A Shape-Guided Deformable Model with Evolutionary Algorithm Initialization for 3D Soft Tissue Segmentation

Tobias Heimann, Sascha Münzing, Hans-Peter Meinzer, and Ivo Wolf

Div. Medical and Biological Informatics, German Cancer Research Center,
69120 Heidelberg, Germany
t.heimann@dkfz.de

Abstract. We present a novel method for the segmentation of volumetric images, which is especially suitable for highly variable soft tissue structures. Core of the algorithm is a statistical shape model (SSM) of the structure of interest. A global search with an evolutionary algorithm is employed to detect suitable initial parameters for the model, which are subsequently optimized by a local search similar to the Active Shape mechanism. After that, a deformable mesh with the same topology as the SSM is used for the final segmentation: While external forces strive to maximize the posterior probability of the mesh given the local appearance around the boundary, internal forces governed by tension and rigidity terms keep the shape similar to the underlying SSM. To prevent outliers and increase robustness, we determine the applied external forces by an algorithm for optimal surface detection with smoothness constraints. The approach is evaluated on 54 CT images of the liver and reaches an average surface distance of 1.6 ± 0.5 mm in comparison to manual reference segmentations.

1 Introduction

Statistical shape models (SSMs), as introduced by Cootes et al. [1], have become a popular choice for analyzing medical images. Due to their strict constraints on the allowable shapes, they offer robust performance even in case of image artifacts and low signal-to-noise ratio. The price for this robustness, however, is that the limited deformations often impede the exact adaptation to the structure of interest. This holds especially true if the target structure consists of soft tissue with a large amount of natural variability: Besides the systematic variation, there is always a part of essentially random perturbation that cannot be captured adequately by a global deformation model as used in SSMs. A solution to this problem is to combine the SSM with a freely deformable, energy-based model as the original Snake by Kass et al. [2]. In this hybrid approach, the external energy describes the difference to the data and the internal energy is based on the difference between the current shape and the closest SSM [3,4]. The challenge is to balance both forces properly and to ensure that the additional freedom does not affect the robust convergence properties of the SSM.

N. Karssemeijer and B. Lelieveldt (Eds.): IPMI 2007, LNCS 4584, pp. 1–12, 2007.

Another critical element when using deformable models is the question of initialization: Although multi-resolution techniques provide a fairly large capture range, the model still has to be roughly aligned and oriented to the structure of interest so that the iterative search procedure can lock onto the target. Apart from the obvious manual initialization, there have been several suggestions how to estimate the target location automatically, e.g. by gray-value thresholding and subsequent morphological operations [5]. While these estimates can work for specialized cases, a more general solution is to conduct an initial global search: In the early 90s, Hill et al. already used a genetic algorithm for this purpose [6], but the method was later given up in favor of the iterative search of the Active Shape Model (ASM). Recently, de Bruijne and Nielsen successfully employed particle filtering for detecting 2D SSMs in radiographs [7]. We are not aware of any former attempts to localize a 3D SSM by global search methods.

In this paper, we present a segmentation method that offers a solution for both of the above presented shortcomings: A reliable initialization of the model using a global search in a down-sampled version of the image, and a robust deformable surface model with enough variability for an accurate segmentation. As an example application, we will employ the case of liver segmentation in abdominal CT datasets.

2 Statistical Model

The foundation of the proposed segmentation approach is a statistical model of the structure of interest, as introduced by Cootes et al. in [1]. It is built from a set of segmented training images and consists of two parts: A geometrical model describing the shape and a local appearance model describing the boundary.

2.1 Geometrical Model

The geometrical model is represented by a point distribution model (PDM), i.e. a dense collection of landmark points on the surface of the object. Each training shape t is described by a single vector $\mathbf{x_t}$ of concatenated landmark coordinates. Averaging all vectors produces the mean shape $\bar{\mathbf{x}}$, and a principal component analysis (PCA) on the covariance matrix yields the principal modes of variation p_m and the respective variances λ_m. Using a limited number of c parameters y_m, all valid shapes $\mathbf{x}$ can then be approximated by

$$\mathbf{x} \cong \bar{\mathbf{x}} + \sum_{m=1}^{c} y_m \mathbf{p}_m \tag{1}$$

A prerequisite for building a shape model is that all landmarks are situated at corresponding locations on all training samples. A number of different methods of how to automatically establish the required correspondences in 3D have been proposed in recent years, including registering mesh to mesh [8], volume to volume [9] or mesh to volume [10]. In this work, we employ a population-based approach for finding correspondences which minimizes a cost function based on the description length of the resulting shape model [11]. As proposed in [12], we utilize continuous landmark resampling during optimization to achieve a homogeneous point distribution over the entire surface.

2.2 Local Appearance Model

To detect the modeled shape in new image data, an additional model of the local gray-value appearance around the boundary is employed. A common method is to sample profiles g_{it} perpendicular to the surface at each landmark i in all training images t. Mean profiles $\bar{g}_i$ and the modes of variation for each landmark can then be determined using PCA. However, due to the essentially non-linear profile distributions in many medical imaging tasks, this method may not be optimal. In [13], an alternative approach to model local appearance based on kNN-classifiers is proposed: In addition to the true profiles g_{it}, a number of shifted profiles (i.e. translated towards the inside and outside of the object) is sampled. During model search, the probability $p(b|g)$ of a profile lying on the boundary can then be estimated by querying the k nearest neighbors to g and calculating the ratio of true profiles $b_k(g)$ among them. To prevent zero probabilities, we use a moderated kNN-classifier [14] and define:

$$p(b|g) = \frac{b_k(g) + 1}{k + 2} \tag{2}$$

Obviously, the accuracy of the classifier improves with the quantity of data available. With a limited number of training images on-hand, clustering landmarks to groups of similar appearance can help to raise the performance.

3 Model Initialization

In this work, we use an algorithm based on the concepts of evolutionary programming [15] and evolution strategies [16] to initialize the shape model. Though these two methods were developed independent from each other, they only differ in details and share most of the important properties. Both are global search algorithms maintaining a population of different solutions which evolve by following the ”survival of the fittest” rule: A higher fitness value, as determined by an evaluation function, increases the probability that individual solutions are mutated and reproduced. After several generations (i.e. iterations of the process), the population converges to one or several local maxima (see also Fig. 1). The basic procedure is:

```
Initialize population
Repeat
  Evaluation of fitness
  Selection by random sampling
  Gaussian mutation
End
```

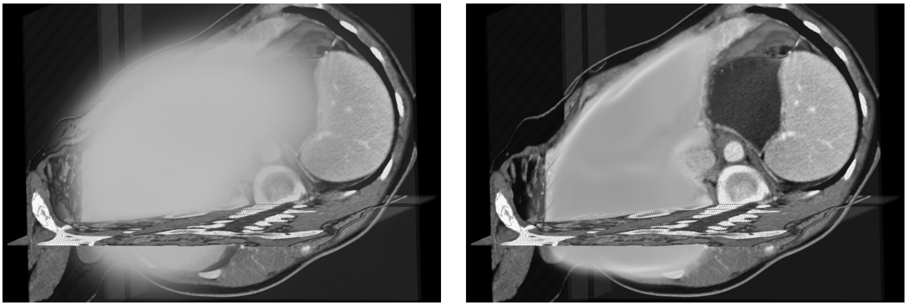

Fig. 1. Shape population in an image after initialization (left) and after convergence (right). All shapes are rendered as transparent solids, overlap increases the density color-coded from light to dark. While the left population is spread widely and only appears as a diffuse cloud, the right population is centered closely around the final solution. Both images display 5000 individual shapes.

The main differences to genetic algorithms [17] that were employed for model matching by Hill et al. [6] are the following: First, solutions do not have to be encoded as bit-strings in artificial chromosomes but are stored as real-valued vectors. Second, there is no cross-over operator for mutation (recombination) and no bit-inversion: Instead, solutions are modified by adding a random vector from a multivariate zero-mean Gaussian distribution.

3.1 Step by Step

One individual in our population represents one possible shape configuration, consisting of a similarity transform and c shape parameters. For initialization, all shape parameters y_m are randomized according to their variance λ_m. The pose parameters (translation, rotation, scale) are estimated from the respective mean values of the training samples (using relative coordinates for location) and are also randomized using a Gaussian distribution.

To evaluate the fitness w_s of an individual shape s, the probabilities $p(b|g)$ from Eq. 2 are estimated for all landmarks i and multiplied:

$$w_s = \exp\left(\frac{v}{n}\sum_i \log p_i(b|g_i)\right) \tag{3}$$

Here, n represents the number of landmarks and v is a constant determining the speed of convergence. We use $v = 5$ for all experiments. The subtle difference to the weighting function used by de Bruijne and Nielsen [7] for particle filtering is that we use the posterior probabilities $p_i(b|g_i)$ instead of the likelihoods $p_i(g_i|b)$. Thus, our fitness function directly expresses the statistical evidence for the individual shape in the image.

The selection process is implemented using a random sampling in which each individual s gets a chance of reproduction proportional to its fitness w_s. Subsequently, all drawn individuals are mutated with the current standard deviation

σ_t. For the next iteration $t+1$, the standard deviation is reduced using $\sigma_{t+1} = 0.95\,\sigma_t$. This corresponds to a reduction in step size of the optimizer and enables us to use a relatively large σ_0 to conduct an exhaustive search in the beginning and still obtain a stable convergence towards the end. After a fixed number of x iterations, we consider the optimization as converged. The individual reaching the maximum fitness during the evolution is the final solution.

3.2 Landmark Reduction

In our segmentation scheme, the evolutionary algorithm is run to find a rough initialization in a strongly down-sampled version of the image. For this purpose, a simplified version of the SSM (i.e. with fewer landmarks) is equally suitable, but considerably faster during the search. The process of choosing the best landmarks for the reduced model is essentially a mesh simplification problem: While we prefer to eliminate landmarks with poorly performing appearance models (details follow in the next section), we have to assure that the reduced landmark set still covers all parts of the surface. Consequently, we do not delete a landmark if the resulting gap would be larger than a certain geodesic radius r, which is estimated by the number of traversed edges in the SSM. For $r = 1$, this means there must exist at least one ”surviving” landmark in the direct neighborhood of each deleted one.

3.3 Benchmarking Local Appearance Models

To compare the performance of different appearance models on a per-landmark basis, we propose the following method: In all training images, the probability $p(b|g)$ is estimated at the true boundary position and at $2K$ positions shifted along the respective normal vector. To simulate realistic search conditions and to avoid testing on the actual training data, the true boundary position g_0 is randomized around each landmark i with a uniform distribution in the polygon determined by the direct neighbors of i. In addition, the employed normal vectors are randomized with a standard deviation of $\sigma = 10$ degrees. This way, a number of $R = 30$ tests is run for each landmark in every image. The performance of one test r is evaluated by a weighted sum of differences between the boundary probability at the true position g_{r0} and the ones at shifted positions g_{rk}:

$$f_{ir} = \left(\sum_{k=-K}^{K} |k|^d \left(p_i(b|g_{r0}) - p_i(b|g_{rk}) \right) \right) / \left(\sum_{k=-K}^{K} |k|^d \right) \tag{4}$$

where d determines the influence of the shifting distance ($d = 1$ in our case). By averaging the results over all tests, a performance index $f_i \in [-1..1]$ is estimated for each landmark. The obtained benchmark values are used for the mesh simplification from Sect. 3.2.

4 Deformable Model

The deformable model used for the final segmentation is defined as a triangulated mesh $M = (V, E)$ with vertices $p, q \in V$ and edges $[p, q] \in E$. M has the same topology as the statistical shape model, i.e. for each vertex p in the mesh, there is a corresponding vertex $\tilde{p}$ in the SSM. The evolution of the deformable model is controlled by the Lagrangian equation of motion: At every vertex p_i, a regularizing internal force $F_{int}(p_i)$ and a data-driven external force $F_{ext}(p_i)$ are applied. In the following, we will present the derivations for both internal and external forces in detail.

4.1 Internal Forces

The internal forces should keep the shape of the deformable model similar to the one of the underlying SSM. We define this similarity by evaluating differences in edge lengths (also used by Weese et al. in [4]) and differences in angles between neighboring faces. This approach is based on the concepts of tension and rigidity which are also used to define the internal energy of a snake [2].

To implement the tension forces, every edge $[p, q]$ is modeled as a linear spring with the neutral length $|\tilde{p} - \tilde{q}|$, which is the length of $[p, q]$ in the template. Consequently, the tension force on a vertex p in the direction towards q is:

$$F_T(p, q) = \alpha \left(1 - \frac{|\tilde{p} - \tilde{q}|}{|p - q|}\right)(p - q) \tag{5}$$

where α defines the strength of the tension force and is constant for all $[p, q] \in E$. Thus, the total tension force for a vertex is the sum over the forces along all of its edges:

$$F_T(p) = \sum_{[p,q] \in E} F_T(p, q) \tag{6}$$

For the description of rigidity forces, we use the following definitions: For every edge $[p_1, p_2] \in E$, the adjacent triangles $[p_1, p_2, q_1]$ and $[p_2, p_1, q_2]$ form an angle θ that strives towards the corresponding angle $\tilde{\theta}$ in the SSM (see Fig. 2). We call q_1, q_2 the outer vertices of $[p_1, p_2]$; both together form the set $V_O([p_1, p_2])$. The rigidity force for an outer vertex $q \in V_O([p_1, p_2])$ is defined as

$$F_R(q, [p_1, p_2]) = T(q, [p_1, p_2], \beta\delta)) - q \tag{7}$$

where $T(q, [p_1, p_2], \varphi)$ is a rotation of point q around the edge $[p_1, p_2]$ by φ degrees, and β is the strength of the rigidity force. To define δ for both outer vertices, we have to consider the case of a constellation where the distance d_1 between $[p_1, p_2]$ and q_1 is different from d_2. To balance forces on both outer vertices, the triangle with the shorter distance d has to rotate more than the other one:

$$\delta = \frac{d_{op}}{d_1 + d_2} \left(\theta - \bar{\theta}\right) \tag{8}$$

where d_{op} is the distance from $[p_1, p_2]$ to the opposing outer vertex (i.e. d_2 when calculating δ for q_1 and vice versa). An important point to take care of is that the internal forces may not alter the overall position of the deformable surface, i.e. all internal forces must sum up to zero. Therefore, the forces on q_1 and q_2

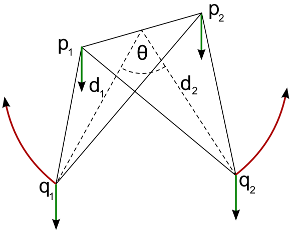

Fig. 2. Two adjacent triangles form the angle θ. Internal rigidity forces striving for a larger angle directly affect the outer vertices q_1 and q_2 (large, curved arrows), but also the edge vertices p_1 and p_2 to maintain the equilibrium of the constellation.

have to be neutralized. The neutralizing force acts equally on all four vertices of a constellation and is defined as

$$F_N([p_1, p_2]) = -\frac{1}{4}(F_R(q_1, [p_1, p_2]) + F_R(q_2, [p_1, p_2])) \tag{9}$$

Overall, this results in a total rigidity force of

$$F_R(p) = \sum_{[p_1,p_2]\in E} \begin{cases} F_R(p, [p_1, p_2]) + F_N([p_1, p_2]) & \text{if } p \in V_O([p_1, p_2]) \\ F_N([p_1, p_2]) & \text{if } p = p_1 \vee p = p_2 \\ \mathbf{0} & \text{else} \end{cases} \tag{10}$$

Finally, the internal force for a given vertex is the sum of tension and rigidity force:

$$F_{int}(p) = F_T(p) + F_R(p) \tag{11}$$

4.2 External Forces

The external forces drive the deformable surface towards the best fit to the data. As in the Active Shape Model search [1], the goodness of fit is evaluated using the local appearance models for all $p \in V$; once at p itself and additional at K positions on each side of the surface. This procedure leads to $2K + 1$ probes for each vertex, enumerated as $k \in [-K..K]$. Defining the optimum probe position as $s(p)$, a linear spring force drives the vertex in the corresponding direction:

$$F_{ext}(p) = \gamma(s(p) - p) \tag{12}$$

where γ is the strength of the external forces. Usually, $s(p)$ is determined independently for each vertex p by picking the maximum fitness value, a procedure generating a considerable amount of outliers. Behiels et al. have shown that a smoothness constraint on the allowable changes significantly increases the robustness of deformation in SSMs [18], albeit only for the 2D case. For 3D, no efficient algorithm to adhere to such constraints was known until recently, when Li et al. presented a graph-based approach to detect optimal surfaces [19]. In the following section, we will briefly summarize this algorithm and show how it integrates with our local appearance models.

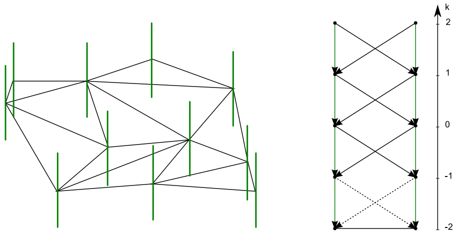

Fig. 3. Optimal surface detection for closed triangle meshes: On the left, a section of the mesh is displayed; the different probe positions for each vertex are displayed by a vertical line. The right side illustrates how two adjacent probe lines are represented in the graph (here for $K = 2$ and $\Delta = 1$). All edges have infinite weight, the dotted connections are optional.

4.3 Optimal Surface Detection

To employ the optimal surface detection algorithm from [19], the first step is to build a directed graph $G = (N, D)$ from the mesh M and its surroundings: For each vertex $p \in V$, $2K + 1$ nodes (corresponding to the probe positions) are inserted into the graph as $n(p, k)$. These nodes are connected by directed edges of infinite weight from $n(p, k)$ to $n(p, k - 1)$ for $k > -K$ and form a so-called column. Next, directed edges between neighboring columns $col(p_1)$ and $col(p_2)$ are added from $n(p_1, k)$ to $n(p_2, \max(-K, k - \Delta)$, where Δ is the smoothness constraint and specifies how many steps adjacent probes may shift against each other during deformation. $col(p_1)$ and $col(p_2)$ count as adjacent if $[p_1, p_2] \in E$. A simplified visualization of this graph structure is given in Fig. 3.

The next step is to assign a weight $w(p, k)$ to each $n(p, k) \in N$. Assuming that the costs for each probe are stored in $c(p, k)$, weights are computed as

$$w(p,k) = \begin{cases} c(p,k) & \text{for } k = -K \\ c(p,k) - c(p,k-1) & \text{for } k > -K \end{cases} \tag{13}$$

Since the algorithm finds the solution that minimizes the sum of costs, we have to transform the probabilities $p(b|g)$ from Eq. 2 accordingly. As in the weight calculation for the evolutionary algorithm (3), we want to maximize the posterior probability of the shape, i.e. the product of all boundary probabilities. Thus, we define the costs as

$$c(p,k) = -\log p_p(b|g_{pk}) \tag{14}$$

Subsequently, an additional source node n_s and a sink node n_t are added to the graph. All previously inserted nodes are connected to these two in the following way: Every node with $w(p, k) \geq 0$ is connected to n_t by a directed edge of weight $w(p, k)$, and n_s is connected to every node with $w(p, k) < 0$ by a directed edge of weight $-w(p, k)$. Employing an s-t cut algorithm like the one presented in [20], the graph can now be divided into a source set S with $n_s \in S$ and sink set T with $n_t \in T$. The optimal displacements $s(p)$ are given by the largest k with $n(p, k) \in S$.

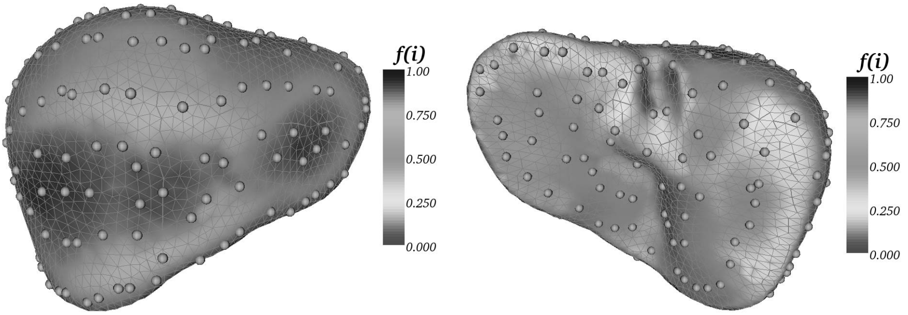

Fig. 4. Reduced set of landmarks on the shape model of the liver (front and back view): The performance f_i of the local appearance models is color-coded over the entire surface. The full set of landmarks is displayed using a triangular grid, the reduced set is shown as small spheres. The maximum gap-radius r is 2. It is clearly visible how landmarks evade the low-performance sections of the surface by grouping around them.

5 Experiments and Results

We chose the task of liver segmentation as an example to validate our approach: An automatic segmentation of this organ without prior shape information is prone to fail due to the low contrast to neighboring structures. However, the large amount of anatomical variation makes it particularly challenging to model and detect using classical SSMs.

5.1 Image Data and Model Generation

All used images are contrast-enhanced CT volumes of the abdomen with an in-plane resolution of 512x512 pixels and 60 to 130 slices (spacing ∼0.7x0.7x3mm). In most cases the anatomy is pathologic, i.e. interspersed with tumors. Additionally, the data was acquired using different protocols, which should allow a reliable prediction of the clinical performance of the presented segmentation scheme.

Out of 86 volumes, 32 were selected as training data for the statistical model, which was built using 2562 equally distributed landmarks. The remaining 54 volumes were used for evaluation. Appearance models as described in Sect. 2.2 were generated for five different resolutions R_0 (original resolution) to R_4 (four times down-sampled), featuring a profile-length of 7 pixels and a pixel-spacing of 1mm in R_0. The k-Means algorithm [21] was employed to group all landmarks into 22–42 clusters (depending on the resolution) and to improve the accuracy of the kNN-classifiers. After that, the performance of the appearance models for the lowest resolution R_4 was benchmarked with the method from Sect. 3.3. The obtained results were used to create a reduced set of 204 landmarks for the evolutionary algorithm (see Fig. 4).

5.2 Segmentation Workflow and Parameter Values

The segmentation of one image consisted of three major steps: To find the initial position and shape parameters of the SSM, the evolutionary algorithm was

Table 1. Parameter values used in the different phases of the deformable model segmentation: Convergence criteria are either the maximum vertex movement per iteration d_{max} or the number of iterations I

Resolution	Convergence criterion	α	β	γ	Δ
3	$d_{max} < 0.3$mm	0.125	0.25	0.01	1
2	$d_{max} < 0.4$mm	0.125	0.25	0.02	2
2	$d_{max} < 1.0$mm	0.125	0.25	0.05	2
2	$I = 50$	0.125	0.25	0.10	2
1	$I = 50$	0.125	0.25	0.10	2
0	$I = 20$	0.125	0.25	0.10	2

run with a population of 1000 individuals over 40 iterations. Fitness was evaluated on the 204 reduced landmarks in R_4. The initial standard deviation for mutation σ_0 was set to 0.4, resulting in a final σ_{40} around 0.05. After that, a normal ASM search (i.e. without allowing extra deformation) was conducted to improve the solution if possible. It was first run in R_4 until the maximum vertex movement D_{max} was less than 4mm, then in R_3 until $D_{max} < 2mm$. Finally, the deformable model search was started in R_3 down to R_0. As in the previous steps, the underlying SSM used 10 modes of variation. To improve the runtime of the algorithm, the external forces were updated every 10 iterations. For an overview over the used parameter values, see Table 1.

5.3 Results

All 54 generated segmentations were compared to manual delineations by radiological experts using three different error metrics: Symmetric average surface distance D_{avg}, symmetric RMS surface distance D_{RMS} and volumetric error V_D. V_D is based on the Dice coefficient and is calculated as $V_D = 1-(2\cdot|A\cap B|)/(|A|+|B|)$ for two sets of voxels A and B. The results achieved by the presented deformable model, a standard ASM search with 30 modes of variation after manual initialization and two other approaches from literature are summarized in Table 2. The three worst segmentations (V_D>10% for the deformable model) were treated as outliers and omitted from the statistics. For a visual performance assessment, the image featuring the median average surface distance is displayed in Fig. 5. The computation time on a 3GHz desktop PC is approximately 10 minutes per image, of which the automatic initialization requires the major part of 6 minutes.

6 Conclusion

We have presented an automated segmentation procedure combining techniques from statistical shape models and deformable surfaces. Main contributions are the initialization of a 3D SSM using an evolutionary algorithm on a simplified model and a robust deformation scheme by means of a constrained optimal surface detection. Fitting costs for both searches are estimated by the posterior

Table 2. Segmentation error in comparison with a standard ASM and numbers from previous work (customized ASM [22] and deformable simplex mesh [5]), given as $\mu \pm \sigma$

Segmentation method	$D_{avg}[mm]$	$D_{RMS}[mm]$	$V_D[\%]$
Deformable model	1.6 ± 0.5	3.3 ± 1.2	5.1 ± 1.4
Active Shape Model	2.9 ± 1.1	5.2 ± 2.3	8.9 ± 2.4
Lamecker et al. [22]	2.3 ± 0.3	3.1 ± 0.5	7.0 ± 1.8
Soler et al. [5]	2±?	*n.a.*	*n.a.*

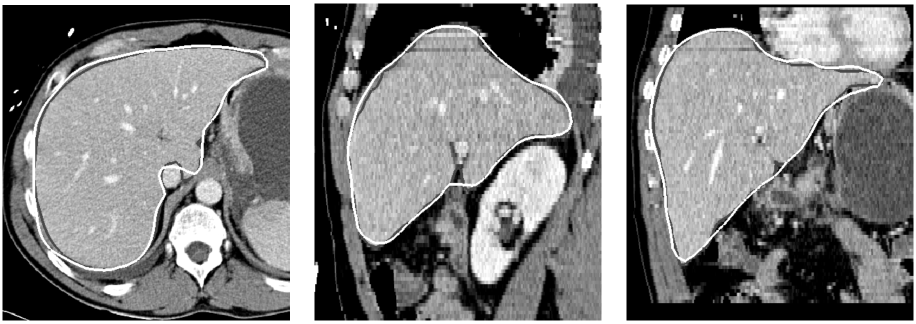

Fig. 5. Transversal, sagittal and coronal slices for the image with median average surface error. The result of the deformable model is displayed in white, the manually traced reference contour in dark gray.

probability for the model given the data. The obtained results on liver CT images are excellent and encourage us to employ the approach for other soft tissue objects in the near future.

Acknowledgements

This work was supported by the German Research Foundation DFG under grant WO 1218/2-1.

References

1. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models – their train- ing and application. Computer Vision and Image Understanding 61(1), 38–59 (1995)
2. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision 1(4), 321–331 (1988)
3. Shen, D., Davatzikos, C.: An adaptive-focus deformable model using statistical and geometric information. IEEE Trans. Pattern Analysis and Machine Intelligence 22(8), 906–913 (2000)
4. Weese, J., Kaus, M., Lorenz, C., Lobregt, S., Truyen, R., Pekar, V.: Shape constrained deformable models for 3D medical image segmentation. In: Insana, M.F., Leahy, R.M. (eds.) IPMI 2001. LNCS, vol. 2082, pp. 380–387. Springer, Heidelberg (2001)

5. Soler, L., Delingette, H., Malandain, G., Montagnat, J., Ayache, N., et al.: Fully automatic anatomical, pathological, and functional segmentation from ct scans for hepatic surgery. In: Proc. SPIE Medical Imaging, pp. 246–255 (2000)
6. Hill, A., Taylor, C.J., Cootes, T.F.: Object recognition by flexible template matching using genetic algorithms. In: Sandini, G. (ed.) ECCV 1992. LNCS, vol. 588, pp. 852–856. Springer, Heidelberg (1992)
7. de Bruijne, M., Nielsen, M.: Shape particle filtering for image segmentation. In: Barillot, C., Haynor, D.R., Hellier, P. (eds.) MICCAI 2004. LNCS, vol. 3216, pp. 168–175. Springer, Heidelberg (2004)
8. Subsol, G., Thirion, J.P., Ayache, N.: A scheme for automatically building three-dimensional morphometric anatomical atlases: application to a skull atlas. Medical Image Analysis 2(1), 37–60 (1998)
9. Frangi, A.F., Rueckert, D., Schnabel, J.A., Niessen, W.J.: Automatic construction of multiple-object three-dimensional statistical shape models: application to cardiac modeling. IEEE Trans. Medical Imaging 21(9), 1151–1166 (2002)
10. Kaus, M.R., Pekar, V., Lorenz, C., Truyen, R., Lobregt, S., Weese, J.: Automated 3-D PDM construction from segmented images using deformable models. IEEE Trans. Medical Imaging 22(8), 1005–1013 (2003)
11. Davies, R.H., Twining, C.J., Cootes, T.F., Waterton, J.C., Taylor, C.J.: 3D statistical shape models using direct optimisation of description length. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2352, pp. 3–20. Springer, Heidelberg (2002)
12. Heimann, T., Wolf, I., Meinzer, H.P.: Optimal landmark distributions for statistical shape model construction. In: Proc. SPIE Medical Imaging: Image Processing. vol. 6144, pp. 518–528 (2006)
13. de Bruijne, M., van Ginneken, B., Viergever, M.A., Niessen, W.J.: Adapting active shape models for 3D segmentation of tubular structures in medical images. In: Taylor, C.J., Noble, J.A. (eds.) IPMI 2003. LNCS, vol. 2732, pp. 136–147. Springer, Heidelberg (2003)
14. Kittler, J., Alkoot, F.M.: Moderating k-NN classifiers. Pattern Analysis & Applications 5(3), 326–332 (2002)
15. Fogel, L.J., Owens, A.J., Walsh, M.J.: Artificial Intelligence through Simulated Evolution. John Wiley, New York (1966)
16. Schwefel, H.P.: Evolution and Optimum Seeking. John Wiley & Sons, Inc, New York (1995)
17. Holland, J.H.: Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor (1975)
18. Behiels, G., Maes, F., Vandermeulen, D., Suetens, P.: Evaluation of image features and search strategies for segmentation of bone structures in radiographs using active shape models. Medical Image Analysis 6(1), 47–62 (2002)
19. Li, K., Millington, S., Wu, X., Chen, D.Z., Sonka, M.: Simultaneous segmentation of multiple closed surfaces using optimal graph searching. In: Christensen, G.E., Sonka, M. (eds.) IPMI 2005. LNCS, vol. 3565, pp. 406–417. Springer, Heidelberg (2005)
20. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Analysis and Machine Intelligence 26(9), 1124–1137 (2004)
21. Hartigan, J.A., Wong, M.A.: A K-means clustering algorithm. Applied Statistics 28, 100–108 (1979)
22. Lamecker, H., Lange, T., Seebass, M.: Segmentation of the liver using a 3D statistical shape model. Technical report, Zuse Institute, Berlin, Germany (2004)