Boundary Extraction Using Statistical Edge and Curvature Model

Hae Chul Choi, J. S. Lee, H. C. Shin, J. H. Cho, S.D. Kim
KAIST Dept. of Electrical Engneering
chc@sdvision.kaist.ac.kr

Abstract

We propose an algorithm for extracting the boundary of an object. In order to take full advantage of global shape, our approach uses global shape parameters derived from Point Distribution Model (PDM). Unlike PDM, the proposed method models global shape using curvature as well as edge. The objective function of applying the shape model is formulated using Bayesian rule. We can extract the boundaries of an object by evaluating iteratively the solution maximizing the objective function. Experimental results show that the proposed method can reduce computation cost than the PDM and it is robust to noise, pose variation, and some occlusion.

I. Introduction

In this paper, we try to solve the problem of extracting the boundary of an object. Because the shape described by the boundary is a powerful property, it is one of the most significant problems in image processing, computer vision, and pattern recognition. Up to the present, two methodologies, model-free and model-based, have been proposed in segmentation approach [1]. Where the model is the technique representing a shape of an object with some parameters. Model-free algorithms [2][3] based on clustering, morphological filtering [4], and a watershed method [5], use local properties such as gray level, texture, and color. These local features can be useful, but these features are not as expressive as global shape and are more sensitive to noise and occlusion by other objects. The model-based methods can generally find more accurate boundary of an object than model-free methods but request more computation cost. Kass et al. [6] introduced deformable contours to model complex shape. With appropriately designing the energy function, their work performs well even if the boundary is deformed a little. Wang and Staib [7] used a point distribution model (PDM) [8] derived by the principal component analysis and employed the statistical shape distribution acquired from training sets. In case that enough training images describing the shape of an object are given, this method is relatively robust to noise and initialization of the parameters than other model-based methods. In order to take full advantage of shape, our approach uses global shape information based on the point distribution model.

The proposed method models global shape using curvature as well as edge of training images and it formulates objective function to find unknown parameters representing boundaries of an object. The objective function is derived analytically using Bayesian rule. Our method can extract more accurate boundary and request less computation cost than PDM does, since it can represents more detailed shape and it finds the true position of high curvature points rapidly.

II. Statistical Edge and Curvature Model

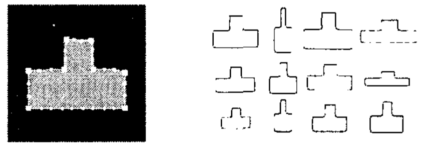
In order to employ global shape statistics obtained from training images, we have to model the shape with parameters. The edge and the curvature of the boundaries derived from training images are parameterized using the point distribution model and the proposed statistical curvature model respectively.

1. Point distribution model

The point distribution model (PDM) [8] is useful for describing shape, but a complex rigid model cannot be easily described. The PDM represents each shape of training images as shown in Fig. 1(a). For setting labeled points, we extract critical points that have high curvature on the boundary. Equally spaced points are interpolated between the critical points. Given M aligned training shapes as Fig. 1(b), each of these is described by a position vector. The position vector is $\mathbf{L}_i=[x_i(1), y_i(1), x_i(2), y_i(2),\ldots, x_i(N), y_i(N)]^T$ $(i=1,\ldots,M)$, where N is the number of total labeled point in a training image. We calculate the mean shape, $\bar{\mathbf{L}}$, and the covariance about the mean. According to the principal component analysis, any shape in the training set can be approximated using the mean shape and a weighted sum of deviations obtained from the eigenvectors of covariance matrix:

$$\mathbf{L} = \bar{\mathbf{L}} + Q\mathbf{b}, \tag{1}$$

where $Q=(\mathbf{q}_1|\mathbf{q}_2|\ldots|\mathbf{q}_t)$ is the matrix of the first t eigenvectors, and $\mathbf{b}=(b_1,b_2,\ldots,b_t)^T$ is a vector of weights which indicate how much variation is exhibited with respect to each of the eigenvectors. This equation allows us to generate plausible shapes that are not part of the training set.



(a) (b)

Fig. 1. Synthetic shape model by PDM (a) synthetic image (128×128) with its 30 labeled points of the boundary, white dots are labeled points and bigger dots are critical labeled points (b) 12 examples of synthetic shapes from a training set

2. Statistical curvature model

Using the same hand-labeled points of the PDM as Fig. 1(b), the statistical curvature model describes the curvature of the shape. Observing edge and curvature with respect to the same deformed template, which is the local contour described by **b**, our method avoids the redundant computation and leads the faster convergence for optimizing the parameter, **b**. In the proposed approach, we define the curvature vector as $\mathbf{R}_i=[r_i(1), r_i(2),\ldots, r_i(N)]^T$ (i=1, 2,..., M) (5). Where $r_i(j)$ is curvature value of the jth labeled point in the ith training image. The mean curvature, $\bar{r}(j)$, and the standard deviation, $\sigma_r(j)$, of the jth labeled point, (j=1,2,...,N), obtained from training images. Then the mean curvature vector is $\overline{\mathbf{R}}=[\bar{r}(1),\bar{r}(2),\ldots,\bar{r}(N)]$. Given these statistics of curvature, we model a curvature image of a training set with noise-corrupted version:

$$R = R_{\mathbf{L}=\bar{\mathbf{L}}+Q\mathbf{b}} + n \tag{2}$$

A curvature image, R, is modeled with one of curvature templates, $R_{\mathbf{L}}$, obtained from the boundary represented by $\bar{\mathbf{L}}+Q\mathbf{b}$ and additive and independent noise, n. Therefore a plausible curvature image that is not part of the training set can be generated applying a curvature operator to a corresponding boundary and adding the noise. For using prior knowledge derived from training images, the noise is characterized with the standard deviation of curvature of training set.

III. Bayesian Objective Function

In order to apply the shape model and curvature statistics to the problem of boundary finding, we derive objective function that is measure of fitting a deformed template, so called a deformed contour in other approaches, to true boundary of an object.

1. Edge-based objective function

Given the point distribution model and pose parameter – scale (s), rotation (θ), and translation (T_x, T_y), the combined parameter is represented by $\mathbf{p}=(s,\theta,T_x,T_y,b_1,b_2,\ldots,b_t)$. The point representation of the nth boundary points (n=0, 1,..., N–1) is

$$x(\mathbf{p},n) = s\cos\theta\left[\bar{x}_n + \sum_{k=1}^{t} Q_{2n,k}b_k\right] - s\sin\theta\left[\bar{y}_n + \sum_{k=1}^{t} Q_{2n+1,k}b_k\right] + T_x . \tag{3}$$
$$y(\mathbf{p},n) = s\sin\theta\left[\bar{x}_n + \sum_{k=1}^{t} Q_{2n,k}b_k\right] + s\cos\theta\left[\bar{y}_n + \sum_{k=1}^{t} Q_{2n+1,k}b_k\right] + T_y$$

We assume that *a prior* follows a multivariate Gaussian density as in [1]. Then *a prior* probability density of the boundary described by **p** is expressed as

$$\Pr(\mathbf{p}) = \prod_{i=1}^{N} \Pr(p_i) = \prod_{i=1}^{N} \frac{1}{\sigma_i\sqrt{2\pi}} e^{-\frac{(p_i-m_i)^2}{2\sigma_n^2}}, \tag{4}$$

where p_i is the ith component of **p**, m_i is the mean of p_i, and σ is the variance. For the pose parameters, the variance can be calculated from the training set alignment.

The maximization problem of the *a posteriori* probability with respect to the parameters **p**, **p** =(s, θ, T_x, T_y, b_1, b_2,..., b_t), be simplified to maximize [7]

$$M_{\text{edge}}(\mathbf{p}) = \sum_{j=1}^{t+4}\left[-\frac{(p_i-m_i)^2}{2\sigma_j^2}\right] + \frac{1}{\sigma^2}\sum_{n=1}^{N} E(x(\mathbf{p},n), y(\mathbf{p},n)), \tag{5}$$

where m_j is the mean of p_j, first term is *a prior* probability and second term is likelihood probability. The likelihood uses only the edge image, E, of an input image which is calculated by the Canny edge detector [9].

2. Curvature-based objective function

The goal of the curvature-based objective function is to find the most probable object considering an input curvature image. In terms of probabilities, we have to decide that curvature image (R_p) of which template described by a particular value of the parameter vector **p** corresponds the true curvature image R. It need to evaluate the probability of the curvature template given the image Pr(R_P|R), and find the maximum over **p**. This can be expressed using Bayes rule as

$$\Pr(R_{\max} \mid R) = \max_{\mathbf{p}} \Pr(R_\mathrm{p} \mid R) = \max_{\mathbf{p}} \frac{\Pr(R \mid R_\mathrm{p})\Pr(R_\mathrm{p})}{\Pr(R)}, \tag{6}$$

where R_{max} is the maximum *a posteriori* solution, Pr(R_P) is *a prior* probability density of the template curvature R_P, and Pr(R|R_P) is the conditional probability, or likelihood, of the curvature image given the template.

The curvature image of deformed template described by the parameter **p**, R_P, is directly obtained from the template boundary using curvature operator such as curvature scale space (CSS) [10]. Therefore, there is no unknown variable except **p** *in* R_p. Pr(R_p), *that can bias the boundary finder to* search for a particular range of shape and pose, is equal to Pr(**p**).

As the statistical curvature model in the chapter II, the curvature image R is represented a noise corrupted version of R_p with noise that is independent and additive: R= R_P+n, then Pr(R|R_P) is equivalent to Pr(R= R_P+n) or Pr(n=R-R_P). The noise at each pixel $n(x,y)$ equals $R(x,y)$-$R_p(x,y)$ and is governed by the probability density Pr(n). Assuming these events are independent for each point and Pr(n) follows Gaussian with zero mean and standard deviation σ_n, the likelihood is

$$\begin{aligned}\Pr(R|R_\mathrm{p}) &= \prod_A \Pr(n(x,y)) \\ &= \prod_A \frac{1}{\sqrt{2\pi}\sigma_n}\exp\left(-\frac{\left(R(x,y)-R_\mathrm{p}(x,y)\right)^2}{2\sigma_n^2}\right)\end{aligned}, \tag{7}$$

where A is a entire area of an image. Because the curvature template has support only along the boundary described by **p**, it is not necessary to sum over the entire image area. By taking the logarithm and making the curvature value in a template (R_P) be the corresponding curvature mean $\bar{r}$ of curvature model along the boundary and zero everywhere else. And we substitute the variation of the noise with the variation of the curvature model. By eliminating constant terms, we get the simple likelihood as

$$\begin{aligned}\Pr(R|R_\mathrm{p}) = \sum_A \ln\frac{1}{\sqrt{2\pi}\sigma_n} - \frac{1}{2\sigma_n^2}\Bigg(&\sum_{A-A_\mathrm{p}}\left(R(x,y)-R_\mathrm{p}(x,y)\right)^2 \\ &+\sum_{A_\mathrm{p}}\left(R(x,y)-R_\mathrm{p}(x,y)\right)^2\Bigg)\end{aligned}, \tag{8}$$

where $A_{\mathbf{P}}$ is the contour defined by the boundary $(x(\mathbf{p},n),y(\mathbf{p},n))$, $(n=1,2,\ldots,N)$, in a template $R_{\mathbf{P}}$. $R(x,y)$ and $R_{\mathbf{P}}(x,y)$ are curvature values of a pixel (x,y). Since $R_{\mathbf{P}}(x,y)$ is the curvature image obtained from the deformed template describe by **p**, we make the curvature value of $A_{\mathbf{P}}$ in a template ($R_{\mathbf{P}}$) be the corresponding curvature mean $\bar{r}$ of curvature model along the boundary and zero everywhere else. And we substitute the variation of the noise with the variation of the curvature model. By eliminating constant terms, we get the simple likelihood as

$$\Pr(R \mid R_{\mathbf{p}}) = -\sum_{j=1}^{N} \frac{(R(x(\mathbf{p},j), y(\mathbf{p},j)) - \bar{r}(j))^2}{2\sigma_r^2(j)}, \qquad (9)$$

We can expand *a posteriori* function for curvature images with the assumption of independent Gaussian noise at each pixel as

$$M_{\text{curvature}}(R_{\mathbf{p}}) = \ln \Pr(R_{\mathbf{p}}) - \sum_{j=1}^{N} \frac{(R(x(\mathbf{p},j), y(\mathbf{p},j)) - \bar{r}(j))^2}{2\sigma_r^2(j)} . \qquad (10)$$

Now, considering both edge and curvature, we define a posterior probability given input image that has edge image E and curvature image R. Assuming the events about edge and curvature are independent each other, a posterior probability follows

$$\Pr(t_{\mathbf{p}}, R_{\mathbf{p}} \mid \text{input image}) = \Pr(t_{\mathbf{p}} \mid E) \cdot \Pr(R_{\mathbf{p}} \mid R) \qquad (11)$$

By the logarithm, the combined Bayesian objective function with respect to parameter **p** is derived as

$$M_{com}(\mathbf{p}) = 2\sum_{j=1}^{t+4}\left[-\frac{(p_i - m_i)^2}{2\sigma_j^2}\right] + \qquad (12)$$
$$\sum_{n=1}^{N}\left[\frac{1}{\sigma_n^2} E\big(x(\mathbf{p},n), y(\mathbf{p},n)\big) - \frac{(R(x(\mathbf{p},n), y(\mathbf{p},n)) - \bar{c}_j)^2}{2\sigma_c^2(j)}\right]$$

The first term expressed by the logarithm of *a prior* probability and eliminating constant term is a double of Pr(**p**) because $\Pr(R_{\mathbf{p}})$ is equal to the Pr(**p**). The combined object function implies that the true boundary is the deformed template that satisfies the three properties. Firstly, the template described by **p** is similar to the boundary reconstructed with PDM. Secondly, the pixels representing the template have high edge magnitude. Thirdly, the curvature of template follows the statistic value acquired from training images using the curvature model

IV. Experimental Result

For implementation our algorithm, The steepest decent methods [8] for optimization of $M_{com}(\mathbf{p})$ is used. And we used the average errors as evaluation criteria for showing the performance of the proposed method. The boundary error of each labeled point on the deformed boundary template is calculated by finding the distance to the closest point on the true boundary.

The image (128×128) shown in Fig. 2(a) is a synthetic image where the target object is not a part of training set and is rotated and is occluded partially by other object. The initial deformed template represented by small white rectangles in Fig. 2(a) was defined by the mean of the training set. Since our method and the PDM use the same deformed boundary template described by **p** and the number of high curvature points considered are generally less than one of edge points, the computation cost of the two method is approximately equal in one iteration time. But the required iteration numbers for finding the accurate boundary were decreased by the proposed method as shown Fig. 2(d). The main cause of this result was that the labeled points having high curvature could find the true corresponding points rapidly avoiding fluctuation in edges of the corner region that occurred in applying only PDM. The final result of our method, Fig. 2(b), shows the insensitivity to rotation of an object and occlusion by other object. But the snake [6] could not extract the precise boundary of the object as Fig. 2(c). In this comparison, we can prove the effect of global shape information derived from training set. In Fig. 3, we experimented the robustness to noise by adding different amounts (from 1000 to 4000) of zero mean Gaussian noise to the synthetic images. The initial labeled points is defined by the mean boundary of training set. Applied low-quality real image like Infrared images, Fig. 4 is experimental results of the proposed method. This figure shows that our work gives quite good final contours although the input image has low-quality and the luminance of the object is similar to one of background.

Fig. 5 illustrates the experimental results of conventional methods. Fig. 5(a) is the result of watershed algorithm [5], Fig. 5(b) is the result of region growing algorithm [2][3]. These results show that the model-free methods are not adequate to finding a boundary of an object in low-quality images. Given the left image Fig. 4(a) as input, the final contour of active contour algorithm [6] is Fig. 5(c). Where the weights of curvature term and edge term were 1.0 and 0.8 respectively. In case that a boundary of an object is indistinct or the strong edge that is not part of the boundary exists near, the model-based method using only the constraints of contour such as continuity, bending energy, and smoothness, is difficult to get accurate boundary.

V. Conclusion

Given a training set that describes a desired object, we solve the problem of extracting the boundary of the object. The proposed method models curvature with statistical curvature model and formulates the curvature-based objective function that is derived using a maximum *a posteriori* criterion in Bayesian rule. The optimum solution of the combined objective function fits the deformed contour to the true boundary of an object. In the experimental results, we showed the robustness to noise, occlusion, and rotation. Comparing conventional PDM, it could decrease the computation cost. This work performs well in extracting a boundary of the object of which structure is a rigid body or is composed of pieces of rigid bodies. When the initial parameters are too far away from the true boundary, the optimization may be trapped by local minima corresponding to nearby edges. So the research about the robustness to the effect of initialization is needed further.

References

[1] L. H. Staib, and J. S. Duncan, "Boundary Finding with Parametrically Deformable Models," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 14, No. 11, pp. 1061-1075, 1992.

[2] P. K. Sahoo, S. Soltani, and A. K. C. Wong, "A Survey of Thresholding Techniques," CVGIP, Vol. 41, pp.233-260, 1988.

[3] Robert M. HarallickK and Linda G. Shapiro, "Survey of Image Segmentation Techniques," Computer Vision, Graphics, and Image Processing, Vol. 29, pp.100-132, 1985.

[4] P. Salembier and M. Pardas, "Hierarchical Morphological Segmentation for Image Sequence Coding," IEEE. Trans. Image Processing, Vol. 3, No. 5, Sep. 1994.

[5] L. Vincent and P. Soille, "Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulation," IEEE. Trans. PAMI, Vol. 13, No. 6, June 1991.

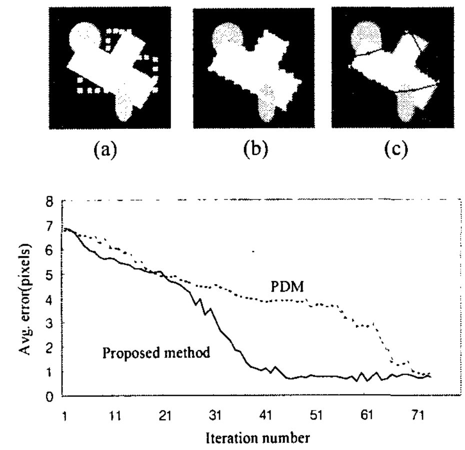
[6] M. Kass, A. Witkin, D. Terzopoulos, "Snakes : Active Contour Model", International Journal of Computer Vision, pp. 321 ~ 331, 1988.

[7] Y. Wang, and L.H. Staib, "Boundary Finding with Prior Shape and Smoothness Methods," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 22, No. 7, July 2000.

[8] M. Sonka, V. Hlavac, R. Boyle, " Image Processing, Analysis, and Machine Vision" PWS publishing, 1999.

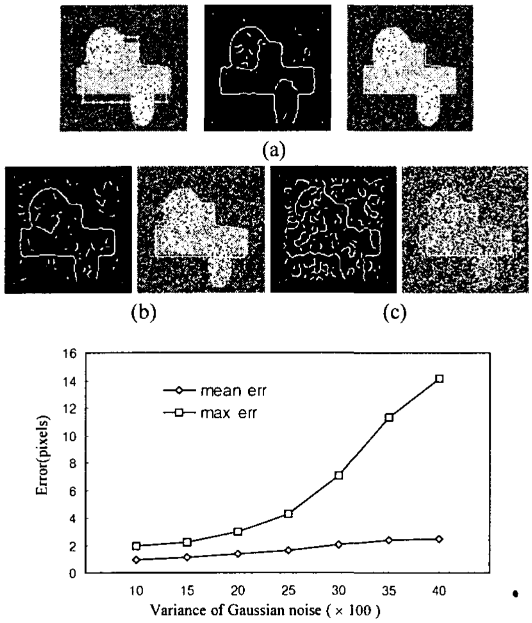
[9] J. F. Canny, "A computational approach to edge-detection," IEEE Trans. Pattern Anyaysis and Machine Intelligence, Vol. 8, No. 6, pp. 679-698, 1986.

[10] F. Mokhtarian, R. Suomela, "Curvature scale space for image point feature detection", in Proceedings of 7th ICIP, Vol. 1, pp. 206-210, 1999.



(d)

Fig. 2. Iteration number for convergence experiment (a) initial labeled points and input image (b) final result by the proposed method (c) final result of snake (d) mean error measure with respect to iteration numbers of the conventional PDM and the proposed algorithm



(d)

Fig. 3. Sensitivity to noise experiment (a) added Gaussian of that the variance is 1000, the input image and initial labeled points, Canny edge image, final boundary. (b) the variance is 2000, Canny edge image and final boundary. (c) the variance is 4000, Canny edge image and final boundary. (d) average error and maximum error with respect to each Gaussian noise

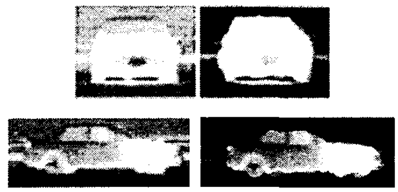


Fig. 4. Infrared image experiment. input image(left) and the result of extraction the object using the proposed method(right)

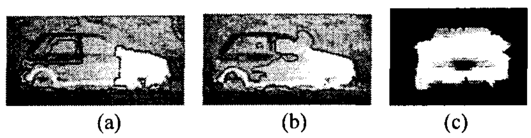


Fig. 5. Boundary extraction result using previous conventional method (a) result of watershed algorithm (b) result of region growing algorithm (c) result of active contour algorithm where weight of curvature term = 1.0, weight of edge term = 0.8 .