Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Image and Vision Computing 22 (2004) 461–470

www.elsevier.com/locate/imavis

Deformable spatio-temporal shape models: extending active shape models to 2D + time

Ghassan Hamarneh[a,*], Tomas Gustavsson[b,1]

[a]School of Computing Science, Simon Fraser University, Burnaby, BC, Canada V5A 1S6
[b]Department of Signals and Systems, Chalmers University of Technology, Göteborg SE-412 96, Sweden

Received 31 July 2002; received in revised form 24 November 2003; accepted 25 November 2003

Abstract

This paper extends 2D active shape models to 2D + time by presenting a method for modeling and segmenting spatio-temporal shapes (ST-shapes). The modeling part consists of constructing a statistical model of ST-shape parameters. This model describes the principal modes of variation of the ST-shape in addition to constraints on the allowed variations. An active approach is used in segmentation where an initial ST-shape is deformed to better fit the data and the optimal proposed deformation is calculated using dynamic programming. Segmentation results on both synthetic and real data are presented.

Keywords: Deformable models; Segmentation; Active shape models; Spatio-temporal shapes; Dynamic programming; Echocardiography

1. Introduction

Much work has been done on tracking rigid objects in 2D sequences. In many image analysis applications, however, there is a need for modeling and locating non-rigid time-varying object shapes. One approach for dealing with such objects is the use of deformable models. Deformable models [21] such as snakes [12] and its variants [4,8,9,15,17], have attracted considerable attention and are widely used for segmenting non-rigid objects in 2D and 3D (volume) images. However, there are several well-known problems associated with snakes. They were designed as interactive models and therefore rely upon a user to overcome initialization sensitivity. They were also designed as general models showing no preference for a particular object shape other than those that are smooth. This generality can cause unacceptable results when snakes are used to segment objects with shape abnormalities arising from occlusion, closely located but irrelevant structures, or noise. Thus, techniques which incorporate a priori knowledge of object shape were introduced [6,22]. In active shape models (ASM) [6], the statistical variation of shapes is modeled beforehand in accordance with a training set of known examples. In order to attack the problem of tracking non-rigid time-varying objects, deformable models were extended to dynamic deformable models [14,16,18–20,23]. These describe the shape changes (over time) in a single model that evolves through time to reach a state of equilibrium where internal forces, representing constraints on shape smoothness, balance the external image forces and the contour comes to rest. Deformable models have been constructed by applying a probabilistic framework and led to techniques such as 'Kalman snakes' [24]. Motion tracking using deformable models has been used for tracking non-rigid structures such as blood cells [14] and much attention has been given to the human heart and the tracking of the left ventricle in both 2D and 3D [13,16,18,20]. In addition to tracking rigid objects, previous work focused on arbitrary non-rigid motion and gave little attention to tracking objects moving in specific motion patterns, without the incorporation of statistical prior knowledge in both 2D and time [2].

In this paper, we present a new method for locating spatio-temporal shapes (ST-shapes) in image sequences.

* Corresponding author. Tel.: +1-604-291-3007; fax: +1-604-291-3045.
E-mail addresses: hamarneh@cs.sfu.ca (G. Hamarneh); gustavsson@s2.chalmers.se (T. Gustavsson).
[1] Tel.: +46-31-772-1802; fax: +46-31-772-1782.

0262-8856/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2003.11.009

We extend ASM [6] to include knowledge of temporal shape variations and present a new ST-shape modeling and segmentation technique. The method is well suited to model and segment objects with specific motion patterns, as in cardiography, optical signature motion recognition, and lip-reading for human computer interaction.

2. Method

In order to model a certain class of ST-shapes, a representative training set of known shapes is collected. The set should be large enough to include most of the shape variations we need to model. Next, all the ST-shapes in the training set are parameterized. A data dimensionality reduction stage is then performed by capturing only the main modes of ST-shape variations. In addition to constructing the ST-shape model, the training stage also includes the modeling of gray-level information. The task is then to locate an ST-shape given a new unknown image sequence. An average ST-shape is first initialized, 'optimal' deformations are then proposed, and then the deformations are constrained to agree with the training data. The proposed changes minimize a cost function that takes into account both the temporal shape smoothness constraints and the gray-level appearance constraints. The search for the optimum proposed change is done using dynamic programming. The following sections present the various steps involved in detail.

2.1. Statistical ST-shape variation

The training set. We collect N training frame-sequences each with F frames. The training set, $\Phi_V = [V_1, V_2, \ldots, V_N]$, displays similar objects and similar object motion patterns. $\Phi_V(i) = V_i = [f_{i1}, f_{i2}, \ldots, f_{iF}]$ is the ith frame-sequence containing F frames and $V_i(j) \equiv \Phi_V(i,j) = f_{ij}$ is the jth frame of the ith frame-sequence containing the intensity value $f_{ij}(r,c) \equiv \Phi_V(i,j,r,c)$ at the rth row and cth column of the frame.

The ST-shape parameters. We introduce S_i to denote the parameter vector representing the ith ST-shape. Parameterization is done using landmarks (other shape parameterization methods may be utilized, e.g. Fourier descriptors [3] or B-splines [23]). Landmarks are labeled either manually, as when a cardiologist labels the heart chamber boundaries [6,11], or (semi-)automatically [10]. Each landmark point is represented by its (x,y) coordinate. Using L landmarks per frame and F frames per sequence, we can write the training set of ST-shapes as $\Phi_S = [S_1, S_2, \ldots, S_N]$, where $\Phi_S(i) = S_i = [r_{i1}, r_{i2}, \ldots, r_{iF}]$ is the ith ST-shape containing F shapes and $S_i(j) \equiv \Phi_S(i,j) = r_{ij}$ is the jth shape of the ith ST-shape. r_{ij} can be written as $r_{ij} = [x_{ij1}, y_{ij1}, x_{ij2}, y_{ij2}, \ldots, x_{ijL}, y_{ijL}]$ where $x_{ijk} = r_{ij}(k,1) \equiv \Phi_S(i,j,k,1)$ and $y_{ijk} = r_{ij}(k,2) \equiv \Phi_S(i,j,k,2)$ are the (x,y) coordinates of the kth landmark of the shape r_{ij}.

ST-shapes alignment. Next, the ST-shapes are aligned in order to allow comparing equivalent points from different ST-shapes. This is done by rotating, scaling and translating the shape in each frame of the ST-shape by an amount that is fixed within one ST-shape. A weighted least-squares approach is used for aligning two sequences and an iterative algorithm is used to align all the ST-shapes. Given two ST-shapes,

$$S_1 = [x_{111}, y_{111}, \ldots, x_{11L}, y_{11L}, x_{121}, y_{121}, \ldots, x_{12L}, y_{12L}, \ldots, x_{1F1}, y_{1F1}, \ldots, x_{1FL}, y_{1FL}]$$

and

$$S_2 = [x_{211}, y_{211}, \ldots, x_{21L}, y_{21L}, x_{221}, y_{221}, \ldots, x_{22L}, y_{22L}, \ldots, x_{2F1}, y_{2F1}, \ldots, x_{2FL}, y_{2FL}],$$

we need to find the rotation angle θ, the scaling factor s, and the value of the translation (t_x, t_y) that will align S_2 to S_1. To align S_2 to S_1, S_2 is mapped to

$$\hat{S}_2 = [\hat{x}_{211}, \hat{y}_{211}, \ldots, \hat{x}_{21L}, \hat{y}_{21L}, \hat{x}_{221}, \hat{y}_{221}, \ldots, \hat{x}_{22L}, \hat{y}_{22L}, \ldots, \hat{x}_{2F1}, \hat{y}_{2F1}, \ldots, \hat{x}_{2FL}, \hat{y}_{2FL}]$$

using $\hat{S}_2 = M(s,\theta)[S_2] + \mathbf{t}$, where $M(s,\theta)[S_2]$ is a rotated then scaled version of each coordinate of S_2 (by θ and s, respectively) and

$$\mathbf{t} = [t_x, t_y, t_x, t_y, \cdots, t_x, t_y]^{\mathrm{T}}$$

is a translation vector of length $2FL$. The weighted distance between S_1 and $\hat{S}_2$ in the $2FL$ dimensional space is given by

$$d^2_{1\hat{2}} = (\hat{S}_2 - S_1)^{\mathrm{T}} \mathbf{W}^{\mathrm{T}} \mathbf{W} (\hat{S}_2 - S_1),$$

where

$$\mathbf{W} = \mathrm{diag}(w_{11x}, w_{11y}, \ldots, w_{1Lx}, w_{1Ly}, w_{21x}, w_{21y}, \ldots, w_{2Lx}, w_{2Ly}, \ldots, w_{F1x}, w_{F1y}, \ldots, w_{FLx}, w_{FLy}).$$

The elements of $\mathbf{W}$ reflect our trust in each coordinate and are chosen to be proportional to the 'stability' of the different landmarks over the training set [6].

To rotate, scale and translate a single coordinate (x_{2kl}, y_{2kl}) we use

$$\begin{bmatrix} \hat{x}_{2kl} \\ \hat{y}_{2kl} \end{bmatrix} = \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \cdot \begin{bmatrix} x_{2kl} \\ y_{2kl} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix},$$

where $a_x = s\cos(\theta)$ and $a_y = s\sin(\theta)$. $\hat{S}_2$ can now be rewritten as $\hat{S}_2 = \mathbf{Az}$, where $\mathbf{z} = [a_x, a_y, t_x, t_y]^{\mathrm{T}}$ and

$$\mathbf{A}^{\mathrm{T}} = \begin{bmatrix} x_{211} & y_{211} & \cdots & x_{21L} & y_{21L} & x_{221} & y_{221} & \cdots & x_{22L} & y_{22L} & \cdots & x_{2F1} & y_{2F1} & \cdots & x_{2FL} & y_{2FL} \\ -y_{211} & x_{211} & \cdots & -y_{21L} & x_{21L} & -y_{221} & x_{221} & \cdots & -y_{22L} & x_{22L} & \cdots & -y_{2F1} & x_{2F1} & \cdots & -y_{2FL} & x_{2FL} \\ 1 & 0 & \cdots & 1 & 0 & 1 & 0 & \cdots & 1 & 0 & \cdots & 1 & 0 & \cdots & 1 & 0 \\ 0 & 1 & \cdots & 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \cdots & 0 & 1 & \cdots & 0 & 1 \end{bmatrix}.$$

The distance $d^2_{1\hat{2}}$ can now be rewritten as $d^2_{1\hat{2}} = (\mathbf{Az} - S_1)^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{W}(\mathbf{Az} - S_1)$ and we can solve for $\mathbf{z}$ (least-squares solution) that minimizes $d^2_{1\hat{2}}$ according to $\mathbf{z} = ((\mathbf{WA})^{\mathrm{T}}(\mathbf{WA}))^{-1}(\mathbf{WA})^{\mathrm{T}}\mathbf{W}S_1) = (\mathbf{A}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WA})^{-1}\mathbf{A}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{W}S_1$. Once $\mathbf{z} = [a_x, a_y, t_x, t_y]^{\mathrm{T}}$ is calculated, s and θ can be found using $\theta = \arctan(a_y/a_x)$ and $s = a_x/\cos(\arctan(a_y/a_x))$. We note that when the observed motion patterns in the training sequences span different time intervals, temporal re-sampling or aligning that incorporates temporal scaling is performed.

Main ST-shape variation modes. The N aligned ST-shapes, each of length $2FL$ and represented by $\{S_1, S_2, \ldots, S_N\}$, map to a 'cloud' of N points in a $2FL$ dimensional space. It is assumed that these N points are contained within a hyper ellipsoid of this $2FL$ dimensional space. We call this region the allowable ST-shape domain (ASTSD). We then apply principal component analysis (PCA) to the aligned training set of ST-shapes in order to find the main modes of ST-shape variation. The resulting principal components (PCs) are the eigenvectors $\mathbf{p}_k$ $(1 \le k \le 2FL)$ of the covariance matrix of the observations, C_S, found from $C_S\mathbf{p}_k = \lambda_k\mathbf{p}_k$. λ_k is the kth eigenvalue of C_S $(\lambda_k \ge \lambda_{k+1})$ and is equal to the variance along the kth PC. The mean ST-shape is calculated as

$$\bar{S} = \frac{1}{N}\sum_{i=1}^{N} S_i.$$

The PCs are normalized to unit length and are mutually orthogonal.

Model representation. Now, we express each ST-shape, S_i, as the sum of the mean ST-shape, $\bar{S}$, and a linear combination of the principal modes of variation, $\mathbf{Pb}_i$. This gives $S_i = \bar{S} + \mathbf{Pb}_i$ where $\mathbf{b}_i = [b_{i,1}, b_{i,2}, \cdots, b_{i,2FL}]^{\mathrm{T}}$ and $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_{2FL}]$. We constrain b_l to $b_{l\,\min} \le b_l \le b_{l\,\max}$ with $b_{l\,\min} = -b_{l\,\max}$ and $1 \le l \le 2FL$. $b_{l\,\max}$ is chosen to be proportional to $\sqrt{\lambda_l}$ $(-3\sqrt{\lambda_l} \le b_l \le 3\sqrt{\lambda_l}$ is typically used). In practice only the first t (out of $2FL$) PCs explaining a sufficiently high percentage of the total variance of the original data are used and the fundamental equation becomes

$$S = \bar{S} + \mathbf{P}_t\mathbf{b} \tag{1}$$

where $\mathbf{b} = [b_1, b_2, \cdots, b_t]^{\mathrm{T}}$, $\mathbf{P}_t = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_t]$, and the constraints on $\mathbf{b}$ become $b_{l\,\min} \le b_l \le b_{l\,\max}$, where $1 \le l \le t$.

2.2. Gray-level training

The information contained in the ST-shape model alone is typically not enough for ST segmentation. Therefore, additional representative information about the intensities or gray-levels relating to the object is also desired and collected in the gray-level training stage. In the search stage, new estimates of the ST-shape are sought that will better match the gray-level prior knowledge. Different gray-level representative information can be used, e.g. gathering the intensity values in the entire patch contained within the object [5] or parameterizing the profiles or patches around the landmark. In this implementation, we follow [6] and use a mean normalized derivative (difference) profile, passing through each landmark and perpendicular to the boundary created by the neighboring ones. For the kth landmark this profile is given by

$$\bar{\mathbf{y}}_k = \frac{1}{FN}\sum_{j=1}^{F}\sum_{i=1}^{N}\mathbf{y}_{ijk} \tag{2}$$

where $\mathbf{y}_{ijk}$ is the representative profile for the kth landmark in the jth shape of the ith ST-shape. Using gray-level information, temporal and shape constraints, the model is guided to a better estimate of the dynamic object hidden in the new frame-sequence.

2.3. ST-shape segmentation algorithm

Given a new frame-sequence, the task is to locate the object in all the frames or equivalently locate the ST-shape. An initial estimate of the ST-shape parameters is chosen at first, then changes to the parameters are proposed. The pose of the current estimate is then changed and suitable weights for the modes of variation are chosen in order to fit the model to the proposed changes. This is done with the restriction that the changes can only be made in accordance with the model (with reduced dimensionality) and the training set. New changes are then proposed and so on. Here, we present a detailed discussion of these steps.

Initial estimate. The search starts by guessing an initial ST-shape:

$$\hat{S}^{\langle 0\rangle} = M(s^{\langle 0\rangle}, \theta^{\langle 0\rangle})[\bar{S} + \mathbf{P}_t\mathbf{b}^{\langle 0\rangle}] + \mathbf{t}^{\langle 0\rangle} \tag{3}$$

where $\mathbf{t} = [t_x, t_y, t_x, t_y, \cdots, t_x, t_y]$ is of length $2FL$. $M(s, \theta)[S] + \mathbf{t}$ scales, rotates, and translates S by s, θ, and

$\mathbf{t}$, respectively. Both $\bar{S}$ and $\mathbf{P}_t$ are obtained from the training stage. A typical initialization would set $\mathbf{b}^{\langle 0\rangle}$ to zero and set $s^{\langle 0\rangle}$, $\theta^{\langle 0\rangle}$, and $\mathbf{t}^{\langle 0\rangle}$ to values that place the initial sequence in the vicinity of the target.

Proposing a new sequence. For each landmark, say the kth landmark in the jth frame, we define a search profile $\mathbf{h}_{jk} = [h_{jk1}, h_{jk2}, \cdots, h_{jkH}]$ that is differentiated and normalized as done with the training profiles. This gives H^F possibilities for the proposed positions of the kth landmarks in the F frames, see Fig. 1.

Since locating the new positions (one out of H^F possible) is computationally demanding, we formulate the problem as a multi-stage decision process and use dynamic programming [1] to find the optimum proposed landmark positions by minimizing a cost function. The cost function comprises two terms: one due to large temporal landmark position changes, and another reflecting the mismatch between the gray-level values surrounding the current landmarks and those expected values found in the gray-level training stage. In the following paragraphs, we detail our implementation of dynamic programming.

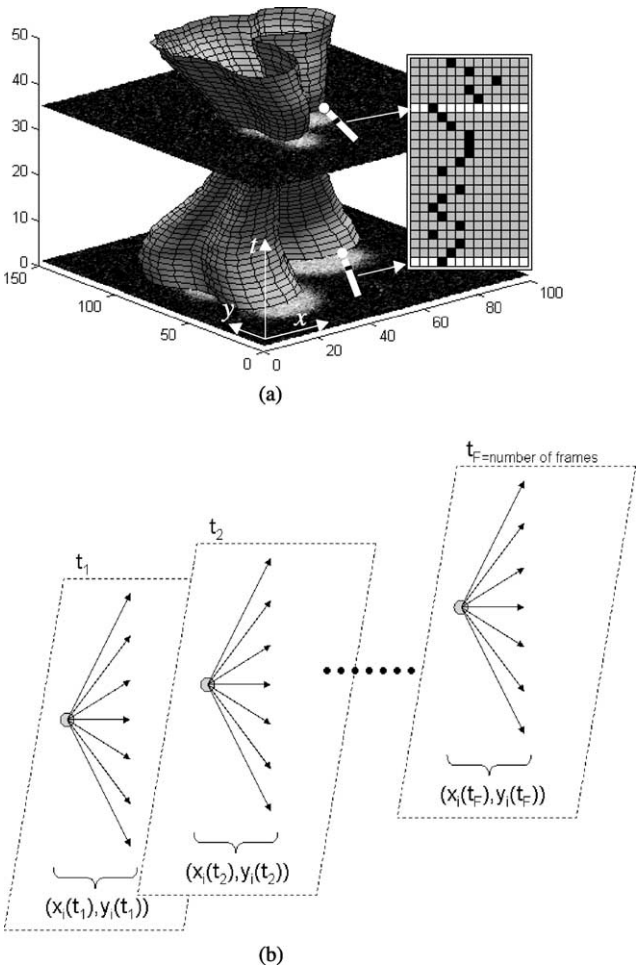

Fig. 1. Proposing a new ST-shape. (a) An illustration of an ST-shape overlaid on an image sequence. The search profiles of one landmark in two frames are shown in white. Examples of proposed landmark positions are shown as black squares. (b) The different choices of the new positions of landmark i in all frames.

We calculate a gray-level mismatch value $M_k(j,l)$ for each point along each search profile in all the frames according to

$$M_k(j,l) = (\mathbf{h}_{jk}(l) - \bar{\mathbf{y}}_k)^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{W}(\mathbf{h}_{jk}(l) - \bar{\mathbf{y}}_k) \tag{4}$$

where $1 \le k \le L$, $1 \le j \le F$, $1 \le l \le H$, $\mathbf{h}_{jk}(l)$ is a sub-profile of length $G-1$ anchored at the lth location of the search profile $\mathbf{h}_{jk}$, and $\mathbf{W}$ is a diagonal weighting matrix ($\mathbf{W} = \mathbf{I}$ was used). Additionally, we calculate a temporal discontinuity value $d_{jk}(l_j, l_{j-1})$, corresponding to moving the kth landmark in frame $j-1$ to location l_{j-1} and the kth landmark in frame j to location l_j, each along its respective search profile, according to

$$d_{jk}^2(l_j, l_{j-1}) = (\mathbf{c}_{jkx}(l_j) - \mathbf{c}_{j-1kx}(l_{j-1}))^2 + (\mathbf{c}_{jky}(l_j) - \mathbf{c}_{j-1ky}(l_{j-1}))^2 \tag{5}$$

where $\mathbf{c}_{jkx} = [x_{jk1}, x_{jk2}, \cdots, x_{jkH}]$ and $\mathbf{c}_{jky} = [y_{jk1}, y_{jk2}, \cdots, y_{jkH}]$ are the search profile coordinates of the kth landmark in the jth frame. We compare the accumulated costs of moving the kth landmark to the lth position in the jth frame, $2 \le j \le F$, from any of the H positions in frame $j-1$ and assign the least value to $A_k(j,l)$, i.e.

$$A_k(j,l) = \min\{t_{jkl1}, t_{jkl2}, \ldots, t_{jklH}\} \tag{6}$$

$$t_{jklm} = w_d d_{jk}(l,m) + w_m M_k(j,l) + A_k(j-1,m), \tag{7}$$

w_d and w_m, satisfy $w_d + w_m = 1$, control the relative importance of temporal discontinuity and gray-level mismatch. We also assign an index or a pointer, $P_k(j,l)$, to the location of the best landmark in the previous frames. Applying the same procedure to the kth landmark in all the F frames yields $F \times H$ accumulated values and $F \times H$ pointers (no temporal discontinuity cost is associated with the first frame).

To find the proposed positions of the kth landmark in all the frames we find the location, call it m_F, of the minimum accumulated cost along the search profile of the landmark in the last frame, frame F. Then we use m_F to find the proposed landmark position in the second last frame, frame $F-1$, as $m_{F-1} = P_k(F, m_F)$. Its coordinates will be $(\mathbf{c}_{F-1kx}(m_{F-1}), \mathbf{c}_{F-1ky}(m_{F-1}))$. In general the proposed coordinates of the kth landmark of the jth frame will be

$$(x,y) \;:\; (\mathbf{c}_{jkx}(m_j), \mathbf{c}_{jky}(m_j)) \tag{8}$$

$$m_j = P_k(j+1, m_{j+1}) \tag{9}$$

Tracking back to the first frame, we acquire the coordinates of the proposed positions of the kth landmark

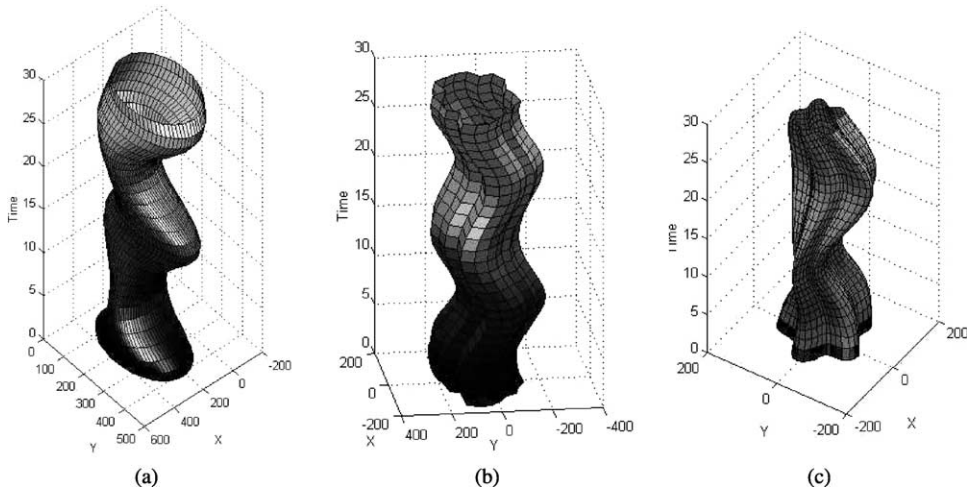

Fig. 2. Examples of synthetic spatio-temporal shapes. (a) Circle with translational motion, expansion and shrinkage in time. (b) 'Random star' with translational motion. (c) 'Sinusoidal star' with translational motion, expansion and shrinkage in time.

in all frames. Similarly, we obtain the proposed positions for all the landmarks ($1 \leq k \leq L$), which define the ST-shape changes $\mathrm{d}\hat{S}^{\langle 0\rangle}_{\text{proposed}}$.

Limiting the proposed sequence. Since the proposed ST-shape ($\hat{S}^{\langle 0\rangle} + \mathrm{d}\hat{S}^{\langle 0\rangle}_{\text{proposed}}$) will generally not conform to our model of reduced dimensionality and will not lie in the ASTSD, it cannot be accepted as an ST-shape estimate. Therefore, we need to find an acceptable ST-shape that is closest to the proposed one. This is done by first finding the pose parameters ($s^{\langle 1\rangle}$, $\theta^{\langle 1\rangle}$, and $\mathbf{t}^{\langle 1\rangle}$) that will align $\bar{S}$ to $\hat{S}^{\langle 0\rangle} + \mathrm{d}\hat{S}^{\langle 0\rangle}_{\text{proposed}}$ by mapping $\bar{S}$ to $M(s^{\langle 1\rangle}, \theta^{\langle 1\rangle})[\bar{S}] + \mathbf{t}^{\langle 1\rangle}$, then finding the extra ST-shape modifications $\mathrm{d}S^{\langle 1\rangle}$ which, when combined with the pose parameters, will map exactly to $\hat{S}^{\langle 0\rangle} + \mathrm{d}\hat{S}^{\langle 0\rangle}_{\text{proposed}}$. The latter is done by solving the following equation for $\mathrm{d}S^{\langle 1\rangle}$

$$M(s^{\langle 1\rangle}, \theta^{\langle 1\rangle})[\bar{S} + \mathrm{d}S^{\langle 1\rangle}] + \mathbf{t}^{\langle 1\rangle} = \hat{S}^{\langle 0\rangle} + \mathrm{d}\hat{S}^{\langle 0\rangle}_{\text{proposed}} \Rightarrow \tag{10}$$

$$\mathrm{d}S^{\langle 1\rangle} = M(s^{\langle 1\rangle}, \theta^{\langle 1\rangle})^{-1}[\hat{S}^{\langle 0\rangle} + \mathrm{d}\hat{S}^{\langle 0\rangle}_{\text{proposed}} - \mathbf{t}^{\langle 1\rangle}] - \bar{S} \tag{11}$$

where $M(s^{\langle 1\rangle}, \theta^{\langle 1\rangle})^{-1} = M((s^{\langle 1\rangle})^{-1}, -\theta^{\langle 1\rangle})$. In order to find the new shape parameters, $\mathbf{b}^{\langle 1\rangle}$ we need to solve $\mathrm{d}S^{\langle 1\rangle} = \mathbf{P}_t \mathbf{b}^{\langle 1\rangle}$, which, in general, has no solution since $\mathrm{d}S^{\langle 1\rangle}$ lies in a $2FL$ dimensional space whereas $\mathbf{P}_t$ spans only a t dimensional space. The best least-squares solution is obtained as

$$\mathbf{b}^{\langle 1\rangle} = \mathbf{P}_t^{\mathrm{T}} \mathrm{d}S^{\langle 1\rangle} \tag{12}$$

Finally, using the constraints discussed earlier, $b_{l\,\min} \leq b_l \leq b_{l\,\max}$ where $1 \leq l \leq t$, we limit these ST-shape variations and obtain an acceptable or allowable shape within the ASTSD. By updating $\mathbf{b}^{\langle 0\rangle}$ to $\mathbf{b}^{\langle 1\rangle}$ we have the new values for all the parameters $s^{\langle 1\rangle}$, $\theta^{\langle 1\rangle}$, $\mathbf{b}^{\langle 1\rangle}$, and $\mathbf{t}^{\langle 1\rangle}$.

Updating the estimate and reiterating. Similarly, new ST-shape estimates can be obtained

$$\begin{aligned}\hat{S}^{\langle i\rangle} &= M(s^{\langle i\rangle}, \theta^{\langle i\rangle})[\bar{S} + \mathbf{P}_t \mathbf{b}^{\langle i\rangle}] + \mathbf{t}^{\langle i\rangle} \rightarrow \hat{S}^{\langle i+1\rangle} \\ &= M(s^{\langle i+1\rangle}, \theta^{\langle i+1\rangle})[\bar{S} + \mathbf{P}_t \mathbf{b}^{\langle i+1\rangle}] + \mathbf{t}^{\langle i+1\rangle}\end{aligned} \tag{13}$$

for $i = 1, 2, 3, \ldots$. Checking for convergence can be done by examining the changes, i.e. if the new estimate is not much different (according to some predefined threshold) then the search is completed, otherwise we reiterate.

3. Results

We tested the method on synthetic and real data. A single synthetic example consisted of an ST-shape (Fig. 2) and a frame-sequence. The ST-shape data is first calculated and then used to generate the frame-sequence.

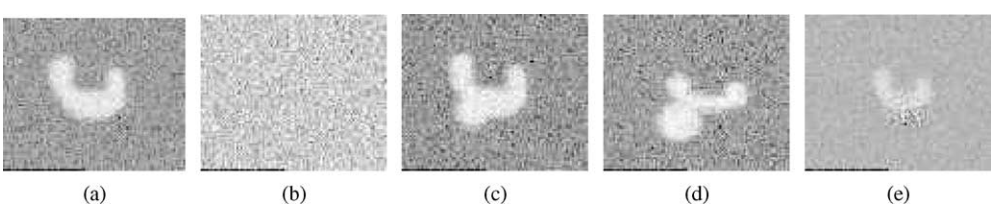

Fig. 3. Examples of synthetic frames with imperfections due to (a) global noise, (b) missing frame, (c) overlapping occlusion, (d) touching occlusion, and (e) local noise.

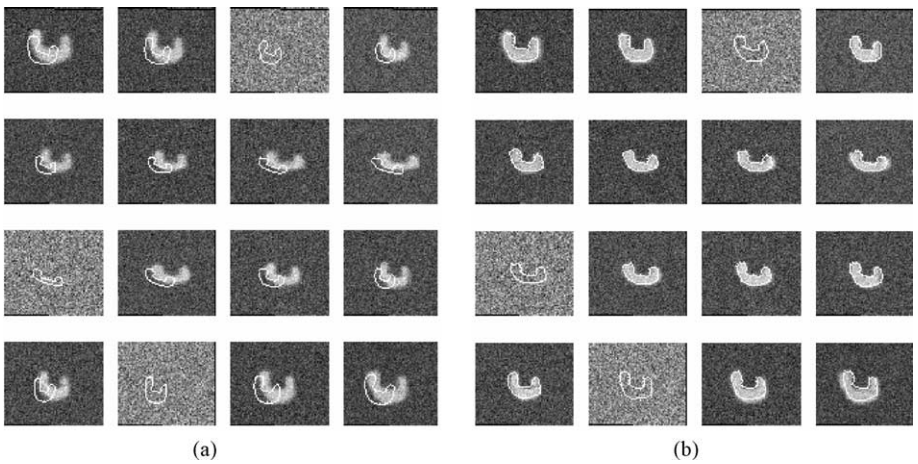

Fig. 4. Spatio-temporal segmentation example with three missing frames and global noise in all frames. After 23 iterations the initial ST-shape (overlaid in white on the leftmost 16 frames) deforms and detects the moving object (rightmost 16 frames).

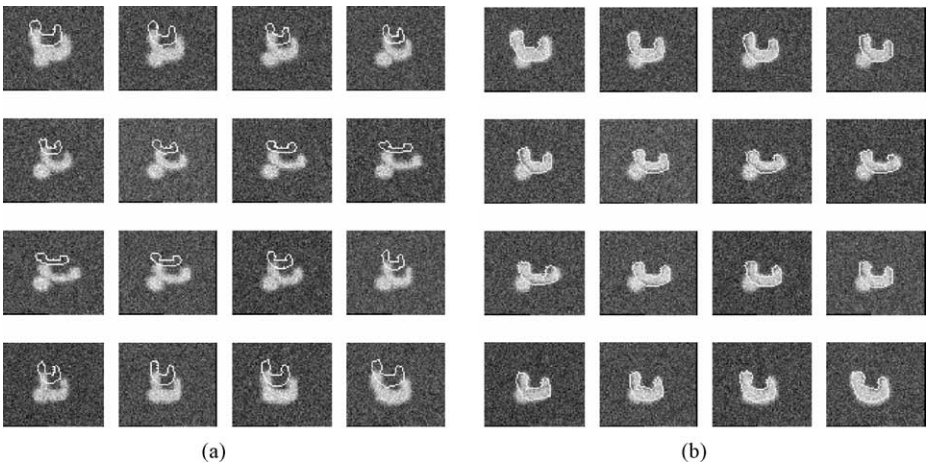

Fig. 5. Spatio-temporal segmentation example with overlapping occlusions and global noise in all frames. After 15 iterations the initial ST-shape (overlaid in white on the leftmost 16 frames) deforms and detects the moving object (rightmost 16 frames).

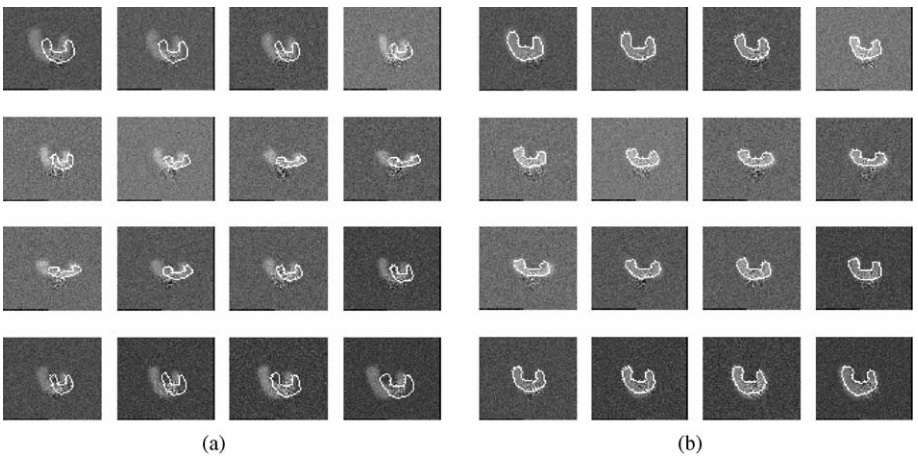

Fig. 6. Spatio-temporal segmentation example with strong local noise and moderate global noise in all frames. After 18 iterations the initial ST-shape (overlaid in white on the leftmost 16 frames) deforms and detects the moving object (rightmost 16 frames).

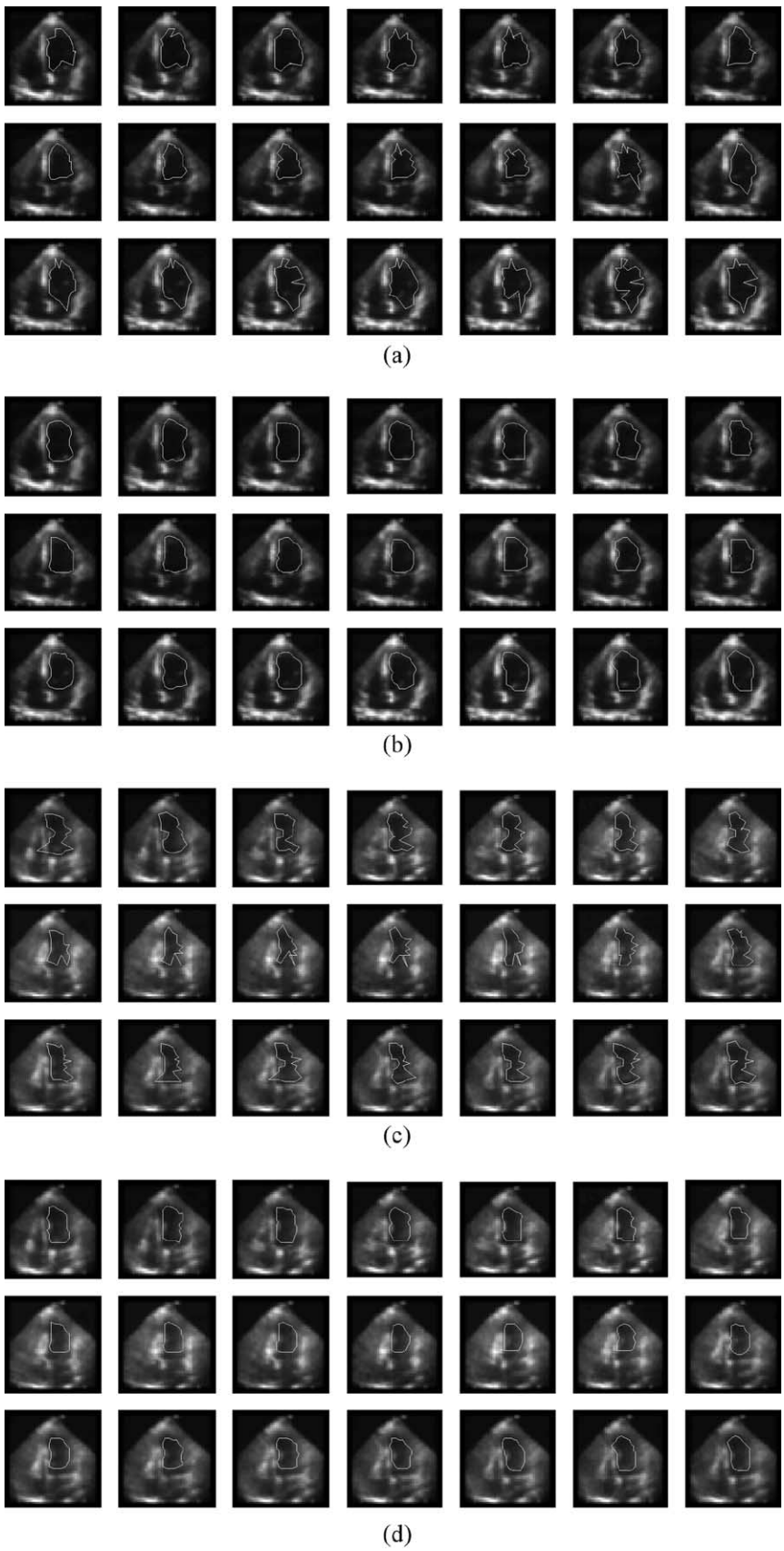

Fig. 7. Left-ventricular segmentation result from two echocardiographic image sequence. Ultrasound frames are shown with the ST-shape overlaid (a and c) before and (b and d) after projection onto the ASTSD (frames progress from left to right, top to bottom).

The ST-shapes are represented by a set of coordinates describing the shapes in all the frames. Each synthetic ST-shape consists of F frames. Each frame contains L landmark coordinates. Both the x and the y coordinates of each landmark move within a sequence according to sinusoidal functions with certain amplitudes and frequencies. The positions of the landmarks in the first frame and the amplitudes and frequencies of the sinusoidal functions are sampled from Gaussian distributed functions with given means and variances. This is done to produce similar ST-shapes to be used in the training stage. After the ST-shapes are produced, binary images are generated for all the frames in the sequences, by 'filling' the polygon areas generated from the landmark coordinates. Then the binary frame-sequences are smoothed by convolution with a Gaussian kernel. Noise and occlusions are added when producing a frame-sequence for testing the search algorithm.

To produce image sequences that imitate real-life imagery including artifacts, the synthetically generated image sequences used for both training and testing were deteriorated in different ways. Some examples of imperfections are shown in Fig. 3.

In the three synthetic examples presented here, the training was performed using 10 image sequences. Each sequence consisted of 16 frames. Each frame was of size 160×182 pixels (i.e. the size of $\Phi_V = 10 \times 16 \times 160 \times 182$). Twenty-five landmarks were used to represent each contour in each frame (size of $\Phi_S = 10 \times 16 \times 25 \times 2$). The gray-level search was conducted on a profile of length 41 pixels and the training profile was of length 13. Six ST-shape parameters were used to describe 98% of the total ST-shape variations. The training set was blurred and noised. In the image sequences, the minimum object intensity was 0, the maximum object intensity was 60, and the global noise variance was 100. Following are three examples of the ST-shape segmentation (the 16 frames in each sub-figure are ordered from left to right and top to bottom):

(1) Missing frames (Fig. 4). The result shows that the deformable ST-shape converged to the target object in all the frames and reasonable guesses were produced for the separated missing frames.
(2) Overlapping occlusion (Fig. 5). The result shows that the deformable ST-shape converged to the target object overcoming the problem of overlapping occlusions that appeared in all the frames. The radius of the occlusion was 15 pixels.
(3) Local noise (Fig. 6). The result shows that the deformable ST-shape converged to the target object in spite of the presence of strong local noise and moderate global noise in all the frames. The local noise variance used was 2500. The radius of the local noise region was 25 pixels and the spatial variance of the local noise was 200 pixels.

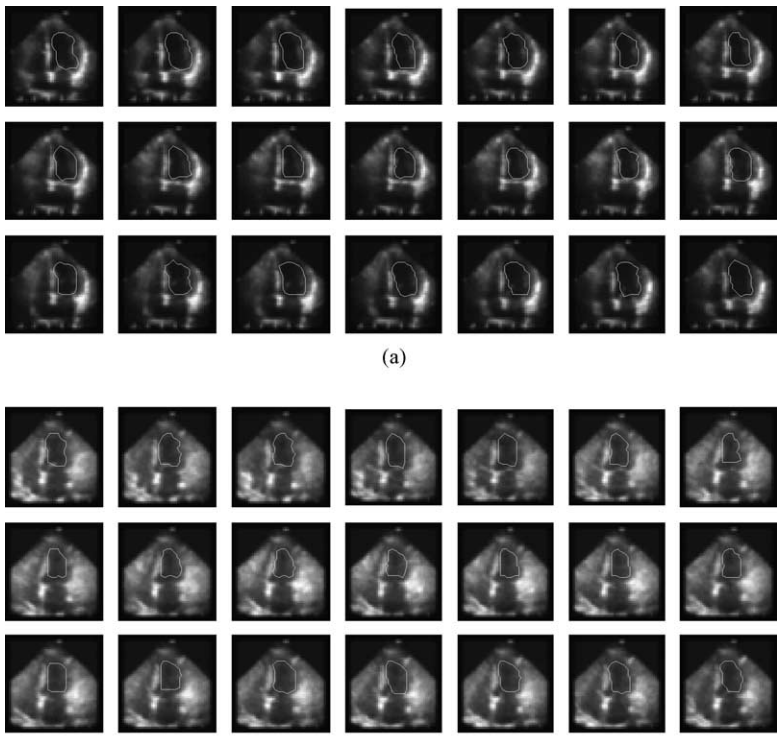

Fig. 8. Additional left-ventricular segmentation results from an echocardiographic image sequence (frames progress from left to right, top to bottom).

Table 1
Execution times and left-ventricular segmentation errors for six echocardiographic image sequences

Search sequence	Execution time (s)			Error (mm)				
	Gray-level training	Shape training	ST search	Mean	Median	SD	Max	Min
1	1.99	57.95	53.18	1.62	1.03	1.51	6.90	0.07
2	1.96	57.57	53.42	1.08	0.90	0.75	4.22	0.00
3	1.93	57.32	53.44	0.91	0.71	0.70	4.30	0.01
4	1.93	57.25	53.34	2.28	1.92	1.67	7.05	0.05
5	1.94	57.62	53.40	1.15	1.04	0.72	3.22	0.02
6	1.94	57.18	53.41	1.31	1.01	1.04	5.21	0.03
Average	1.95	57.48	53.36	1.39	1.10	1.06	5.15	0.03

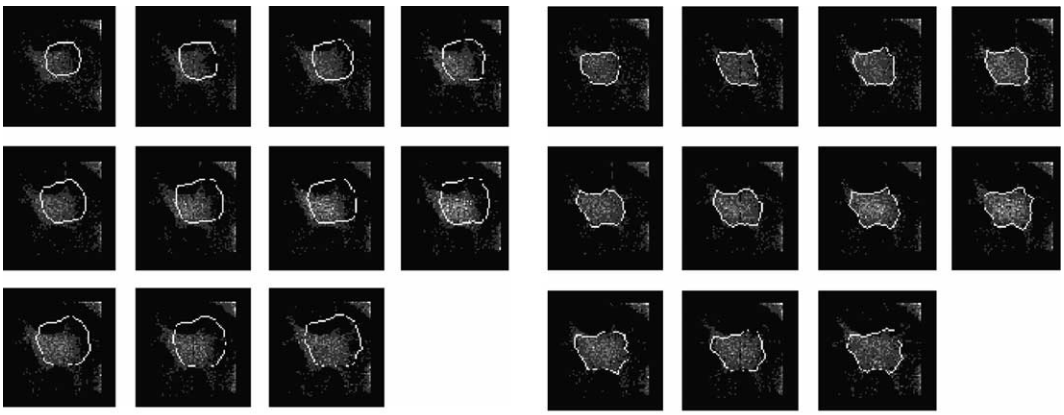

Fig. 9. Segmenting a 3D astrocyte cell (spatial z-axis replaces time). (a) The initial shape model and (b) the segmentation result overlaid in white on a fluorescence 3D image.

Furthermore, we tested the performance of the method on real echocardiographic image sequences for locating the ST-shape of the left ventricle. The training data set consisted of six frame sequences, each sequence included 21 frames, each frame was of size 255×254 pixels (i.e. the size of $\Phi_V = 6 \times 21 \times 255 \times 254$). The number of (x, y) landmark coordinates in each frame was 25 (size of $\Phi_S = 6 \times 21 \times 25 \times 2$). Three ST-shape parameters were used to explain 94.2% of the total ST-shape variations. The gray-level search was conducted on a profile of length 60 pixels and the training profile was of length 26 pixels. Fig. 7 illustrates how statistical ST prior knowledge is used to constrain the proposed segmentation and produce the final left ventricular segmentation. Fig. 8 shows additional segmentation results. Table 1 reports execution times and associated segmentation errors (MATLAB 5.3, 1.70 GHz Intel® Pentium® M Processor, 2 GB RAM). Leave-one-out cross-validation was used (i.e. the ST segmentation of each image sequence out of the six, utilized the shape and gray-level training results obtained from the remaining five manually labeled sequences).

We have also applied our method to segmenting astrocyte cells in a 3D fluorescence image, where the spatial z-axis replaces time. The training data set consisted of eight volumes (out of nine, leave-one-out validation), each included 11 image slices, each image was of size 128×128 pixels (i.e. the size of $\Phi_V = 8 \times 11 \times 128 \times 128$). The number of (x, y) landmark coordinates in each slice was 40 (size of $\Phi_S = 8 \times 11 \times 40 \times 2$). Seven shape parameters were used to explain 99.5% of the total shape variations. The gray-level search was conducted on a profile of length 40 pixels and the training profile was of length 12 pixels. Fig. 9 illustrates an example segmentation results.

4. Conclusion

Motivated by the fact that many image analysis applications require robust methods for representing, locating, and analyzing non-rigid time-varying shapes, we presented an extension of 2D ASM to 2D + time. This method models the gray-level information and the ST variations of a time-varying object in a training set. The model is then used for locating similar moving objects in a new image sequence. The segmentation technique is based

on deforming a ST-shape to better fit the image sequence data only in ways consistent with the training set. The proposed deformations are calculated by minimizing an energy function using dynamic programming. The energy function includes terms reflecting temporal smoothness and gray-level information constraints. We demonstrated the suitability of the method for segmenting objects with specific motion patterns using synthetic and real echo cardiographic data. Extending our current work to include temporal translation and temporal scaling parameters may assist in searching through longer image sequences for target dynamic shapes of varying velocities. Usefulness of multi-resolution search was demonstrated for 2D ASM [7], a similar extension that includes multiple temporal resolutions may be equally beneficial.

References

[1] A. Amini, T. Weymouth, R. Jain, Using dynamic programming for solving variational problems in vision, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (9) (1990) 855–867.
[2] M. Black, Y. Yacoob, Recognizing facial expressions in image sequences using local parametrized models of image motion, International Journal of Computer Vision 25 (1) (1997) 23–48.
[3] C. Bonciu, C. Léger, J. Thiel, A Fourier–Shannon approach to closed contour modelling, Bioimaging 6 (1998) 111–125.
[4] L. Cohen, On active contour models and balloons, Computer Vision, Graphics, and Image Processing: Image Understanding 53 (2) (1991) 211–218.
[5] T. Cootes, G. Edwards, C. Taylor, Active appearance models, Proceedings of the European Conference on Computer Vision 2 (1998) 484–498.
[6] T. Cootes, C. Taylor, D. Cooper, J. Graham, Active shape models—their training and application, Computer Vision and Image Understanding 61 (1) (1995) 38–59.
[7] T. Cootes, C. Taylor, A. Lanitis, Active shape models: evaluation of a multi-resolution method for improving image search, Proceedings of the British Machine Vision Conference (1994) 327–336.
[8] R. Grzeszczuk, D. Levin, 'Brownian strings': segmenting images with stochastically deformable contours, Pattern Analysis and Machine Intelligence 19 (10) (1997) 1100–1114.
[9] I. Herlin, C. Nguyen, C. Graffigne, A deformable region model using stochastic processes applied to echocardiographic images, Proceedings of the Computer Vision and Pattern Recognition (1992) 534–539.
[10] A. Hill, C. Taylor, Automatic landmark generation for point distribution models, Proceedings of the British Machine Vision Conference (1994) 429–438.
[11] A. Hill, A. Thornham, C. Taylor, Model-based interpretation of 3D medical images, Proceedings of the British Machine Vision Conference (1993) 339–348.
[12] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, International Journal of Computer Vision 1 (4) (1988) 321–331.
[13] B. Lelieveldt, S. Mitchell, J. Bosch, R. van der Geest, M. Sonka, J. Reiber, Time-continuous segmentation of cardiac image sequences using active appearance motion models, Proceedings of the Information Processing in Medical Imaging (2001) 446–452.
[14] F. Leymarie, M. Levine, Tracking deformable objects in the plane using an active contour model, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (6) (1993) 617–634.
[15] S. Lobregt, M. Viergever, A discrete dynamic contour model, IEEE Transactions on Medical Imaging 14 (1) (1995) 12–24.
[16] T. McInerney, D. Terzopoulos, A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4D image analysis, Computerized Medical Imaging and Graphics 19 (1) (1995) 69–83.
[17] T. McInerney, D. Terzopoulos, Topologically adaptable snakes, Proceedings of the International Conference on Computer Vision (1995) 840–845.
[18] W. Niessen, J. Duncan, M. Viergever, B. Romeny, Spatiotemporal Analysis of Left Ventricular Motion, SPIE Medical Imaging, SPIE Press, 1995, pp. 250–261.
[19] S. Sclaroff, J. Isidoro, Active blobs, Proceedings of the Sixth IEEE International Conference on Computer Vision (1998) 1146–1153.
[20] A. Signh, L. Von Kurowski, M. Chiu, Cardiac MR image segmentation using deformable models, SPIE Proceedings of the Biomedical Image Processing and Biomedical Visualization 1905 (1993) 8–28.
[21] A. Singh, D. Goldgof, D. Terzopoulos, Deformable Models in Medical Image Analysis, IEEE Computer Society, Silver Spring, MD, ISBN: 0818685212, 1998.
[22] L. Staib, J. Duncan, Boundary finding with parametrically deformable models, IEEE Transactions on Pattern Analysis and Machine Intelligence 14 (11) (1992) 1061–1075.
[23] S. Stark, S. Fuchs, A method for tracking the pose of known 3-D objects based on an active contour model, Proceedings of the International Conference on Pattern Recognition (1996) 905–909.
[24] D. Terzopoulos, R. Szeliski, Tracking with Kalman snakes, in: A. Blake, A. Yuille (Eds.), Active Vision, MIT Press, Cambridge MA, 1992, pp. 3–20, Chapter 1.