# Image Segmentation Based on Shape Space Modeling

Daehee Kim and Yo-Sung Ho

Kwangju Institute of Science and Technology
1 Oryong-dong Puk-gu, Kwangju, 500-712, Korea
{kimdh, hoyo}@kjist.ac.kr

**Abstract.** In this paper, we propose a new image segmentation method based on the active contour. If we define a shape space as a set of all possible variations from the initial curve and we assume that the shape space is linear, it can be decomposed into the column space and the left null space of the shape matrix. In the proposed method, the shape space vector in the column space describes changes from the initial curve to the imaginary feature curve, and a dynamic graph search algorithm describes the detailed shape of the object in the left null space. Since we employ the shape matrix and the SUSAN operator to outline object boundaries, the proposed algorithm can ignore unwanted feature points generated by low-level image processing operations and is therefore applicable to images of the complex background. We can also compensate for limitations of the shape matrix with the dynamic graph search algorithm.

## 1 Introduction

The MPEG-4 visual coding standard [1] enables content-based functionalities by introducing the concept of the video object plane (VOP). A VOP is defined as a coding unit in the MPEG-4 natural visual coding, determined by the shape of the predefined video object, at a certain frame over the entire sequence. However, since it is not easy to define a mathematical model or a similarity measure for extracting video objects adequately, automatic segmentation algorithms cannot provide satisfactory segmentation results over various image sequences. If the user can define VOPs in the first frame in a user-assisted manner, we may obtain better segmentation results in the subsequent picture frames. Therefore, the user-assisted segmentation approach is more practical in generating VOPs of moving objects. In this paper, we propose a new image segmentation method based on an active contour algorithm to define objects in a given image.

In the active contour or snake algorithm, we try to find an energy-minimizing curve from the initial curve indicated by the user. Performance of the active contour algorithm depends mainly on the definition of the energy function and the shape of the active contour is controlled by internal, external and constraint forces. While the internal force is the smoothness constraint on the curve, the external force guides the active contour towards image features. The constraint force allows interactivity in manipulating the active contour. The energy functional $E^*(s)$ is represented as a parametric curve $\mathbf{r}(s)=(x(s), y(s))$, where $s$ is the parameter for the given interval [2].

A functional of the active contour can be defined by

M.H. Shafazand and A M. Tjoa (Eds.): EurAsia-ICT 2002, LNCS 2510, pp. 164–171, 2002.

$$E^*(s) = \int_0^1 E_{snake}(\mathbf{r}(s))ds \tag{1}$$
$$= \int_0^1 [E_{int}(\mathbf{r}(s)) + E_{ext}(\mathbf{r}(s)) + E_{con}(\mathbf{r}(s))]ds$$

where $E_{int}$, $E_{ext}$ and $E_{con}$ represent the internal energy of the contour, the external image force and the constraint force, respectively. The final location of the active contour corresponds to the local minimum of the energy functional.

However, the minimization operation of $E^*(s)$ by variational calculus may cause a problem of numerical instability [3]. Since dynamic programming techniques [3] can provide more stable results, most active contour algorithms adopt dynamic programming. Unfortunately, dynamic programming techniques are extremely slow because of computational complexity of $O(nm^3)$, where $n$ is the number of points to be processed on the active contour and $m$ is the number of points in the search window. Greedy algorithms [4] are sensitive to the distance between adjacent points on the initial contour. Those conventional algorithms are designed for objects in simple and homogeneous backgrounds.

In this paper, we define a shape space as a set of all possible variations from the initial curve. This shape space is divided into two subspaces by the shape matrix, which is used to represent variations of the initial curve with a few parameters. Therefore, the shape space consists of a subspace defined by the shape matrix and the other space cannot be represented by a few parameters. In this paper, we first describe changes from the initial contour with a few parameters and employ a graph search algorithm that compensates for limitations of the shape matrix.

## 2 Shape Space Model

### 2.1 B-Spline Curve

The parametric curve $\mathbf{r}(s)=(x(s), y(s))$ is a particular function of the parameter $s$. A B-spline function $x(s)$ can be constructed as a weighted sum of $N_B$ basis functions $B_n(s)$, where $n=0,\ldots,N_B-1$. If we set the order $d=3$, the curve has a continuous gradient. Therefore, the constructed spline function satisfies the internal energy requirement in its nature.

The spline function can be represented by

$$x(s) = \mathbf{B}(s)^T \mathbf{Q}_x \tag{2}$$

$$\mathbf{B}(s) = [B_0(s), B_1(s), \ldots, B_{N_B-1}(s)]^T \quad \mathbf{Q}_x = [x_0, x_1, \ldots, x_{N_B-1}]^T$$

where $x_n$ is the weight for a basis function $B_n(s)$. Therefore, the parametric curve $\mathbf{r}(s)$ is also represented in the matrix form.

$$\mathbf{r}(s) = \mathbf{U}(s)\mathbf{Q} \tag{3}$$

where

$$\mathbf{U}(s)=\mathbf{I}_2\otimes\mathbf{B}(s)^T=\begin{pmatrix}\mathbf{B}(s)^T & 0\\ 0 & \mathbf{B}(s)^T\end{pmatrix},\ \mathbf{Q}=\begin{pmatrix}\mathbf{Q}_x\\ \mathbf{Q}_y\end{pmatrix} \tag{4}$$

In Eq. (4), $\mathbf{I}$ is a $2\times 2$ identity matrix and the operation of $\otimes$ is the Kronecker product.

### 2.2 Shape Matrix

A shape space is defined by a set of all possible variations from the initial curve. If we assume that the universal shape space is a linear space, it can be composed of the column space $\mathcal{R}(\mathbf{W})$ and the left null space $\mathcal{N}(\mathbf{W}^T)$ of the shape matrix which will be defined in this section. The left null space is orthogonal to the column space [6].

The column space $\mathcal{R}(\mathbf{W})$ is constructed from a set of vectors of dimension $N_x$. It is desirable to restrict the displacement of control points to a lower dimensional shape space if it preserves the frame of the shape. An unconstrained control vector $\mathbf{Q}$ may lead to unstable active contours [7]. A change of the curve in $\mathcal{R}(\mathbf{W})$ is a linear mapping of the shape space vector $\mathbf{X}$ into a control vector $\mathbf{Q}$

$$\mathbf{Q}-\mathbf{Q}_0=\mathbf{WX} \tag{5}$$

where $\mathbf{W}$ is the shape matrix of size $N_Q\times N_x$, and $\mathbf{Q}_0$ is a control vector of the initial curve. The shape space vector $\mathbf{X}$ describes the change of the initial curve.

In this paper, we describe contour changes by the 6-parameter affine model ($N_x=6$). This class can be represented by the initial curve $\mathbf{Q}_0$ and the shape matrix [7]:

$$\mathbf{W}=\begin{pmatrix}1 & 0 & \mathbf{Q}_{x0} & 0 & 0 & \mathbf{Q}_{y0}\\ 0 & 1 & 0 & \mathbf{Q}_{y0} & \mathbf{Q}_{x0} & 0\end{pmatrix} \tag{6}$$

Each column of $\mathbf{W}$ forms a basis vector of the column space $\mathcal{R}(\mathbf{W})$ of the shape matrix $\mathbf{W}$, but it is not necessary to be orthogonal to the other vectors.

### 2.3 Projection onto Column Space of Shape Matrix

In order to find the boundaries of video objects, we need to define a distortion measure that can be expressed by the curve norm of shape differences between the estimated curve $\mathbf{r}(s)$ and the desired curve $\mathbf{r}_d(s)$.

$$\left\|\mathbf{r}(f(s))-\mathbf{r}_d(s)\right\|^2=\frac{1}{L}\int_0^L\left|\mathbf{r}(f(s))-\mathbf{r}_d(s)\right|^2 ds \tag{7}$$

where $L$ is the length of the interval of $s$ and $f(s)$ is the adjustment function which is necessary to match the corresponding points of the two curves. If two curves are very similar to each other, $\|\mathbf{r}(f(s))-\mathbf{r}_d(s)\|$ can be replaced by $[\mathbf{r}(s)-\mathbf{r}_d(s)]\cdot\mathbf{n}(s)$ [7]. If the integral of Eq. (7) is approximated by a summation and the normal vector $\mathbf{n}(s)$ is used to measure the shape difference, our minimization criterion becomes

$$\| \mathbf{r} - \mathbf{r}_d \|^2 \approx \frac{1}{N} \sum_{i=1}^{N} [(\mathbf{r}_d(s_i) - \mathbf{r}(s_i)) \cdot \mathbf{n}(s_i)]^2 \tag{8}$$

where $N$ is the number of regularly-spaced points in the interval of $s$. Eq. (8) can be rewritten in terms of the initial estimated curve $\mathbf{r}_0$ and the shape space vector $\mathbf{X}$.

$$\| \mathbf{r} - \mathbf{r}_d \|^2 \approx \frac{1}{N} \sum_{i=1}^{N} [(\mathbf{r}_d(s_i) - \mathbf{r}_0(s_i)) \cdot \mathbf{n}(s_i) - \mathbf{n}^T \mathbf{U}(s_i) \mathbf{W} (\mathbf{X} - \mathbf{X}_0)]^2 \tag{9}$$

In order to minimize $\|\mathbf{r}(s) - \mathbf{r}_d(s)\|^2$ in Eq. (9) and to find the estimated shape space vector $\mathbf{X}^*$, we employ the least square approach in this paper. The minimal $\mathbf{X}^*$ can be estimated by setting $\partial \|\mathbf{r} - \mathbf{r}_d\|^2 / \partial \mathbf{X} = 0$, which leads to

$$\mathbf{X}^* = \left( \sum_{i=1}^{N} \rho_i \mathbf{W}^T \mathbf{U}^T \mathbf{n} \mathbf{n}^T \mathbf{U} \mathbf{W} \right)^{-1} \left( \sum_{j=1}^{N} \rho_i \mathbf{W} \mathbf{U}^T \mathbf{n} (\mathbf{r}_d - \mathbf{r}_0)^T \mathbf{n} \right) \tag{10}$$

where $\rho_i$ is a weighting factor instead of $1/N$ to compensate for non-uniform sampling. Once we find $\mathbf{X}^*$, we can obtain the optimal control vector $\mathbf{Q}$ by Eq. (5). For more accurate results, we can repeat the same procedure several times by setting the previously fitted curve $\mathbf{r}(s)$ as the initial curve $\mathbf{r}_0(s)$.

Mathematically, $\mathbf{X}^*$, which describes the change from the initial curve, is the projection onto $\mathcal{R}(\mathbf{W})$. The column space of $\mathbf{W}$ is too restrictive for an active contour that tries to model arbitrary shape deformations. In order to address this problem, we consider the left null space $\mathcal{N}(\mathbf{W}^T)$ in Section 3.

### 2. 4 Estimate of Desired Curve

Since the aim of the active contour algorithm is to fit the initial curve to image features, $\mathbf{r}_d(s)$ of Eq. (7) should be estimated from the given image. In this paper, we employ the smallest univalue segment assimilating nucleus (SUSAN) operator [8] to extract image features. Although the SUSAN edge detector requires lower computational complexity than morphological gradient tools or the Canny edge detector, it works well for images of complex background.

The SUSAN edge detector consists of three steps. In the first step, we place the nucleus of a circular mask around a pixel. In the second step, we calculate the number of pixels that have similar brightness to the nucleus within the circular mask. This number is defined as the area of the univalue segment assimilating nucleus (USAN). In the third step, we subtract the size of USAN from the geometric threshold to produce an image of edge strength.

After the SUSAN edge detection, we utilize the edge image to extract the imaginary feature curve $\mathbf{r}_d(s)$. Since we try to fit the initial curve to image features by the active contour algorithm, we need to find feature points in the search region which includes the neighboring area of the initial curve. Features should be detected by scanning along each of the sampled normal lines to the initial curve, because of the

approximation in Eq. (8). Therefore, if normal lines are constructed at points along $\mathbf{r}(s)$, we obtain a sequence of $\mathbf{r}_d(s_i)$ along the imaginary feature curve $\mathbf{r}_d(s)$.

## 3 Left Null Space of Shape Matrix

In Section 2, we have projected the change of the initial contour onto $\mathcal{R}(\mathbf{W})$. Here, we can ignore effects of unwanted feature points generated by low-level image processing operators and describe the overall change of the contour. However, it is not easy to describe detailed changes of the contour because the column space of the shape matrix consists of only six basis vectors.

In this section, we incorporate the change of the contour in $\mathcal{N}(\mathbf{W}^T)$ to compensate for the fitted contour and obtain the final object boundaries. However, we do not know the actual dimension of the shape space because we cannot describe the object in a finite-dimensional space due to subjectivity of the object definition. Thus, we do not employ general linear algebra approaches to solve problems of the left null space. We consider that feature points on the fitted contour obtained in Section 2 are represented well in $\mathcal{R}(\mathbf{W})$ and other feature points on the fitted contour are distorted by the projection operation onto $\mathcal{R}(\mathbf{W})$ due to the absence of the information about $\mathcal{N}(\mathbf{W}^T)$. Here, we define a seed point sequence as a sequence of points that are represented well in $\mathcal{R}(\mathbf{W})$ without the information about $\mathcal{N}(\mathbf{W}^T)$. In order to estimate the final object boundaries, we modify the distorted points of the fitted contour using the seed point sequence.

Boundary definition can be formulated as a graph search problem where the key idea is to form an image by a weighted graph where pixels represent nodes with weighted directed edges connecting each pixel and its eight adjacent neighboring pixels [9]. We set the first point and the second point of the seed point sequence to be the start point and the end point, respectively. Once we find the optimal path, we replace the start point with the end point and set the third point to a new end point. This procedure is performed repeatedly until we arrive the last point of the seed point sequence. The optimal path is defined as the minimum cumulative cost path from the start point to the goal point.

Since the minimum cost path should correspond to boundaries of the object, pixels of strong edge features should have low local costs and vice versa. In order to reflect various edge features to local costs, we define the local cost in terms of the static and dynamic costs and the total cost as the weighted sum of the static and dynamic costs.

### 3.1 Static Cost

First, we define the static local cost function as the weighted sum of cost functions derived from Laplacian zero-crossings and gradient magnitudes. We use multiple kernel widths because multiple kernels provide good performance psychologically [10]. Each kernel is generated from the 2-D Gaussian distribution with a different standard deviation. We define cost functions so that strong feature points produce low costs and vice versa.

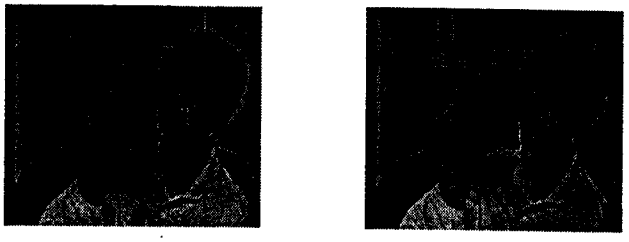
### 3.2 Dynamic Cost

Some parts of the object boundary may have weak gradient magnitudes relative to nearby strong gradient edges. If the nearby strong edge has a relatively lower cost, the optimal path moves to the strong edge rather than the desired edge. Thus, the gradient magnitude cost function should be modified dynamically to resolve this problem.

We can define the dynamic cost function from the segmentation results along the path from the previous start point to the previous end point of the seed point sequence. In order to adapt gradual changes in edge characteristics, we can update the dynamic cost function as the interval of seed points is changed. The dynamic cost function can be obtained from the histogram of costs in the previous optimal path.

## 4 Experimental Results

In order to evaluate performance of the proposed algorithm, we run computer simulations on MPEG-4 test images of the CIF format. Figure 1 demonstrates the effects of the dynamic cost functions.



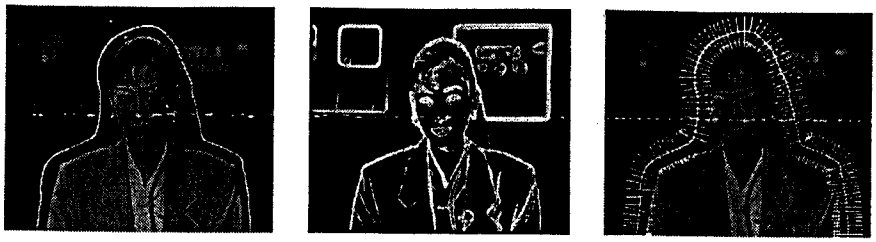
(a) Static Cost (b) Dynamic Cost

**Fig. 1.** Effect of Dynamic Costs

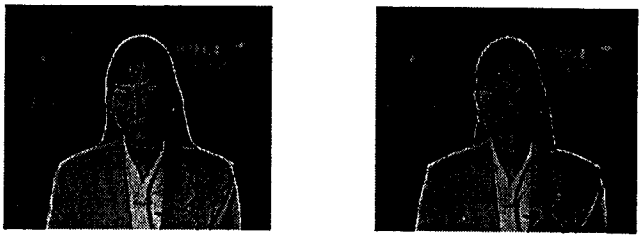
As we mentioned earlier, when a section of the desired object boundary has weak gradient magnitudes relative to strong gradient magnitudes in the neighborhood of the object boundary, the section of the optimal path is set on the strong magnitudes, as shown in Figure 1(a). The desired boundary is the mother's cheek. However, since its part is so close to her high contrast lip, the optimal path is set on her lip. In order to avoid this problem, we consider the dynamic cost defined in Section 3.2. Figure 1(b) demonstrates how a segment of the optimal path latches onto the edge that is similar to the previous segment. In Figure 1(b), the white line of her shoulder is obtained only by static costs, but the white line of her cheek is obtained by dynamic costs derived from her shoulder line.

Figure 2 shows simulation results of the proposed algorithm applied to AKIYO image. Figure 2(a) shows the initial contour provided by a user-pointing device. Image features obtained by the SUSAN operator are displayed in Figure 2(b). The search region displayed in Figure 2(c) is formed by sweeping normal vectors along the initial curve. In Figure 2(c), points on white lines are candidates of $\mathbf{r}_d(s_i)$ corresponding to $\mathbf{r}(s_i)$. Figure 2(d) is the fitted curve by the repeated least square method. Figure 2(e) is the final segmentation result that is compensated with the graph search algorithm. In Figure 2(d), the fitted curve is misaligned with AKIYO's ears and parts of her shoulder connected to her arm due to the smoothness property of

the 6-parameter approximation and the B-spline curve. The final curve in Figure 2(e) tracks the object boundary in more detail.



(a) Initial Contour (b) Feature Map (c) Search Region

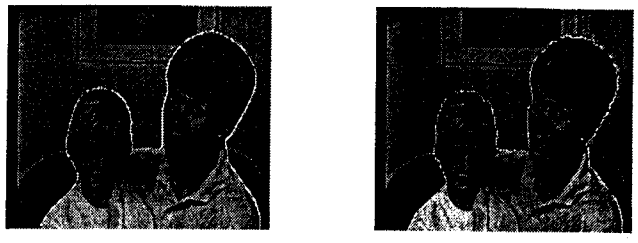


(d) Fitted Curve (e) Final Result

**Fig. 2.** Results for AKIYO Image

Figure 3 shows segmentation results for MOTHER AND DAUGHTER image that has more complex shapes than AKIYO image. Figure 3(e) demonstrates a more accurate shape of the object boundary than Figure 3(d), particularly in the area of the mother and daughter's hair. In general, since conventional active contour algorithms are mainly designed for images with homogeneous backgrounds, they may not work well for objects having complex backgrounds.

(a) Initial Contour (b) Feature Map (c) Search Region



(d) Fitted Curve (e) Final Result

**Fig. 3.** Results for MOTHER AND DAUGHTER Image

## 5 Conclusions

In this paper, we have proposed a new user-assisted active contour algorithm to extract objects from a given image. Since we employ the shape matrix and the SUSAN operator to outline object boundaries, the proposed algorithm can ignore some outliers or unwanted feature points generated by low-level image processing operations and is applicable to images with complex backgrounds. These outliers make it difficult for conventional active contour algorithms to find object boundaries in non-homogeneous backgrounds. We also use a dynamic graph search algorithm to overcome limitations of the 6-parameter affine model by describing changes of the curve in the left null space of the shape matrix. The dynamic graph search algorithm can describe the detailed shape of the object, which cannot be represented by the shape matrix.

**Acknowledgement.**
This work was supported in part by the Korea Science and Engineering Foundation (KOSEF) through the Ultra-Fast Fiber-Optic Networks (UFON) Research Center at Kwangju Institute of Science and Technology (K-JIST), and in part by the Ministry of Education (MOE) through the Brain Korea 21 (BK21) project.

## References

1. ISO/IEC FDIS 14496-2: Information Technology - Generic Coding of Audio-Visual Objects, Part 2: Visual. ISO/IEC JTC1/SC29/WG11 (1998)
2. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active Contour models. First International Conference on Computer Vision (1987) 259-269
3. Amimi, A., Weymouth, T., Jain, R.C.: Using Dynamic Programming for Solving Variational Problems in Vision. IEEE. Trans. Patt. Anal. Mach. Intel., Vol. 12, No. 9 (1990) 855-867
4. Williams, D. J., Shah, M.: A Fast Algorithm for Active Contours and Curvature Estimation. CVGIP:Image Understanding, Vol. 55, No. 1 (1992) 14-26
5. Foley, J. D., Dam, A., Feiner, S. K., Hughes, J. F., Phillips, R. L.: Introduction to Computer Graphics. Addison-Wesley, New York (1995)
6. Strang, G.: Linear Algebra and Its Applications. 3rd edn. Harcourt Brace Jovanovich (1988)
7. Blake, A., Isard, M.: Active Contours. Springer-Verlag, Berlin Heidelberg New York (1998)
8. Smith, S.M, Brady, J.M.: SUSAN – A New Approach to Low Level Image Processing. Int. Journal of Computer Vision, 23(1) (1997) 45-78
9. Mortensen, E.N., Barrett, W.A.: Interactive Segmentation with Intelligent Scissors. Graphical Models and Image Processing (1998) 349-384
10. Marr, D., Hildreth, E,: Theory of Edge Detection. Proc. R. Soc. Lond. B 270 (1980) 187-217
11. Kim, D., Ho, Y.S.: A User-Assisted Segmentation Algorithm Using B-Spline Curves. SPIE Visual Communications and Image Processing (2001) 734-744