regularization theory [9], [10]. According to

D. Modeling the Global Deformation

In elastic registration with DP, the use of a local AR model for the displacement vectors provided a suitable error criterion. In some cases, however, the use of a local AR model has certain limitations since it cannot incorporate any information about the global deformation. To accomplish this, we propose to first formulate the global deformation by using a Moving Average (MA) model in the temporal direction for the successive skeletons $z_1(m)$ and $z_2(n)$, and then use this information in evaluating the step cost function, $c(m, n, m-1, u(m-1))$ [20]. That is, for each pixel of the first image $z_1(m)$ a predicted pixel $\hat{z}_2(m)$ is found based on the primitives of the entire image, instead of just upon the $p$ predecessors used in the AR case. Comparing this prediction $\hat{z}_2(m)$ to the closest pixel in the picture yields an additional similarity measure as a rough guideline for the correspondences.

Let us assume that the following model describes the deformation of the skeleton from the $(k - 1)$th frame to the $k$th frame

$$z_k(m) = b_k \cdot z_{k-1}(m) + v_k(m)$$ (13)
these results, the contribution of each error term in (17) towards the total error is weighted inversely proportional to its variance. Therefore, an error term with a higher variance will be weighted proportionately less than another error term whose variance is lower. In our experiments, it was found that the variance of $\|e_k(m,n)\|$ was between 20 and 100 times higher than the variance of $\|e_k(m,n)\|$, yielding $\lambda_{gm}$ to be between 0.01 and 0.05, agreeing with our choice for $\lambda_{gm}$.

It should be noted that the added computational cost from finding $b_k$ and the other set of innovations, $e_k(i,j)$, is not significant. This is because $e_k(i,j)$ is computed for every possible primitive pair combination ($O(MN^2)$), but $e_k(i,j)$ is computed only for every primitive ($O(MN)$). The simplicity of the moving average model ensures low computational cost, yet, at the same time, provides a good working estimate.

E. Elimination of Crossed Motion Vectors

One clear constraint in matching skeletons is that the ordering of the contour must be preserved. In other words, there should be no crossed motion vectors. This was called the monotonicity constraint in [6]. In images with a single-branch contour, such as those used in Geiger's and Maitre's algorithms, the sequence of pixels obtained by proceeding along the contour maintains the ordering present in the image. However, for more complicated images, this cannot be done, since one cannot proceed along the contours when bifurcation points are reached. Therefore, some method to account for crossed motion vectors needs to be developed. For every pixel in the first frame, once a matching pixel in the second frame is found, the displacement vector is checked with the displacement vector of the immediate predecessor of the first frame. If they do cross, then the correspondences are switched, so that the vectors no longer overlap.

A crossed motion vector means that a given pixel and its correspondence, $z_1(m) \rightarrow z_2(u(m))$, has a predecessor, $z_1(m - k)$, for some $k \geq 1$, that matches with $z_2(u(m - k))$, where $u(m) < u(m - k)$. That is, the scan-ordered predecessor, $z_1(m - k)$, corresponds to $z_2(u(m - k))$, the scan-ordered successor of $z_2(u(m + k))$. After the two crossed vectors are identified their correspondences are switched. Finding the predecessor of a given pixel $z_1(m)$ in the first frame is done by using the masking approach that is introduced in Section III.C. Finding the vectors that cross is done by determining the equations of the vectors (treated as line segments), solving these equations for its point of intersection, and determining if its point of intersection lies within the vectors. Let $(x^*, y^*)$ be the point of intersection. Then, a crossed vector is found if
\[
\min \{x_1, x_2\} < x^* < \max \{x_1, x_2\} \\
\min \{y_1, y_2\} < y^* < \max \{y_1, y_2\}.
\]

(18)

Note that, when the point of intersection lies on one of the points, then, the corresponding vectors are not treated as crossed. This is mainly due to the concept of elastic deformation, that is, one pixel may correspond to one or more pixels in the next frame, thus giving "one-to-many" and "many-to-one" correspondences.

IV. EXPERIMENTAL RESULTS

Two sets of experiments are presented, the first on the registration of skeletons produced by simulated deformations, and the second on the registration of real skeleton images. For the experiments, three $64 \times 64$ sections were extracted from each frame, depicted in the white boxes of Figs. 1, 2, and 3.

A. Synthesized Motion

In order to evaluate the algorithm's performance quantitatively, the correct displacement vector field must be known. For this reason, the algorithm is applied to frames that have been deformed in a known manner.

For uniform translation and rotation of the skeleton, the deformation can be modeled as
\[
\begin{bmatrix}
x_2(m) \\
y_2(m)
\end{bmatrix} = \begin{bmatrix} T_1 & T_2 \\
T_1 & T_2 \
\end{bmatrix} \begin{bmatrix} x_1(m) \\
y_1(m)
\end{bmatrix} + \begin{bmatrix} \cos \theta & \sin \theta \\
-\sin \theta & \cos \theta
\end{bmatrix} \begin{bmatrix} x_1(m) \\
y_1(m)
\end{bmatrix}.
\]

(19)

where $T = T_1 + jT_2$ is the translation or displacement vector, $\theta$ the angle of rotation, and $(x_1(m), y_1(m))$ are the coordinates of the $m$th pixel of the $i$th frame. Note that, in a simulation transformation the mapping $n = u(m)$ in (2) degenerates to $n = u(m)$, and the correct registration result is available for comparison purposes.

Translating the first frame of the skeleton of Fig. 4 by eight pixels to the right and two pixels down ($T = 8 + j2$), and then rotating the frame by $\theta = 6$ degrees yields the displaced vector field given by Fig. 9, with $M = N = 191$. Note that rotating the frame introduces a nonlinear deformation, as can be seen by comparing the right side of the catheter to the top left side of the "Y."

Since $u(m) = n$, the mean square error, $E_{MSE}$, is simply
\[
E_{MSE} = \frac{1}{N} \sum_{i=1}^{N} \|d(i) - \hat{d}(i)\|^2
\]

(20)

where $d(m)$ is the correct displacement vector given by
\[
d(m) = z_1(m) - z_2(m),
\]
$\hat{d}(m)$ is the vector obtained from the algorithm using (12), and $N$ is the number of correspondences found.
Applying the algorithm without using either causal masking or global modeling yields the vector field of Fig. 10. Note that, with a few exceptions, the displacement vectors are oriented at just about the same angles as that of Fig. 9. However, the errors in the registration can be seen in the crossed motion vectors and in the "many-to-one" mappings. Using (20), with \( N = 191 \), the \( E_{MSE} = 22.483 \).

Next, Fig. 11 shows the result after decrossing motion vectors. There is still some error along the catheter, however. Improved results are obtained by using causal masking, as shown in Fig. 12. This vector field seems to be more accurate, noting the improvement along the catheter between the left and right branches of the "Y." Indeed, the error measurements suggest an improvement, with a reduced \( E_{MSE} = 20.307 \).

Fig. 13 shows the result of using both causal masking as well as global matching, with \( \lambda_{pm} = 0.02 \). Here, the MSE errors decrease to \( E_{MSE} = 18.109 \). Note that introducing global matching makes the vector field more uniform, with many of the "two-to-one" correspondences resolved to the more accurate "one-to-one" mappings along the right branch of the "Y".

Thus, by comparing the registration result after each stage as shown in Figs. 10, 11, 12, and 13, to the correct vector field, Fig. 9, the improvements offered by each stage can be clearly seen, with the most improvement gained from the global matching step.

B. Real Motion

In this section, three registration results of three different image sequences will be shown. The first image sequence, Fig. 4, contained three frames, where \( M = 191, N = 182 \), and the third frame contained 175 pixels. Registering the first two frames without the use of causal masking and global matching yields Fig. 14. Note the presence of crossed vectors and several "two-to-one" mappings, which may occur in elastic deformations of complex contours. From this figure, it appears that both branches and the catheter move similarly. Using causal masking and global matching (\( \lambda_{pm} = 0.02 \)) gives Fig. 15. The variance of the error for AR modeling is 169.687, while the variance of the error due to global matching is 19544.9. The ratio of the variances is roughly 1/100, so the error due to global matching should be weighted less by about two orders of magnitude. Note that the vector field of Fig. 15 is smoother than Fig. 14.
Fig. 14. Exp. #1, frames 1–2, $\lambda_{pm} = 0.0$.

Fig. 15. Exp. #1, frames 1–2, causal masking, $\lambda_{pm} = 0.02$.

Fig. 16. Exp. #1, frames 2–3, causal masking, $\lambda_{pm} = 0.0$.

Fig. 17. Exp. #1, frames 2–3, causal masking, $\lambda_{pm} = 0.03$.

Fig. 18. Exp. #2, frames 1–2, $\lambda_{pm} = 0.00$.

Registering the next two frames with causal masking but not global matching gives Fig. 16. Note that crossed vectors, in the upper right branch and the right end of the catheter, are present. Again, the motion of all three objects seem to be very similar. Upon global matching ($\lambda_{pm} = 0.03$), the crossed vectors are removed, yielding Fig. 17.

In applying the AR-DP algorithm, it had been assumed that neighboring pixels behave (move) similarly. This assumption was necessary to predict a displacement vector from past vectors that corresponded to physically close pixels. However, this assumption does not hold when objects overlap each other, since these contours can move quite differently. Therefore, additional experiments were performed on skeletons with bifurcation points.

In Fig. 5, the two branches belonging to the oval represent a physically different object (a wire hook) from the artery, and are not physically connected. Fig. 18 shows the vector field after registering the first two frames, with no decrossing, masking, or global matching. Note that the pixels belonging to the artery near the top left are mapped into pixels that belong to the wire hook. Another section that causes errors is where the hook and the artery overlap towards the middle. Here the two different sections cannot be resolved from this vector field. In order to improve this, causal masking, decrossing, and global matching are employed, giving Fig. 19. The improvement is very noticeable. First, note that there is a better resolution between the two objects near the top left, especially right before the section where they overlap. Pixels belonging to
the artery are mapped to pixels in the next frame belonging to the same artery. Also, note the reduction in the number of crossed motion vectors, noticeable in the bottom right branch of the artery (the small "Y").

Registering the next two frames using the original AR-DP algorithm is shown in Fig. 20. Fig. 21 shows the improvement in registration by using causal masking, decrossing, and global matching ($\lambda_{pm} = 0.02$). Again, note that the two objects are better resolved, accompanied by a significant reduction in the number of crossed motion vectors. However, this comes at the cost of the number of correspondences found. In particular, not as many correspondences have been found in the upper artery as those found in the overlapping section.

Finally, Fig. 22 shows the result of registering Fig. 6. Although this image sequence is of two physically connecting arteries, there is still quite a noticeable difference in motion between the three branches. The upper two branches are moving down while the bottom branch is moving up. Because of this, there is some confusion in the area around the bifurcation point. This results in at least two erroneous mappings shown in the middle of Fig. 22, represented by the long motion vectors that map the bottom branch to the upper left branch. The next figure, Fig. 23, shows the result after using masking and global matching, with $\lambda_{pm} = 0.03$. Since global matching smooths the vector field, all three branches now are more similar to each other, and the bifurcation points are more accurately registered, with distinct branches being very visible near this area.
V. CONCLUSIONS

In this paper, the estimation of motion in angiographic images was performed by registering corresponding multi-branched skeletonized frames. Previous approaches to registration were limited to ordered sequences of pixels, or equivalently, simple skeleton images with a single body. This elastic registration method used dynamic programming (DP) in conjunction with an AR model of the estimated displacement vectors. To be able to register multibranched skeletons, several extensions have been proposed. First, causal masking was used to ensure that the pixels belonging to the same rigid body were used in predicting the displacement vectors. In addition, a global matching model was proposed for improving the registration by smoothing the entire vector field, and the error function was modified to account for contributions on a local as well as a global level. This significantly improved the registration when the deformation was fairly linear. Also, the theory of regularization was used to define the relative weightings of the two error terms.

In this paper, it has been assumed that neighboring pixels behave similarly. However, this assumption breaks down in cases where two or more contours are not connected physically, where one overlaps the other. Rather than investigate how to distinguish the two cases, we have not changed anything in our algorithm, leaving causal masking and global matching to handle both cases. Experimental results have shown that these techniques do an adequate job of registration, especially around bifurcation points.

REFERENCES