

Silhouette Extraction based on Iterative Spatio-temporal Local Color Transformation and Graph-Cut Segmentation

Yasushi Makihara, Yasushi Yagi
Osaka University
{makihara, yagi}@am.sanken.osaka-u.ac.jp

Abstract

We propose an iterative scheme of spatio-temporal local color transformation of background and graph-cut segmentation for silhouette extraction. Given an initial background subtraction, spatio-temporal background color transformation is processed for fitting modeled background colors to input background ones under a different illumination condition. After foreground colors are modeled based on the fit background, spatio-temporal graph-cut algorithm is applied to acquire a foreground/background segmentation result. Because these two processes need well-segmented background and well-fit background each other, they are iterated in turn to obtain better silhouette extraction results. Silhouette extraction experiments for a walking human on a treadmill show the effectiveness of the proposed method.

1. Introduction

Silhouette extraction is an essential preprocessing in many computer vision areas of object recognition, motion analysis, and visual hull-based 3-D reconstruction. In particular, silhouette qualities have great impact on silhouette-based recognition including gesture and gait recognition. For these purposes, background subtraction have been widely used under scenes with a static camera by modeling the background pixel as a Gaussian distribution [8] or a mixture of Gaussian distribution [5]. The methods, however, often suffer from the following well-known two problems: (1) Under-segmentation in case where foreground colors are similar to background ones, (2) Over-segmentation in case of background color changes between background modeling and test phases (see Fig. 1(a)-(d) for example).

In order to overcome the illumination changes, selective background update [6] or adaptive background modeling by Parzen density estimation [7] were proposed. Though correct background selection (i.e. foreground/background segmentation) is essential for efficient and accurate background update, it is difficult to

segment out the background if pixel value distributions between the background modeling and test phases are considerably different due to illumination changes by time elapse.

On the other hand, substantial segmentation works using graph-cut algorithm have achieved outstanding performances through user interaction [1][4] or by iterative scheme with prior knowledge [2]. For static-background scenes, if the background subtraction results are reliable, it can be used as powerful prior knowledge for graph-cut segmentation.

Therefore, we propose an iterative scheme of background update by color transformation and a graph-cut segmentation based on the updated background. After obtaining an initial background subtraction, the colors of a modeled background are transformed so as to fit to a current background region under a different illumination condition considering spatio-temporal smoothness. Note that the color transformation-based method can update all the pixels in an image whereas pixel-based update methods [6][7] can only update pixels recognized as background. Then, the updated background is provided to a graph-cut segmentation module, and better segmentation results are returned to the background update module. Finally, these two processes are iterated until convergence.

2 Background update by spatio-temporal local color transformation

In this paper, background pixels are modeled as Gaussian distribution separately in advance, and then Mahalanobis distances based on the distribution are exploited as background subtraction measure [8].

Input background colors are often different from the modeled colors due to illumination changes such as soft cast shadow from foreground objects. To fit the modeled background colors to the input ones under such illumination changes, a linear color transformation derived from the finite-dimensional linear model [3] is introduced as follows,

$$\begin{bmatrix} R_{in} \\ G_{in} \\ B_{in} \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{bg} \\ G_{bg} \\ B_{bg} \\ 1 \end{bmatrix} \quad (1)$$

$$\mathbf{c}_{in} = A \mathbf{c}_{bg}, \quad (2)$$

where $\mathbf{c}_{in} = [R_{in}, G_{in}, B_{in}, 1]^T$ and $\mathbf{c}_{bg} = [R_{bg}, G_{bg}, B_{bg}, 1]^T$ are an input and a modeled background color respectively, and $A = \{a_{ij}\}$ is a linear color transformation matrix. Then, a global color transformation is first obtained by minimizing the following objective function S_{global}

$$\mathbf{d}_{tr}(x, y) = A \mathbf{c}_{bg}(x, y) - \mathbf{c}_{in}(x, y) \quad (3)$$

$$S_{global} = \sum_{x, y} w_{bg}(x, y) \mathbf{d}_{tr}(x, y)^T \Sigma_{bg}(x, y)^{-1} \mathbf{d}_{tr}(x, y), \quad (4)$$

where $\mathbf{d}_{tr}(x, y)$, $\Sigma_{bg}(x, y)$, $w_{bg}(x, y)$ are color transformation error, covariance matrix of modeled background color, and background weight for each pixel respectively. The background weight $w_{bg}(x, y)$ is initialized as follows,

$$\mathbf{d}(x, y) = \mathbf{c}_{bg}(x, y) - \mathbf{c}_{in}(x, y) \quad (5)$$

$$w_{bg}(x, y) = \exp(-\kappa_{bg} \mathbf{d}(x, y)^T \Sigma(x, y)^{-1} \mathbf{d}(x, y)), \quad (6)$$

where κ_{bg} is coefficient for the weight. Note that the weight is replaced by background region mask resulted from the graph-cut segmentation in iteration phases.

Figure 1 (e) and (f) show an updated background by global color transformation and updated background weight. We can see that the background color is fit as a whole and that the background weights become better than the initial one (Fig 1(d)). However, this global transformation cannot deal with spatially local illumination changes such as soft cast shadow on the belt and the carpet, and saturation on the green surrounding screens as shown in Fig. 1 (f).

Thus, we extend this color transformation to spatio-temporal local ones. First, control points of color transformation are aligned at s_x, s_y, s_t intervals in the spatio-temporal (x, y, t) domain. Second, color transformation parameters at each point (x, y, t) is expressed as linear interpolation:

$$A(x, y, t) = \sum_{(u, v, s)} w(x, y, t; u, v, s) A(u, v, s), \quad (7)$$

where $(u, v, s) = (x/s_x, y/s_y, t/s_t)$ is control point coordinate, $A(u, v, s)$ is color transformation parameters at the control point (u, v, s) , and $w(x, y, t; u, v, s)$ is weight on control point (u, v, s) for linear interpolation at point (x, y, t) . Finally, the local parameters are determined by minimizing the following objective function S_{local}

$$\mathbf{d}_{tr}(x, y, t) = A(x, y, t) \mathbf{c}_{bg}(x, y, t) - \mathbf{c}_{in}(x, y, t) \quad (8)$$

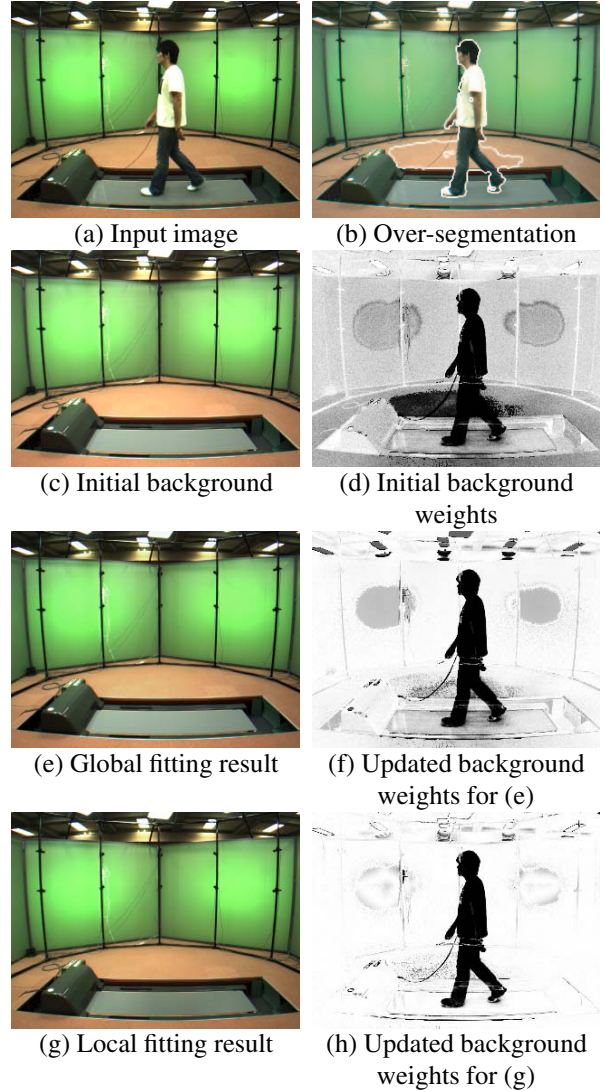
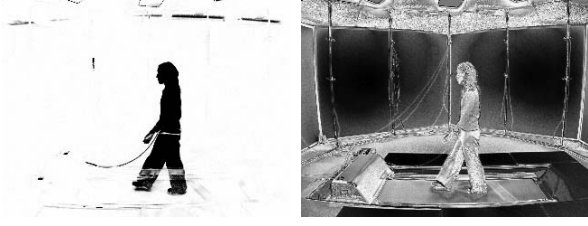


Figure 1. Spatio-temporal background color transformation

$$S_{local} = \sum_{x, y, t} w_{bg}(x, y, t) \mathbf{d}_{tr}(x, y, t)^T \Sigma_{bg}(x, y, t)^{-1} \mathbf{d}_{tr}(x, y, t) + \sum_{u, v, s} \left(\alpha_u \left| \frac{\partial \mathbf{a}}{\partial u} \right|^2 + \alpha_v \left| \frac{\partial \mathbf{a}}{\partial v} \right|^2 + \alpha_s \left| \frac{\partial \mathbf{a}}{\partial s} \right|^2 \right) \quad (9)$$

where \mathbf{a} is a vector containing 12 color transformation parameters a_{ij} ($i = 1, 2, 3, j = 1, 2, 3, 4$) at each control point. The first and second terms are called data term and regularization term respectively. Coefficients $\alpha_u, \alpha_v, \alpha_s$ indicate smoothness weights for horizontal, vertical, and temporal axes respectively.

Figure 1 (g) and (h) show an updated background and background weights by the spatio-temporal local color transformation respectively. We can see that the background colors fit more accurately than the global color transformation (cf. Fig. 1(e)) and that background weights are much improved (cf. Fig. 1(f)).



(a) Background data term (b) Foreground data term

Figure 2. Data term

3 Graph-cut segmentation

Graph-cut segmentation is generally formulated as the following energy minimization problem

$$E(X) = \sum_{q \in Q} g_q(X_q) + \sum_{(p,q) \in E} h_{pq}(X_p, X_q), \quad (10)$$

where X is foreground/background label, Q is a set of all sites (x, y, t) , and E is all combinations of neighborhood sites. The first term is called as *data term* and the second term is *smoothness term*. Based on the data term and smoothness term defined in the following subsections, max-flow algorithm gives segmentation results so as to minimize eq. (10) globally.

3.1 Data term

First, the updated background weights are used as background data term $g_q(X_q = BG)$ (see Fig. 2(a)). Next, foreground object colors are approximated by GMM (Gaussian Mixture Model) and foreground data term $g_q(X_q = FG)$ is expressed as

$$\mathbf{d}_q^k = \mathbf{c}_q - \mathbf{e}^k \quad (11)$$

$$g_q(X_q = FG) = \exp(-\kappa_{fg} \min_k \{ \mathbf{d}_q^{kT} \Sigma^k \mathbf{d}_q^k \}), \quad (12)$$

where \mathbf{e}^k and Σ^k are mean and covariance of k th Gaussian respectively, \mathbf{c}_q is an input color at site q , and κ_{fg} is coefficient for the foreground data term. This GMM is trained by k -clustering from a foreground sample regions, which are obtained by thresholding background data term and morphological operation containing closing, opening, and area filter. We can see foreground region have relatively high foreground data values as shown in Fig. 2 (b), although some background regions such as a carpet and a treadmill belt have also high values.

3.2 Smoothness term

Smoothness term contributes to foreground/background boundary decision based on boundary edges. In this paper, 6 neighbor connections in the spatio-temporal 3D domain are treated. The smoothness term considering intensity value normalization is expressed as

$$h_{pq}(X_p, X_q) = \begin{cases} 0 & (X_p = X_q) \\ \alpha_{sm} \exp(-\kappa_{sm} \frac{|c_q - c_p|^2}{|c_q + c_p|^2 + \epsilon}) & (X_p \neq X_q), \end{cases} \quad (13)$$

where α_{sm} , κ_{sm} , and ϵ are coefficients for the smoothness term.

4 Experiments

We made silhouette extraction experiments using a sequence of a walking human on a treadmill. The camera used was Point Gray Research Flea2, and images were captured by 640×480 pixel size at 60 fps and were scaled down to 320×240 pixel size. A total of 60 images were provided for spatio-temporal color transformation and graph-cut segmentation as a block.

In this experiment, control point intervals and smoothness constraint coefficients for spatio-temporal color transformation were experimentally set as $s_x = s_y = 20, s_t = 60, \alpha_u = \alpha_v = \alpha_s = 1.0$ respectively. In addition, the parameters for graph-cut segmentation were set as $\kappa_{bg} = 8.0, \kappa_{fg} = 1.5, \alpha_{sm} = 1.0, \kappa_{sm} = 1.0, \epsilon = 1.0$ respectively.

Figure 3 shows segmentation results in iteration processes for the first frame of the sequence. Given an initial background (Fig. 3(a)) and an input image (Fig. 3(b)), an initial background subtraction (Fig. 3(c)) is acquired. Because the initial subtraction is not accurate, the background color fitting in the first iteration is not enough. As a result, background update and segmentation result are poor as shown in the first row of Fig. 3 (d) and (e). Then, based on the segmentation result, the background fitting gradually works better and over-segmentation is also reduced as shown in the second and the third rows of Fig. 3 (d) and (e).

Finally, an accurately fit background and segmented region are obtained as shown in forth row of Fig. 3 (d) and (e). Note that under-segmentation at a leg part is inevitable if a simple background subtraction is used as easily expected from background subtraction result (Fig. 3(f)). Actually, the under-segmentation is avoidable thanks to foreground data term and smoothness term in graph-cut segmentation.

Figure 4 shows quantitative performance evaluation in the iteration processes. First, residuals of background color fitting measured by averaged Mahalanobis distances in the true background region are picked up (Figure 4(a)). It turns out that the residuals decrease as iteration processes go on, and that residuals for spatio-temporal fitting are always lower than those for global fitting.

Second, precision of segmentation result compared with manually segmented ground truth are shown in Fig. 4(b). As a result, we can see that graph-cut segmentation with local color fitting achieved the best performance after several iterations.

Finally, background color fitting and graph-cut segmentation take about 250 msec and 30 msec per frame respectively on computer with 3.2GHz CUP and 2.0GB RAM. Both cost and accuracy for background color fitting are strongly dependent on the number of control points, therefore the optimal number of control point should be investigated considering tradeoff between them in the future works.

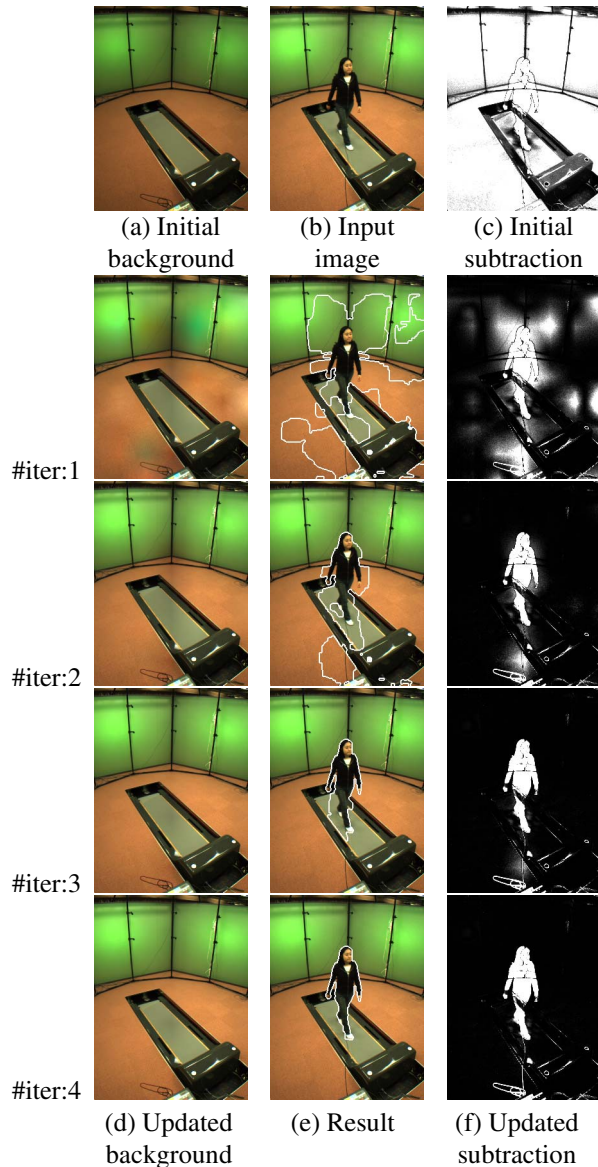
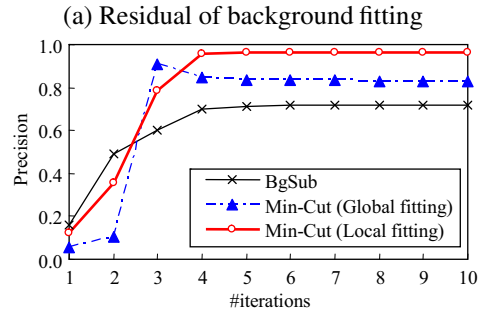
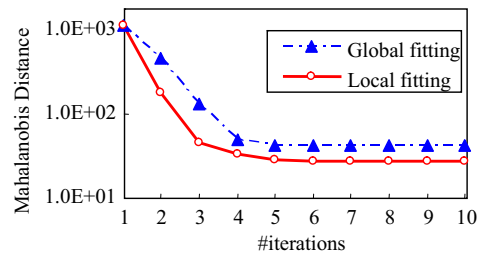


Figure 3. Segmentation results in iteration process

5 Conclusions

We proposed an iterative scheme of spatio-temporal local color transformation of background and graph-cut segmentation for silhouette extraction. First, spatio-temporal background color transformation is used for fitting modeled background colors to input background ones under a different illumination condition to avoid the over-segmentation. Then, based on the fit background, graph-cut segmentation involving foreground color modeling reduces the under-segmentation. These two processes are iterated to obtain better silhouette extraction results. From silhouette extraction experiments for a walking human on a treadmill, it turns out that the proposed method successfully fits the background color in the spatio-temporally local domain and that segments



(b) Precision

Figure 4. Performance in iteration process

the silhouettes accurately.

Acknowledgements

This work is supported by the Special Coordination Funds for Promoting Science and Technology of Ministry of Education, Culture, Sports, Science and Technology.

References

- [1] Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *Proc. of Int. Conf. on Computer Vision*, 2001.
- [2] W. Lee, W. Woo, and E. Boyer. Identifying foreground from multiple images. In *Proc. of the 8th Asian Conf. on Computer Vision*, pages 580–589, Nov. 2007.
- [3] D. Marimont and B. Wandell. Linear models of surface and illuminant spectra. *J. Opt. Soc. Amer. A*, 9(11):1905–1913, 1992.
- [4] C. Rother, V. Kolmogorov, and A. Blake. Grabcut-interactive foreground extraction using iterated graph cuts. In *Proc. of ACM SIGGRAPH 2004*, 2004.
- [5] S. Rowe and A. Blake. Statistical mosaics for tracking. *Image and Vision Computing*, 14(8):549–564, 1996.
- [6] B. Shoushtarian and H. Bez. A practical adaptive approach for dynamic background subtraction using an invariant colour model and object tracking. *Pattern Recognition Letter*, 26(1):5–26, 2005.
- [7] T. Tanaka, A. Shimada, D. Arita, and R. Taniguchi. A fast algorithm for adaptive background model construction using parzen density estimation. In *Proc. of IEEE International Conference on Advanced Video and Signal based Surveillance*, Sep. 2007.
- [8] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfunder: Real-time tracking of the human body. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.