Deformable Spatio-Temporal Shape Models: Extending ASM to 2D+Time

Ghassan Hamarneh and Tomas Gustavsson
Dept. of Signals and Systems, Chalmers University of Technology
{ghassan , gustavsson}@s2.chalmers.se

Abstract

This paper extends 2D Active Shape Models to 2D+time by presenting a method for modelling and segmenting spatio-temporal shapes (ST-shapes). The modelling part consists of constructing a statistical model of ST-shape parameters. The model obtained describes the principal modes of variation of the ST-shape in addition to certain constraints on the allowed variations. An active approach is used in segmentation; an initial ST-shape is deformed to better fit the data and the optimal proposed deformation is calculated using dynamic programming. The results presented show the proposed method detecting ST-shapes in a variety of synthetic noisy data. Preliminary results on real data are also reported.

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 modelling and locating non-rigid time-varying object shapes. One approach for dealing with such objects is the use of deformable models. Deformable models [18] such as Snakes [12] and its variants [4,8,9,14,16], 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 initialisation sensitivity. They were also designed to be a general model 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 led by Active Shape Models (ASM) [6] were introduced. In ASM the statistical variation of shapes is modelled 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 [13,17,15]. 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' [20]. Motion tracking using deformable models has been used for tracking non-rigid structures such as blood cells [13] and much attention has been given to the human heart and the tracking of the left

ventricle in both 2D and 3D [17,15]. 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. We extend ASM [6] to include knowledge of temporal shape variations and present a ST-shape modelling 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 (HCI).

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 parameterised. 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 modelling of grey-level information. The task is then to locate an ST-shape given a new unknown image sequence. An average ST-shape is first initialised, 'optimal' deformations are then proposed, and then the deformations are forced 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 grey-level appearance constraints. The search for the optimum proposed change is done using dynamic programming.

2 Method

2.1 Statistical ST-Shape Variation

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

The ST-shape parameters. We now introduce S_i to denote the parameter vector representing the i^{th} ST-shape. Parameterisation is done using landmarks (other shape parameterisation methods may be utilized, e.g. Fourier descriptors [3] or B-Splines [19]). Landmarks are labelled 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 $\Phi_S = \left[S_1, S_2, \ldots, S_N\right]$, where $\Phi_S(i) = S_i = \left[p_{i1}, p_{i2}, \ldots, p_{iF}\right]$ is the i^{th} ST-shape containing F shapes and $S_i(j) \equiv \Phi_S(i,j) = p_{ij}$ is the j^{th} shape of the i^{th} ST-shape. p_{ij} can be written as $p_{ij} = \left[x_{ij1}, y_{ij1}, x_{ij2}, y_{ij2}, \ldots, x_{ijL}, y_{ijL}\right]$ where $x_{ijk} = p_{ij}(k,1) \equiv \Phi_S(i,j,k,1)$ and $y_{ijk} =$

$p_{ij}(k,2) \equiv \Phi_S(i,j,k,2)$ are the x, y coordinates of the k^{th} landmark of the shape p_{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. However, if the observed motion patterns in the training sequences span different time intervals, temporal re-sampling or aligning that incorporates temporal scaling might be needed. If these differences are insignificant, their effects can be interpreted and modelled as shape variations.

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 region of dthis $2FL$ 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 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 k^{th} eigenvalue of C_S ($\lambda_k \ge \lambda_{k+1}$) and is equal to the variance along the k^{th} PC. The mean ST-shape is calculated as $\overline{S} = \frac{1}{N}\sum_{i=1}^{N} S_i$. 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, $\overline{S}$, and a linear combination of the principal modes of variation, $\mathbf{Pb}_i$. This gives $S_i = \overline{S} + \mathbf{Pb}_i$ where $\mathbf{b}_i = \begin{bmatrix} b_{i,1} & b_{i,2} & \ldots & b_{i,2FL} \end{bmatrix}^T$ and $\mathbf{P} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \ldots & \mathbf{p}_{2FL} \end{bmatrix}$. 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 λ_l. Assuming that the first t (out of $2FL$) PCs explain a sufficiently high percentage of the total variance of the original data, the fundamental equation becomes

$$S = \overline{S} + \mathbf{Pb} \tag{1}$$

where $\mathbf{b} = \begin{bmatrix} b_1 & b_2 & \ldots & b_t \end{bmatrix}^T$, $\mathbf{P} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \ldots & \mathbf{p}_t \end{bmatrix}$, 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 Grey-Level Training

The information contained in the ST-shape model alone is typically not enough for spatio-temporal segmentation. Therefore, additional representative information about the intensities or grey-levels relating to the object is also desired and collected in the grey-level training stage. In the search stage, new estimates of the ST-shape are sought that will better match the grey-level prior knowledge. Different grey-level representative information can be used, e.g. gathering the intensity values in the entire patch contained within the object [5] or parameterising 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 neighbouring ones. For the k^{th} landmark this profile is given by

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

where $\mathbf{y}_{ijk}$ is the representative profile for the k^{th} landmark in the j^{th} shape of the i^{th} ST-shape. Using grey-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:

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

where $\mathbf{t} = \begin{bmatrix} t_x & t_y & t_x & t_y & \dots & t_x & t_y \end{bmatrix}$ is of length $2FL$. $M(s,\theta)[S] + \mathbf{t}$ scales, rotates, and translates S by s, θ, and $\mathbf{t}$, respectively. Both $\overline{S}$ and $\mathbf{P}$ are obtained from the training stage. A typical initialisation would set $\mathbf{b}^{\langle 0\rangle}$ to zero, and $s^{\langle 0\rangle}$, $\theta^{\langle 0\rangle}$, and $\mathbf{t}^{\langle 0\rangle}$ to values that put the initial sequence in the vicinity of the target.

Proposing a new sequence. For each landmark, say the k^{th} landmark in the j^{th} frame, we define a search profile $\mathbf{h}_{jk} = \begin{bmatrix} h_{jk1} & h_{jk2} & \dots & h_{jkH} \end{bmatrix}$ that is differentiated and normalized as done with the training profiles. This gives H^F possibilities for the proposed positions of the k^{th} landmarks in the F frames, see Figure 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 grey-level values surrounding the current landmarks and those

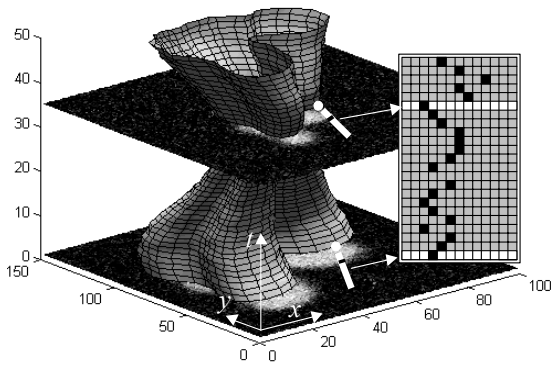


Figure 1. An illustration of an ST-shape overlaid on an image sequence. The search profiles of one landmark in 2 frames are shown in white. Examples of proposed landmark positions are shown as black squares.

expected values found in the grey-level training stage. In the following paragraphs, we detail our implementation of dynamic programming.

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

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

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 l^{th} location of the search profile $\mathbf{h}_{jk}$, and $\mathbf{W}$ is a weighting matrix. Additionally, we calculate a temporal discontinuity value $d^2_{jk}(l_j, l_{j-1})$, corresponding to moving the k^{th} landmark in frame $j-1$ to location l_{j-1} and the k^{th} landmark in frame j to location l_j, each along its respective search profile, according to

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

where $\mathbf{c}_{jkx} = \begin{bmatrix} x_{jk1} & x_{jk2} & \dots & x_{jkH} \end{bmatrix}$ and $\mathbf{c}_{jky} = \begin{bmatrix} y_{jk1} & y_{jk2} & \dots & y_{jkH} \end{bmatrix}$ are the search profile coordinates of the k^{th} landmark in the j^{th} frame (in our implementation, the same temporal discontinuity weight is assigned to all landmarks in all frames. This need not be the case, especially when prior knowledge about the typical velocities of different regions of the object is available). We compare the accumulated costs of moving the k^{th} landmark to the l^{th} position in the j^{th} 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.

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

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

w_d and w_m, satisfy $w_d + w_m = 1$, control the relative importance of temporal discontinuity and grey-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 k^{th} 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 k^{th} 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 $\left(\mathbf{c}_{F-1kx}(m_{F-1}), \mathbf{c}_{F-1ky}(m_{F-1})\right)$. In general the proposed coordinates of the k^{th} landmark of the j^{th} frame will be

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

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

Tracking back to the first frame, we acquire the coordinates of the proposed positions of the k^{th} landmark in all frames. Similarly, we obtain the proposed positions for all the landmarks ($1 \le k \le L$), which define the ST-shape changes $d\hat{S}^{\langle 0 \rangle}_{proposed}$.

Limiting the proposed sequence. Since the proposed ST-shape ($\hat{S}^{\langle 0\rangle} + d\hat{S}^{\langle 0\rangle}_{proposed}$) 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 $\overline{S}$ to $\hat{S}^{\langle 0\rangle} + d\hat{S}^{\langle 0\rangle}_{proposed}$ by mapping $\overline{S}$ to $M\left(s^{\langle 1\rangle}, \theta^{\langle 1\rangle}\right)\left[\overline{S}\right] + \mathbf{t}^{\langle 1\rangle}$, then finding the extra ST-shape modifications $dS^{\langle 1\rangle}$ which, when combined with the pose parameters, will map exactly to $\hat{S}^{\langle 0\rangle} + d\hat{S}^{\langle 0\rangle}_{proposed}$. The latter is done by solving the following equation for $dS^{\langle 1\rangle}$

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

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

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

$$\mathbf{b}^{\langle 1\rangle} = \mathbf{P}^T dS^{\langle 1\rangle} \tag{12}$$

Finally, using the constraints discussed earlier, $b_{l\min} \le b_l \le b_{l\max}$ where $1 \le l \le t$, 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

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

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.

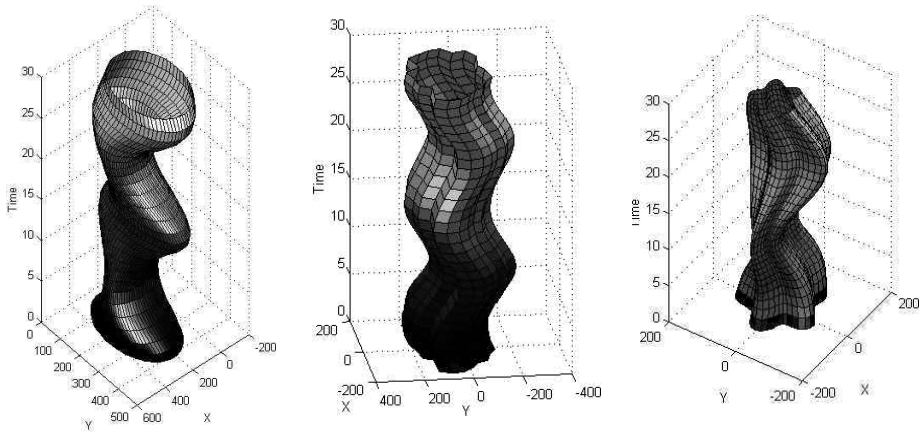
3 Results

We tested the method on synthetically generated data (Figure 2). A single synthetic example consisted of an ST-shape and a frame-sequence. The ST-shape data is first calculated and then used to generate the frame-sequence. 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 artefacts, the synthetically generated image sequences used for both training and testing were deteriorated in different ways. Some examples of imperfections are shown in Figure 3.

In the three examples results presented, the training was performed using 10 image sequences. Each sequence consisted of 16 (160x182 pixels) frames. 25 landmarks were used to represent each contour in each frame. The grey-level search was conducted on a profile of length 41 pixels and the training profile was of length 13. In our model we used six ST-shape parameters describing 98% of the total ST-shape variations. Following are three examples of the ST-shape segmentation (the frames in the figures are ordered from left to right and top to bottom): (1) Missing Frames (Figure 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 (See Figure 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. (3) Local noise (Figure 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.



(a) (b) (c)

Figure 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.

(a) (b) (c) (d) (e)

Figure 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.

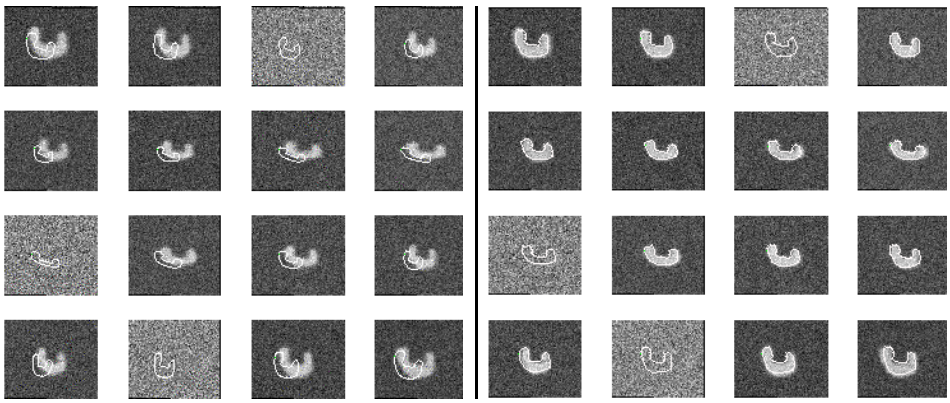


Figure 4. Spatio-temporal segmentation example with 3 missing frames and global noise in all frames. After 23 iterations the initial ST-shape (overlaid in white on the leftmost 16 frames) deforms and detects the moving object (rightmost 16 frames).

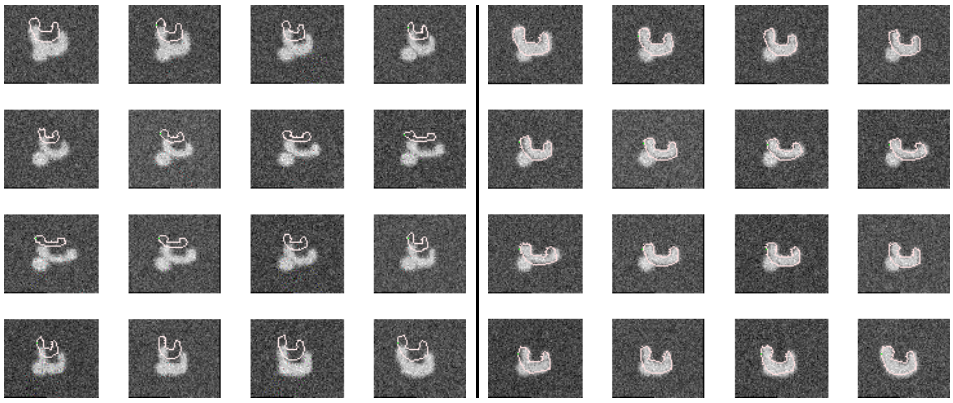


Figure 5. Spatio-temporal segmentation example with overlapping occlusions and global noise in all frames. After 15 iterations the initial ST-shape (overlaid in white on the leftmost 16 frames) deforms and detects the moving object (rightmost 16 frames).

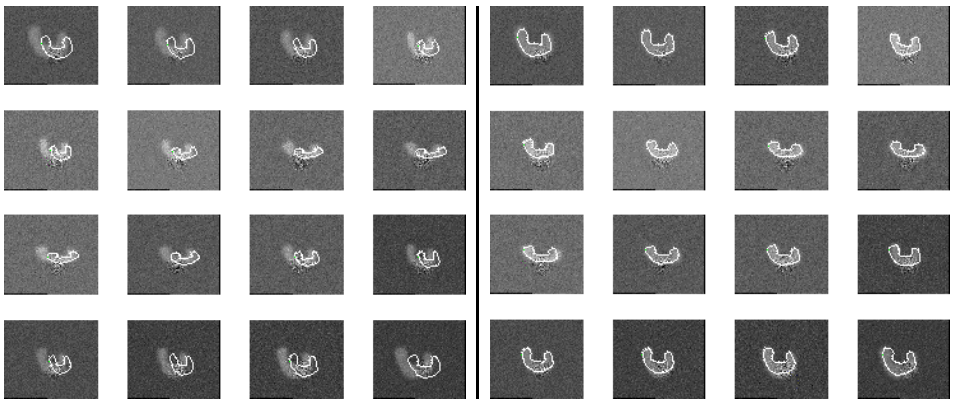
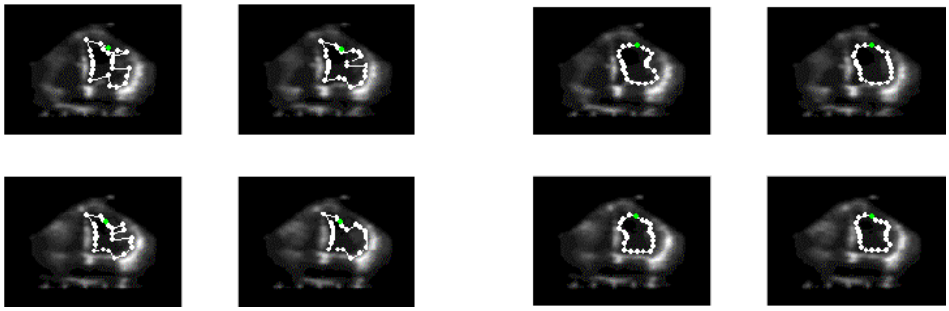
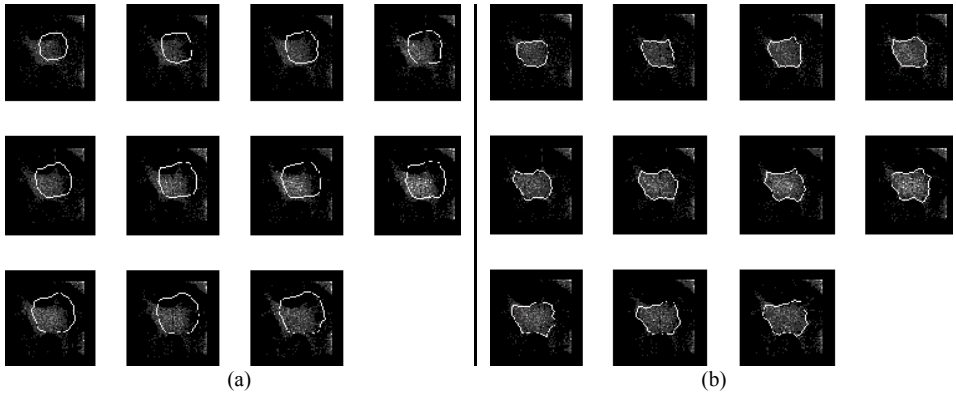


Figure 6. Spatio-temporal segmentation example with strong local noise and moderate global noise in all frames. After 18 iterations the initial ST-shape (overlaid in white on the leftmost 16 frames) deforms and detects the moving object (rightmost 16 frames).



(a) (b)

Figure 7. Left-ventricular segmentation result on smoothed real echocardiographic data. Four frames are shown with the ST-shape overlaid (a) before and (b) after projection onto the ASTSD.



(a) (b)

Figure 8. 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.

4 Conclusion

Motivated by the fact that many image analysis applications require robust methods for representing, locating, and analysing time-varying shapes, we presented Deformable Spatio-Temporal Shape Models: an extension of 2D ASM to 2D+time. This method models the grey-level information and the ST-shape variations of a time-varying object in a training set. The model is then used for locating similar objects in a new image sequence. The segmentation technique is based on deforming a spatio-temporal 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 grey-level information constraints. The method was tested and succeeded in segmenting synthetic spatio-temporal shapes in noisy image sequences. We are aware of remaining issues. More work is needed in order to assess accuracy, robustness and applicability for real-life imagery (specifically, boundary detection in real-time cross-sectional echocardiography). Besides our current work on applying the technique to real medical data (Figure 7, Figure 8), we are also considering a multi-resolution extension (similar to the one presented in [7]) and a time-scaling and time-translation feature.

References

[1] Amini A., Weymouth T., Jain R., Using dynamic programming for solving variational problems in vision. IEEE PAMI, 1990, 12(9), 855 -867.

[2] Black M., Yacoob Y., Recognizing Facial Expressions in Image Sequences using Local Parametrized Models of Image Motion. IJCV, 1997, 25(1), 23-48.

[3] Bonciu C., Léger C., Thiel J., A Fourier-Shannon approach to closed contour modeling. Bioimaging, 1998, 6, 111-125.

[4] Cohen L., On active contour models and balloons. CVGIP: Image understanding, 1991, 53(2), 211-218.

[5] Cootes T., Edwards G., Taylor C., Active Appearance Models. Proceedings of the European Conference on Computer Vision, 1998, (H.Burkhardt & B.Neumann Ed.s). Vol.2, pp.484-498, Springer, 1998.

[6] Cootes T., Taylor C., Cooper D., Graham J., Active Shape Models - Their training and application. Computer Vision and Image Understanding, 1995, 61(1), pp.38-59.

[7] Cootes T., Taylor C., Lanitis A., Active Shape Models: Evaluation of a Multi-Resolution Method for Improving Image Search. Proceedings of the British Machine Vision Conference, 1994, 327-336.

[8] Grzeszczuk R., Levin D., "Brownian strings": Segmenting images with stochastically deformable contours. IEEE PAMI, 1997, 19(10), pp.1100 -1114.

[9] Herlin I., Nguyen C., Graffigne C., A deformable region model using stochastic processes applied to echocardiographic images. Proceedings of Computer Vision and Pattern Recognition, 1992, pp.534 -539.

[10] Hill A., Taylor C., Automatic Landmark Generation for Point Distribution Models. BMVC, 1994, 2, pp. 429-438.

[11] Hill A., Thornham A., Taylor C., Model-Based Interpretation of 3D Medical Images. Fourth BMVC Conference, 1993, Guilford, England, pp.339-348.

[12] Kass M., Witkin A., Terzopoulos D., Snakes: Active Contour Models. International Journal of Computer Vision, 1988, 1(4), pp.321-331.

[13] Leymarie F., Levine M., Tracking deformable objects in the plane using an active contour model. IEEE Transactions on PAMI, 1993, 15(6), pp.617-634.

[14] Lobregt S., Viergever M., A discrete dynamic contour model. IEEE Transactions on Medical Imaging, 1995, 14(1), pp.12 -24.

[15] McInerney T., Terzopoulos D., 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, 1995, 19(1), pp.69-83.

[16] McInerney T., Terzopoulos D., Topologically adaptable snakes. Proceedings of the Fifth International Conference on Computer Vision, 1995, pp.840 -845.

[17] Signh A., Von Kurowski L., Chiu M., Cardiac MR image segmentation using deformable models. Biomedical Image Processing and Biomedical Visualization, SPIE Proc., Vol.1905, pp.8-28.

[18] Singh A., Goldgof D., Terzopoulos D., Deformable Models in Medical Image Analysis. IEEE Computer Society, ISBN: 0818685212.

[19] Stark K., Fuchs S., A method for tracking the pose of known 3-D objects based on an active contour model. Proceedings of the 13th International Conference on Pattern Recognition, 1996, Vol.1, pp.905 -909.

[20] Terzopoulos D., Szeliski R., Tracking with Kalman snakes. Active Vision, A. Blake and A. Yuille (eds.), MIT Press, Cambridge 1992, MA, Ch.1, pp.3-20.