35th Annual International Conference of the IEEE EMBS
Osaka, Japan, 3 - 7 July, 2013

Performance Evaluation of Point Matching Algorithms for Left Ventricle Motion Analysis in MRI

Carlos Santiago[1], Jacinto C. Nascimento[1] and Jorge S. Marques[1]

Abstract— **Finding correspondences between contour points in consecutive frames is crucial for the left ventricular motion analysis. In many medical applications, point correspondences can be determined by using distinctive anatomical features, called anatomical landmarks. However, in the case of cardiac images, these landmarks are scarce and insufficient for the registration. Several methods have been proposed using semi-landmarks, but this may lead to incorrect correspondences. This paper proposes and evaluates the performance of three point matching algorithm. Results show that the matching by resampling method leads to the best overall correspondences and compares favorably with the performance of a state of the art shape alignment algorithm [9].**

I. INTRODUCTION

Point matching is a well-studied topic in computer vision that aims to determine correspondences between specific points in different images (or time frames). This task is often required as a preliminary step in many applications, including tracking, object recognition and image segmentation.

In most medical image problems, point correspondences are necessary to perform anatomical registration. They can be determined by using distinctive anatomical features, called landmarks. In some cases, anatomical landmarks are easily recognizable (either across patients or across time) and determining the correspondences between them becomes a relatively simple task. However, in the case of cardiac images anatomical landmarks are scarce and insufficient for the registration.

Even though tagged magnetic resonance (MR) images can be used to obtain anatomical markers from the myocardium, the clinical protocol only acquires these type of images when local function needs to be assessed. This means that the number of images available is significantly shorter than normal MR images, particularly in healthy patients. A common approach in the literature [1] to perform anatomical registration is to consider a set of semi-landmarks, which are spatial points with no real (material) meaning. These semi-landmarks usually describe a curve/shape that can be used to establish point correspondences and image registration is performed through shape alignment methods. However, there is a major drawback in this approach since the trajectories obtained from the semi-landmarks often do not describe the actual cardiac motion.

This paper evaluates the accuracy of several point matching algorithms. Knowing which approach better describes the trajectory of each (material) point of the endocardium is essential to study the left ventricular motion. Four different strategies are used to determine point correspondences between the left ventricle (LV) in consecutive frames, in MR images. A thorough evaluation of each method will give insights into the best approach.

The paper is organized as follows. Section II gives an overview of the state of the art methodologies. Section III describes the problem and Sections IV-V describe the point matching algorithms. Section VI shows the evaluation results and, finally, Section VII concludes the paper.

II. RELATED WORK

Image registration is commonly used in LV segmentation/tracking problems to determine statistical models of shape variability (known as active shape models (ASM) [5]). The registration step is usually performed by applying a shape alignment algorithm. Two consecutive contours can be aligned by applying a transformation $T(\cdot)$ to one of them. The transformation parameters and point correspondences are usually determined by combining both into an energy minimization framework [6]. The Iterative Closest Point (ICP) algorithm, proposed by Besl and McKay [7], is one of the simplest approaches to determine an affine transformation that aligns the shapes. It computes the transformation by alternately updating the transformation parameters and the point correspondences until convergence. O'Brien et al. [8] aligns the shapes into a common coordinate system by simply removing translation between them. The authors state that the scaling factor is captured by the ASM and that rotational effects are not significant. Point matches are then superimposed by re-sampling the shapes into a fixed number of points. Alternatively, Tsai et al. [9] proposed an algorithm that does not require point correspondences to align shapes. Instead, they compute the affine transformation parameters by minimizing an energy function based on binary images delimited by the shapes.

III. PROBLEM STATEMENT

In this work, the LV endocardium contours are represented by sets of points outlined in 2D+time MR images. In order to obtain the trajectories of each point in time, the point correspondences between consecutive frames need to be known. The problem can be formally introduced as follows. Let $\mathcal{C}(t) \in \mathbb{R}^{2\times N}$ and $\mathcal{C}(t+1) \in \mathbb{R}^{2\times M}$ be two sets of N points and M points, respectively, corresponding to two consecutive contours. A point matching algorithm allows us

This work was supported by the FCT projects PEst-OE/EEI/LA0009/2011, HEARTTRACK PTDC/EEA-CRO/103462/2008 and SFRH/BD/87347/2012.

[1]C. Santiago, J.C. Nascimento and J.S. Marques are with Institute for Systems and Robotics, Instituto Superior Técnico, Portugal

978-1-4577-0216-7/13/$26.00 ©2013 IEEE

to know the position of a point in $\mathcal{C}(t)$ in the next frame by matching it to a point in $\mathcal{C}(t+1)$.

Point matches will be obtained in a pair-wise way, i.e., by taking into account pairs of consecutive contours (e.g., $\mathcal{C}(t)$ and $\mathcal{C}(t+1)$). We assume that each contour can be densely-sampled (e.g., by cubic-spline interpolation), so that they can be considered continuous curves. Dense contours will be denoted by $\tilde{\mathcal{C}}$. Three different point matching algorithms are proposed to determine the correspondences between points in consecutive frames. These are described in the following subsections. A fourth state of the art algorithm [9] will also be used to compare with the proposed ones.

IV. PROPOSED ALGORITHMS

A. Matching by resampling

The first approach consists of determining correspondences by simply applying a sampling algorithm that extracts a set of points $\mathcal{C} \in \mathbb{R}^{2\times N}$ from each contour at pre-defined angle intervals $\Delta\theta$ relative to the contour mass center o (see Fig. 1). The angle interval, $\Delta\theta$, relates to the number of sampling points N by

$$\Delta\theta = \theta_i - \theta_{i-1} = \frac{2\pi}{N} \tag{1}$$

and each sampled point c_i is obtained by intersecting the contour in the following way

$$c_i = o + \alpha_i \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \end{bmatrix} \tag{2}$$

where $\alpha_i \in \mathbb{R}^+ : c_i \in \tilde{\mathcal{C}}$. The initial angle was set to be $\theta_1 = 0$.

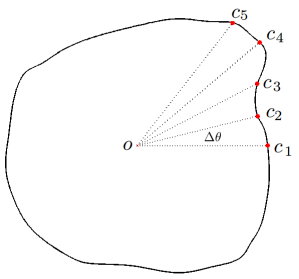

Fig. 1. Resampling by angle $\Delta\theta$ to the center o.

The same procedure is performed for two consecutive contours $\mathcal{C}(t)$ and $\mathcal{C}(t+1)$. Point correspondences are then straightforward by matching the points $c_i(t)$ to $c_i(t+1)$, $i = 1, \ldots, N$.

B. Matching by motion correspondences

The second method is inspired by the point matching algorithm described by Veenman et al. [10], which uses a first order motion model to track multiple (independent) moving points. In this work, we use a modified version of this algorithm that works as follows. Consider a set of moving points $\mathcal{C}(t) \in \mathbb{R}^{2\times N}$ that describes the LV contour at time frame t. Then consider a set of observation points from a densely-sampled contour $\tilde{\mathcal{C}}(t+1) \in \mathbb{R}^{2\times M}$ $(M \gg N)$ at $t+1$. Let $D_{N\times M}$ be a cost matrix where each entry d_{ij} is the cost of matching the point $c_i(t)$ to observation $c_j(t+1)$

$$d_{ij} = ||c_i(t) + v_i(t) - c_j(t+1)||, \tag{3}$$

where $v_i(t) = c_i(t) - c_i(t-1)$ is the first order motion model, i.e. the velocity of each point. The best global matches are determined using the Hungarian algorithm [11], which minimizes the total cost (sum of all matching costs) under the assumption that assignments must be a one-to-one mapping, i.e., each point has one and only one correspondence. Fig. 2 shows an example of the trajectories obtained with this matching algorithm.

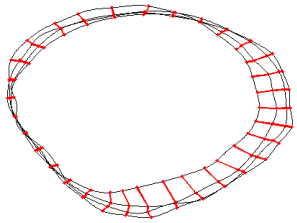

Fig. 2. Point matching using a modified version of [10]. The black lines represent the endocardium contours in 4 consecutive frames and the red lines and dotes are the trajectories obtained from the matched points.

C. Matching by Nearest-Neighbor with correction

The third method begins by assigning correspondences using the Nearest-Neighbor criterion (NNC), based on the Euclidean distance between points. The NNC consists in matching a point $c_i(t)$ of a contour $\mathcal{C}(t) \in \mathbb{R}^{2\times N}$ to the closest point, $c_j(t)$, of another contour, $\tilde{\mathcal{C}}(t+1) \in \mathbb{R}^{2\times M}$ $(M \gg N)$, such that

$$||c_i(t) - c_j(t+1)|| < ||c_i(t) - c_k(t+1)|| \ , \ \forall\ k \neq j. \tag{4}$$

This leads to a new set, $\mathcal{C}(t+1)$, of N points from $\tilde{\mathcal{C}}(t+1)$ that are matched to N points in $\mathcal{C}(t)$. However, successively assigning the closest point may cause two points that were initially apart to converge to the same trajectory. This is not desirable, since it cannot happen with material points from the endocardium. Therefore, after matching the points, a correction algorithm is applied to obtain a better point distribution in $\mathcal{C}(t+1)$.

The correction algorithm works as follows. Let us denote $c_{c_i(t)}$ as a point from $\mathcal{C}(t+1)$ matched with $c_i(t)$ and let L be the perimeter of $\mathcal{C}(t+1)$. Points $c_{c_i(t)}$ that need correction are the ones that do not preserve one of the following spatial distribution criteria

$$||c_{c_i(t)} - c_{c_{i-1}(t)}|| \geq d_1 \ \wedge \ ||c_{c_i(t)} - c_{c_{i+1}(t)}|| \leq d_2 \tag{5}$$

$$||c_{c_i(t)} - c_{c_{i-1}(t)}|| \leq d_2 \ \wedge \ ||c_{c_i(t)} - c_{c_{i+1}(t)}|| \geq d_1 \tag{6}$$

where

$$d_1 = \beta\frac{L}{N} \qquad d_2 = (2-\beta)\frac{L}{N} \tag{7}$$

and $\beta \in \ ]0,1[$. d_1 and d_2 determine the minimum and maximum distance allowed between consecutive points in

$\mathcal{C}(t+1)$, respectively. When (5) (or (6)) do not hold, it means that $c_{c_i(t)}$ is too close to the previous (next) point or too far from the next (previous) one. The correction algorithm needs to correct these points. The correction is done by iteratively sliding them along $\tilde{\mathcal{C}}(t+1)$ accordingly, until all points meet the criteria. Fig. 3 shows an example of the correction algorithm course of action.

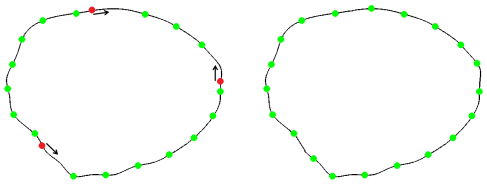

Fig. 3. Correction of points matched by the NNC. The red dots are matched points that do not preserve the spacial distribution criteria and the arrows show the direction where the correction algorithm moves them along $\tilde{\mathcal{C}}(t+1)$.

The parameter β determines how flexible the correction algorithm is. As $\beta \to 1$, the spatial distribution criteria becomes more rigid and imposes that the points are equally spaced by arc-length ($d_1 = d_2 = L/N$), whereas $\beta \to 0$ means that the criteria are more flexible and that the matched points may be unevenly distributed along $\mathcal{C}(t+1)$. The value that showed the best performance was $\beta = 0.9$.

V. SHAPE ALIGNMENT ALGORITHM

This section describes the fourth algorithm that uses a state of the art shape alignment method [9], which will also compared with the algorithms described Section IV.

D. Matching by binary alignment

This algorithm consists of aligning the two binary shapes using the shape alignment algorithm described in [9] and then matching points based on the NNC. The shape alignment algorithm uses binary images delimited by each shape to determine the transformation parameters that align them. This means that there is no need for explicit point correspondences prior to alignment.

Let $I_{\mathcal{C}(t)}(x,y)$ be the binary image associated to a contour $\mathcal{C}(t)$ and let $\tilde{I}_{\mathcal{C}(t+1)}(\tilde{x},\tilde{y})$ be the transformed binary image associated to a contour $\mathcal{C}(t+1)$, where

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ 1 \end{bmatrix} = T \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{8}$$

and T is the transformation matrix. The algorithm consists in minimizing the energy function

$$E = \frac{\iint_\Omega (I_{\mathcal{C}(t)} - \tilde{I}_{\mathcal{C}(t+1)})^2 dA}{\iint_\Omega (I_{\mathcal{C}(t)} + \tilde{I}_{\mathcal{C}(t+1)})^2 dA} \tag{9}$$

using the gradient descent method (see [9] for details). Fig. 4 shows an example of the alignment of two binary images.

After obtaining the transformation parameters that aligns the two images, we apply the transformation T to the contour $\mathcal{C}(t+1)$. Then, the points from $\mathcal{C}(t)$ are matched to points from $\mathcal{C}(t+1)$ using the NNC. Finally, the matched points from $\mathcal{C}(t+1)$ are transformed back to the original coordinate system (see Fig. 4).

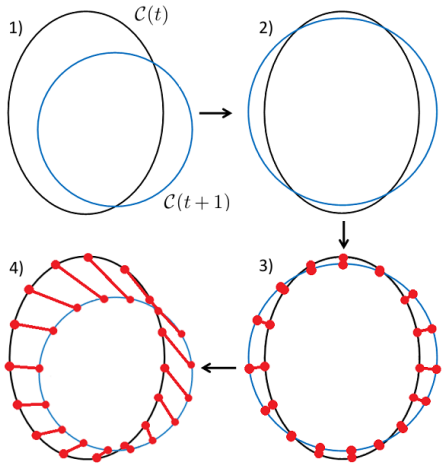

Fig. 4. Point matching using the shape alignment algorithm described in [9]. 1) the initial contours; 2) contours after the shape alignment; 3) point matching using the NNC; and 4) matched points transformed back to the original coordinates.

VI. EVALUATION

The evaluation of the point matching algorithms is performed using 4 2D+time MR sequences from healthy patients, each with 20 frames. These sequences belong to a database of 3D+time MR images publicly available in [12]. This database also includes manual annotations of the endocardium, which include the papillary and trabeculae muscles (PTM) as part of the blood pool (i.e., within the endocardium).

The algorithms are evaluated by comparing the trajectories from the point matches with registration points from the PTM, which are scarce but reliable material landmarks [2], [3], [4]. The PTM were obtained by manually indicating their location in each frame and were then projected to the corresponding contour (see Fig. 5). A total of 6 PTM points were used to evaluate the algorithms.

Quantification of the error in each frame is obtained as follows. Let $c_k(1)$ and $c_{k+1}(1)$ be the points from $\mathcal{C}(1)$ that are closest to the projected PTM point $p(1)$ in the first frame. The point, $c_p(t)$, that is used to evaluate the matching algorithms is obtained by linear interpolation

$$c_p(t) = \eta c_k(t) + (1-\eta) c_{k+1}(t), \tag{10}$$

where η is a constant defined by

$$\eta = \frac{||c_{k+1}(1) - p(1)||}{||c_k(1) - c_{k+1}(1)||}. \tag{11}$$

The error is obtained by comparing the trajectories of the $c_p(t)$ (according to the matching algorithm) and the trajectories of the $p(t)$ (considered as ground truth). The error metric $e(t)$ used is based on the Euclidean distance between these two points throughout time

$$e(t) = \frac{||c_p(t) - p(t)||}{L(t)} \tag{12}$$

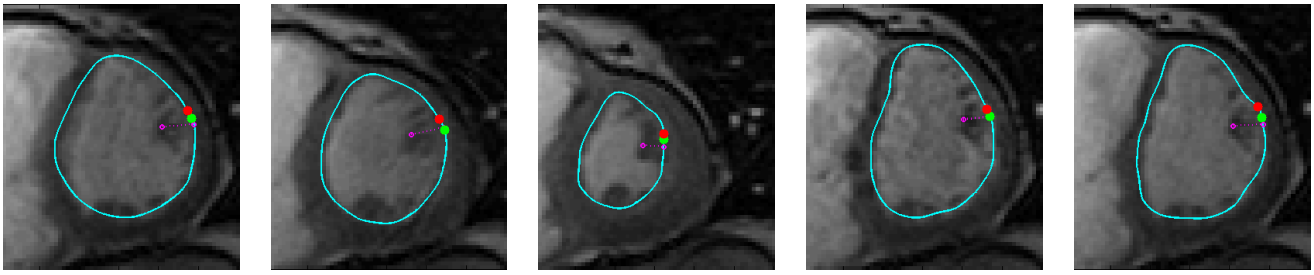

Fig. 5. Example of matches obtained using method A (green) and B (red) at frames 1, 4, 8, 15 and 20. The purple marker shows the tracked PTM point.

where $L(t)$ is a normalization term that corresponds to the perimeter of the contour at frame t. The normalization is needed because the size of the LV changes significantly throughout time and Euclidean distances would be smaller during the systolic phases. Note that from (10), $c_p(1) = p(1)$, which means that $e(1) = 0$ for all matching algorithms.

Fig. 6 shows the evolution of the average error metric (12) over time for each of the four methods. During the first six frames, all the methods show a similar increase in the error. These frames correspond to the systolic phase, in which the LV begins to contract significantly. The following 5 to 10 frames correspond to the relaxation of the LV (diastole). In these frames, the performance of the method is significantly different. Methods A, C and D remain with error values in the range 0.010-0.025, while method B increases towards values of 0.03-0.045. This can be explained by the following. While method A ignores the natural rotation of the LV by imposing fixed angles, this property also guarantees that the points being track do not accumulate error throughout time. On the opposite side, in method B correspondences are computed successively as if they are independent moving points, which allows them to continuously grow farther apart from its original location. And although the matches obtained with methods C and D can also accumulate error, they show a more stable behavior.

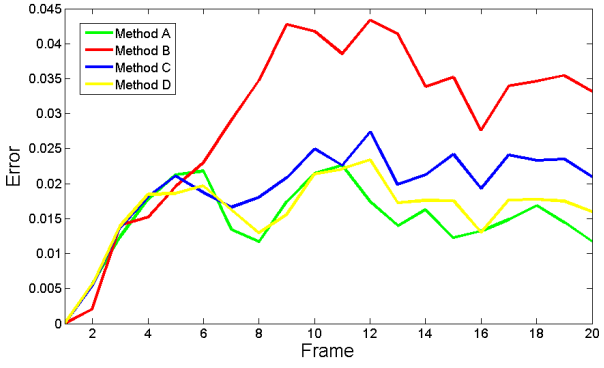

Fig. 6. Average error over time for each matching algorithm.

From the analysis of Fig. 6, methods A and D show a similar performance and achieve better results than the other methods. Table I also confirms this by showing that the best average error over all sequences and frames are achieved by methods A and D, followed by methods C and B.

Fig. 5 illustrates an example comparing the trajectories obtained with methods A (best) and B (worst).

TABLE I

AVERAGE ERROR ACROSS ALL SEQUENCES AND FRAMES.

Method	**A**	**B**	**C**	**D**
Error	0.015	0.030	0.019	0.016

VII. CONCLUSIONS

This paper evaluates the performance of 4 different point matching algorithms. The results show that the proposed method: *A. Matching by resampling* outperforms the remaining methods, with an average error of 0.015, and achieves similar results to the state of the art method proposed in [9]. In future work, this performance evaluation should be extended to further data and the results should be compared to the information obtained from tagged-MRI.

REFERENCES

[1] T. Heimann and H. Meinzer, "Statistical shape models for 3D medical image segmentation: a review," Medical image analysis, vol. 13, no. 4, p. 543, 2009.

[2] T. Baks, et al., "Recovery of left ventricular function after primary angioplasty for acute myocardial infarction," European heart journal, vol. 26, no. 11, pp. 1070–1077, 2005.

[3] J. Ortiz-Pérez, J. Rodríguez, S. Meyers, D. Lee, C. Davidson, and E. Wu, "Correspondence between the 17-segment model and coronary arterial anatomy using contrast-enhanced cardiac magnetic resonance imaging," JACC: Cardiovascular Imaging, vol. 1, no. 3, pp. 282–293, 2008.

[4] Y. Notomi, et al., "Measurement of ventricular torsion by two-dimensional ultrasound speckle tracking imaging," Journal of the American College of Cardiology, vol. 45, no. 12, pp. 2034–2041, 2005.

[5] T. Cootes, et al., "Active shape models-their training and application," Computer vision and image understanding, vol. 61, no. 1, pp. 38–59, 1995.

[6] C. Petitjean and J. Dacher, "A review of segmentation methods in short axis cardiac MR images," Medical image analysis, vol. 15, no. 2, pp. 169–184, 2011.

[7] P. Besl and N. McKay, "A method for registration of 3-D shapes," IEEE Transactions on pattern analysis and machine intelligence, vol. 14, no. 2, pp. 239–256, 1992.

[8] S. O'Brien, O. Ghita, and P. Whelan, "A novel model-based 3D+time left ventricular segmentation technique," IEEE Transactions on Medical Imaging, vol. 30, no. 2, pp. 461–474, 2011.

[9] A. Tsai, et al., "A shape-based approach to the segmentation of medical imagery using level sets," IEEE Transactions on Medical Imaging, vol. 22, no. 2, pp. 137–154, 2003.

[10] C. Veenman, M. Reinders, and E. Backer, "Resolving motion correspondence for densely moving points," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 1, pp. 54–72, 2001.

[11] H. Kuhn, "The Hungarian method for the assignment problem," Naval research logistics quarterly, vol. 2, no. 1-2, pp. 83–97, 2006.

[12] A. Andreopoulos and J. Tsotsos, "Efficient and generalizable statistical models of shape and appearance for analysis of cardiac MRI," Medical Image Analysis, vol. 12, no. 3, pp. 335–357, 2008.