Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 320141, 12 pages
http://dx.doi.org/10.1155/2014/320141

Research Article

The Dynamic Model Embed in Augmented Graph Cuts for Robust Hand Tracking and Segmentation in Videos

Jun Wan,[1,2] Qiuqi Ruan,[1,2] Gaoyun An,[1,2] Wei Li,[1,2] Yanyan Liang,[3] and Ruizhen Zhao[1,2]

[1] *Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China*
[2] *Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China*
[3] *Space Science Institute, Macau University of Science and Technology, Macau 999078, China*

Correspondence should be addressed to Gaoyun An; gyan@bjtu.edu.cn

Received 10 April 2014; Accepted 15 August 2014; Published 31 August 2014

Academic Editor: Paolo Lonetti

Segmenting human hand is important in computer vision applications, for example, sign language interpretation, human computer interaction, and gesture recognition. However, some serious bottlenecks still exist in hand localization systems such as fast hand motion capture, hand over face, and hand occlusions on which we focus in this paper. We present a novel method for hand tracking and segmentation based on augmented graph cuts and dynamic model. First, an effective dynamic model for state estimation is generated, which correctly predicts the location of hands probably having fast motion or shape deformations. Second, new energy terms are brought into the energy function to develop augmented graph cuts based on some cues, namely, spatial information, hand motion, and chamfer distance. The proposed method successfully achieves hand segmentation even though the hand passes over other skin-colored objects. Some challenging videos are provided in the case of hand over face, hand occlusions, dynamic background, and fast motion. Experimental results demonstrate that the proposed method is much more accurate than other graph cuts-based methods for hand tracking and segmentation.

1. Introduction

There are four main kinds of object tracking methods which are points, skeleton, contour, and silhouette tracking in recent papers [1, 2]. As an important branch of tracking, hand tracking is a critical step in computer vision systems, such as human computer interaction (HCI) [3], sign language interpretation [4], and gesture recognition [5]. Besides, vision-based hand gesture recognition [3] is a meaningful direction to enable computers to understand the meaning in robot systems where the first key step is to achieve robust hand tracking. Hence, we concentrate on silhouette tracking which means that hand silhouette or region should be split from cluttered backgrounds.

In the last decade [6], human hand motion capture has gained widespread interest in pattern recognition area. For example, Yang et al. [5] presented a method to obtain hand trajectories based on pixel matches with affine transformations. Then an optical flow-based method [7] is proposed for hand tracking. Although the method [7] can capture quick motion and fast hand shape deformations, it still fails to hand tracking when hands and skin-colored objects are occluded. Some other works [4, 8] try to use linear quadratic estimation model (e.g., Kalman filter) or sequential Monte Carol model (e.g., particle filter) to hand track trajectory. Later on, a real-time hand tracking method is applied in a mechanical device by the authors [9] who utilized the advantages of particle filter and mean shift (MS). They incorporate MS optimization into particle filter to improve the sampling efficiency considerably. Though these approaches have delivered promising results, they are difficult to handle occlusions.

In recent years, graph cuts-based methods have been applied in tracking or segmentation systems. Xu and Ahuja [10] firstly proposed a method to track object contour by graph cuts. They dilate object contour into a narrow band and construct a graph only on this band. Nevertheless, it cannot deal with large displacements because there is no dynamic model to estimate object location. Freedman and Turek [11] presented a method based on graph cuts to track objects when the illumination drastically changes. Yet, they do not achieve object segmentation from their experimental results. Later, Malcolm et al. [12] incorporated a distance penalty into graph cuts to realize object segmentation and used a simple filter to estimate the location of interested objects. Although this method can achieve multiobject tracking, it still cannot deal with occlusions. Bugeau and Pérez [13] proposed a method based on optical flow and graph cuts to simultaneously track and segment objects. However, this method needs a reference background image that would restrict its application and popularization. In the work of [14], the authors managed to track objects in live videos via reseeding strategy. And Papadakis and Bugeau [1] presented that the interested object is comprised by visible and occluded parts which are tracked, respectively. Regardless of the fact that those methods have achieved success in some areas, they still have some drawbacks in some situations, such as hand over face and hand occlusions.

Hand tracking is a challenging problem because the hand presents 27 degrees of freedom (DOFs), including 21 DOFs for the joint angles and 6 DOFs for orientation and location [6]. Therefore, hand shape and motion are more arbitrary than rigid objects. In this paper, we present an effective approach to track and segment hands even though hands have arbitrary shape deformations. Similar to the methods of [12, 13], a dynamic model and graph cuts are used. However, compared with these methods, the key contributions of our method are summarized as follows.

(i) To avoid the degeneracy problem of interest points [12, 13], we combine the resampling strategy and optical flow algorithm to robustly track interest points from hand regions.

(ii) An augment graph cuts method is introduced to track and segment hand regions and different hands labelled with different colors.

(iii) The proposed method can track and segment hands on some challenging environments, such as hands overlap, hand fast motion, and hand over face. Also the proposed method can track and segment hands in dynamic backgrounds where some skin-colored objects may be present.

The framework of our method is shown in Figure 1, which consists of optical flow estimation and augmented graph cuts introduced in Section 3.

This paper is a substantial extension of our conference paper [15]. Compared with [15], further details of our method are presented, and more extensive performance evaluation is conducted. We also give a more comprehensive literature review to introduce the background of our method and make the paper more self-contained. Therefore, this paper provides a more comprehensive and systematic report of our work. The rest of the paper is organized as follows. We describe basic notions of multiobject tracking based on graph cuts in Section 2. The proposed method is described in Section 3. Section 4 shows the experimental results and the performance evaluation. The conclusion is given in Section 5.

2. Notion of Traditional Graph Cuts

Here, we describe the basic principle of graph-cuts based methods for object tracking and segmentation. We review image segmentation via graph cuts at first. Then, object tracking is described via graph cuts and dynamic model.

2.1. Segmentation via Graph Cuts. We briefly outline multi-label graph cuts technique. The detailed information can be found in [16, 17]. The simple segmentation of the background and objects can be obtained by minimizing the following energy with respect to the labelling function in (1):

$$\varepsilon(\lambda) = \varepsilon_D(\lambda) + \varepsilon_S(\lambda), \quad (1)$$

where data term ε_D evaluates the likelihood $p_n(i)$ of a pixel i belong to the nth object and ε_D is defined as

$$\varepsilon_D(\lambda) = -\sum_{i\in I}\sum_{n=0}^{N} \ln\left(p_n(i)\right)\delta(\lambda, n), \quad (2)$$

where $\delta(a, b)$ is delta function (equal to 1 if $a = b$ and 0 otherwise); I represents an image; N is the number of tracked objects; $p_n(i)$ is calculated by a normalized histogram of the nth object.

The smooth term $\varepsilon_S(\lambda)$ evaluates the penalty for assigning two neighboring pixels to different labels. $\varepsilon_S(\lambda)$ is defined as

$$\varepsilon_S(\lambda) = \omega_S \sum_{(p,q)\in w, \lambda_p \neq \lambda_q} \left(\exp\left(-\frac{\left(I_p - I_q\right)^2}{\delta^2} \right) \cdot \left(\frac{1}{\|p - q\|} \right) \right) \times \left(1 - \delta\left(\lambda_p, \lambda_q\right)\right), \quad (3)$$

where p and q are coordinates of pixels, $\|p - q\|$ is the Euclidean distance, $\omega_S \geq 0$ is a smooth parameter, $\sigma = \langle \|I_p - I_q\|^2 / \|p - q\|^2 \rangle$, and w is all neighborhood pixel pairs which are 4 or 8 neighborhood systems.

2.2. Tracking via Graph Cuts. Suppose N objects are tracked and $I^t \subset IR^2$ is a set of pixels at time t of an image $I(x, t)$. o_n^t represents the nth object at time t (o_0^t denotes the background region). Therefore, we can know that $I^t = \bigcup_{n=0}^{N} \mathrm{o}_n^t$. Equations (1)–(3) can be rewritten by adding temporal information:

$$\varepsilon(\lambda, t) = \varepsilon_D(\lambda, t) + \varepsilon_S(\lambda, t), \quad (4)$$

$$\varepsilon_D(\lambda, t) = -\sum_{i\in I}\sum_{n=0}^{N} \ln\left(p_n^t(i)\right)\delta(\lambda, n), \quad (5)$$

$$\varepsilon_S(\lambda, t) = \omega_S \sum_{(p,q)\in w^t, \lambda_p \neq \lambda_q} \left(\exp\left(-\frac{\left(I_p^t - I_q^t\right)^2}{\delta^2} \right) \cdot \left(\frac{1}{\|p - q\|} \right) \right) \times \left(1 - \delta\left(\lambda_p, \lambda_q\right)\right). \quad (6)$$

In [12], it assumes that the mean velocity is known for each object, the authors translate the current object o_n^t at time

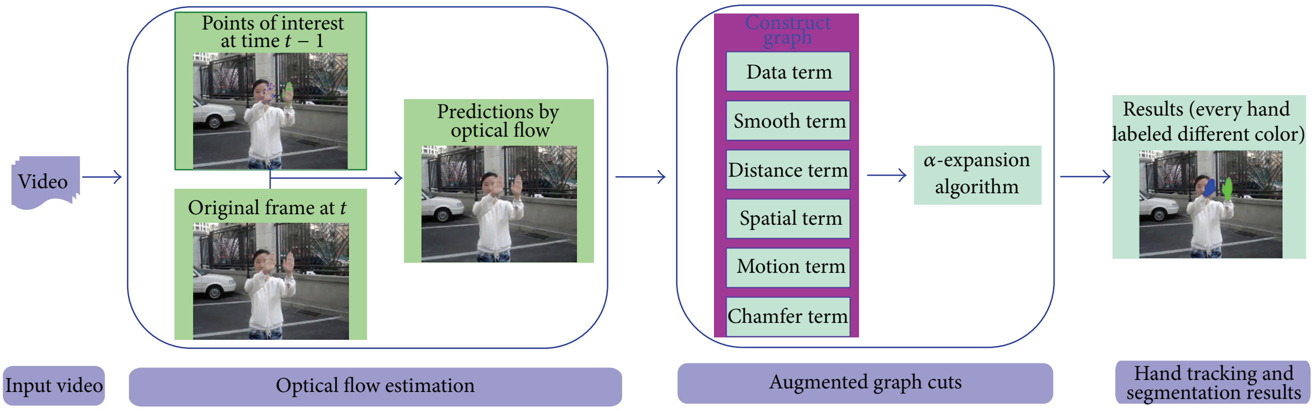

Video
Points of interest at time $t-1$
Original frame at t
Predictions by optical flow
Construct graph
Data term
Smooth term
Distance term
Spatial term
Motion term
Chamfer term
α-expansion algorithm
Results (every hand labeled different color)
Input video
Optical flow estimation
Augmented graph cuts
Hand tracking and segmentation results

Figure 1: The framework of our method for hand tracking and segmentation.

t to have a prediction $o_n^{t+1|t}$ at time $t+1$. A new term called distance term ε_T is introduced, which discourages pixels to be associated with the nth object, when the pixels do not belong to the predicted set $o_n^{t+1|t}$. ε_T is defined as

$$\varepsilon_T(\lambda, t) = \omega_T \sum_{i\in I} \sum_{n=1}^{N} \alpha(\tilde{c} - c_n)\, d_n(i)\, \delta(\lambda(i), n), \qquad (7)$$

where $\omega_T > 0$, $\alpha(\tilde{c} - c)$ is a scaling function explained in Section 3, $d_n(i) = \min_{Z\in o_n^{t+1|t}} \|i - Z\|$ which constraints a new estimate to be in the spatial neighborhood of the prediction. For example, if a pixel i is in the mask of predicted object $o_n^{t+1|t}$, then $d_n(i) = 0$. If a pixel i is out of the mask of $o_n^{t+1|t}$, $d_n(i)$ is equal to the nearest distance between i and other pixels $Z, \forall Z \in o_n^{t+1|t}$. $d_n(i)$ can be quickly calculated with fast matching algorithm [18]. Therefore, the energy function is reformulated as

$$\varepsilon(\lambda, t) = \varepsilon_D(\lambda, t) + \varepsilon_S(\lambda, t) + \varepsilon_T(\lambda, t). \qquad (8)$$

Although the methods [12, 13] have achieved to track and segment objects which are partly occluded in some occasions, they cannot access to track overlapped objects when these objects are similar colors (e.g. the hands and face are overlapped). We give an example to illustrate the limitations of these methods in Figure 2. In the initialization step at time $t = 0$, the result of the left/right hand is labelled blue/green as shown in Figure 2(a). At time $t = 1$, in Figures 2(b) and 2(c), we can see that the colors are confused between the left and right hand by [12, 13]. Besides, some pixels in the red circle are wrongly labelled by [13] as shown in Figure 2(b) while pixels in the background are correctly segmented by the method [12] (see Figure 2(c)). That is, because ε_T is added in [12] to constrain the estimation to be in the spatial neighbourhood of the prediction. However, the method [12] still does not distinguish pixels of each hand (see Figure 2(c)).

3. The Proposed Method

Suppose that N hands are tracked and each hand is totally visible at time $t = 0$. This means that $o_{n1}^0 \cap o_{n2}^0 = \phi$, $n1 \neq n2$. The initialized segmentations o_n^0 labelled different colors are provided by manual operation at time $t = 0$. At time $t > 0$, our approach can sequentially process the frames for simultaneously hand tracking and segmentation.

3.1. State Prediction

3.1. State Prediction. When the segmentation result o_n^t is correct at time t, the prediction set $o_n^{t+1|t}$at time $t+1$ is estimated by the mean velocity $\overline{v}_n^t$ using (9) as

$$o_n^{t+1|t} = o_n^t + \overline{v}_n^t; \quad o_n^t \subset I^t, \quad o_n^{t+1|t} \subset I^{t+1}, \quad n = 1, 2, \ldots, N. \qquad (9)$$

To compute unknown mean velocity $\overline{v}_n^t$, some methods (such as autoregression model [19] and interest points detector [20, 21]) have been proposed in the past decades. Compared with these methods, optical flow delivers excellent results on fast moving objects with a high computational efficiency [7]. Therefore, we choose optical flow (the same as the method [13]) based on pyramid Lucas-Kanade multiresolution scheme [22] as our dynamic model. However, there are two problems shown in the methods [13]. The first problem is that some interesting points may be wrongly detected by optical flow as shown in the first row of Figure 3, and the second is the degeneracy problem which perhaps happens in the second row of Figure 3. In our dynamic model, two strategies are introduced for avoiding these two problems.

To compute the unknown velocities, a set of interest points is considered. At time $t = 0$, the interest points $\{f_{nm}^0\}_{m=1,\ldots,M_n^0} \in o_n^0$ are found by good-feature-to-track [7, 21] which suggests seeking a steep brightness gradient along at least two directions for promising feature candidates. Then, at time $t > 0$, $\{f_{nm}^t\}_{m=1,\ldots,M_n^t}$ can be detected [22]. So the velocity is computed between two successive frames as

$$v_{nm}^t = f_{nm}^t - f_{nm}^{t-1}, \quad n = 1, \ldots, N, m = 1, \ldots, M_n^{t-1}. \qquad (10)$$

And the mean velocity $\overline{v}_n^t$ at time t is calculated as

$$\overline{v}_n^t = \sum_{m=1}^{M_n^{t-1}} \frac{v_{nm}^t}{M_n^{t-1}}. \qquad (11)$$

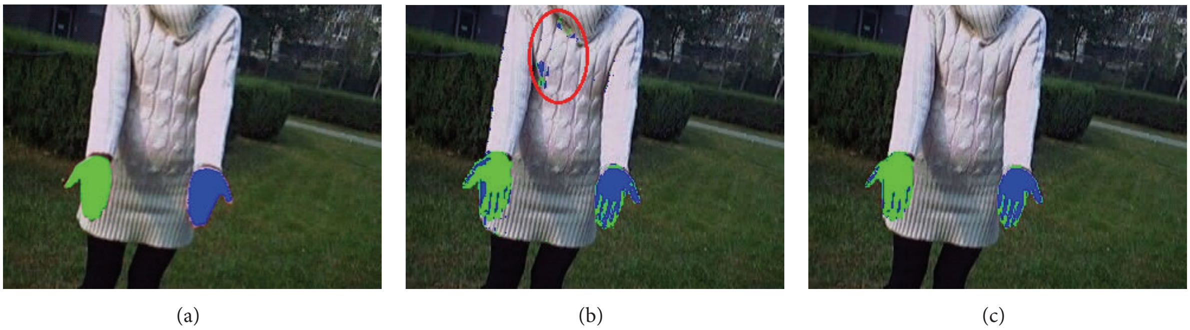


Figure 2: (a) Initialization at time $t = 0$, (b) results by [13] with $\omega_S = 5$ at time $t = 1$, and (c) results by [12] with $\omega_S = 10$, $\omega_T = 1$ at time $t = 1$.

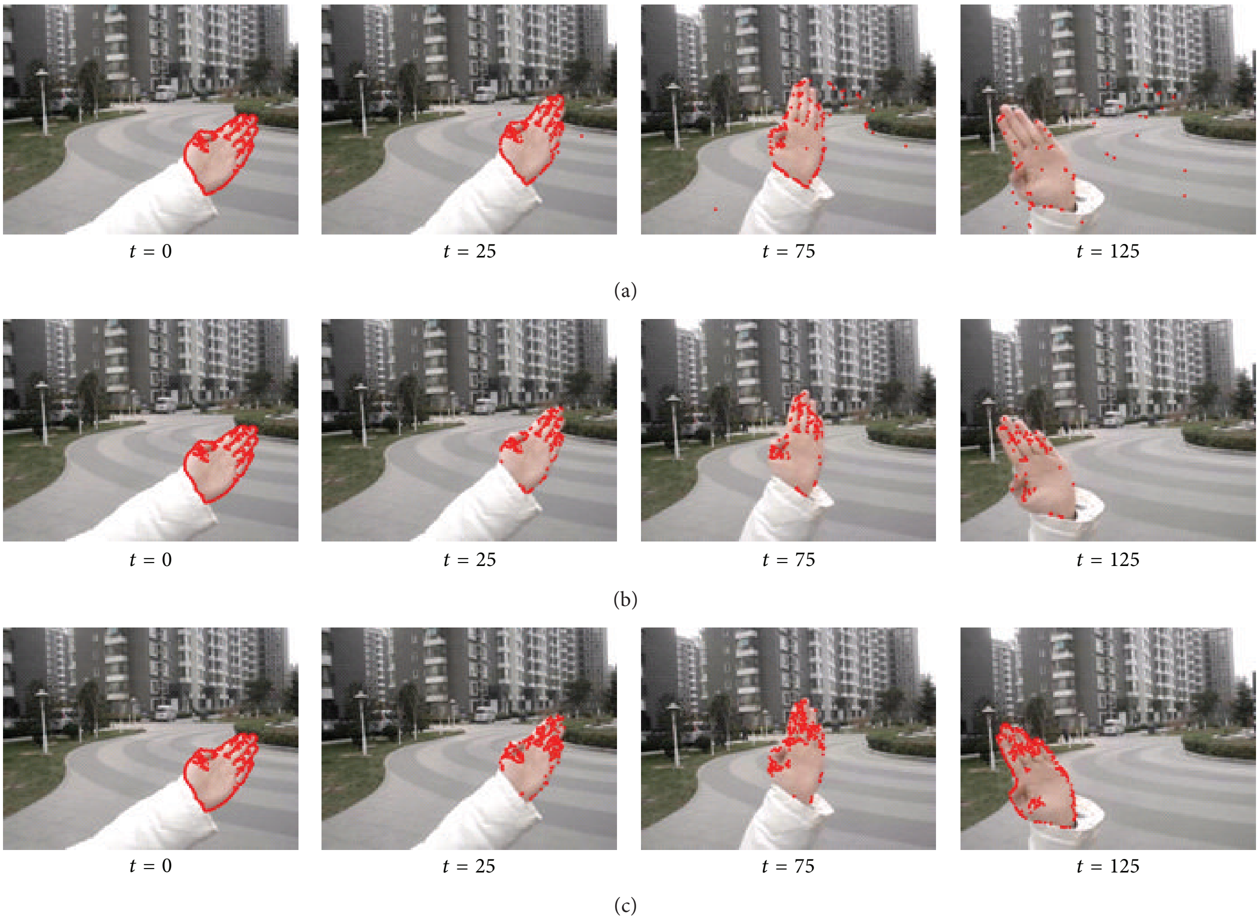


Figure 3: (a) Some points are out of hand range via optical flow at $t = 75, 125$. (b) The degeneracy problem occurs (the number of points is drastically reduced at time $t = 125$). (c) Results by our method.

From (10), we know that every detected point has contributions to the mean velocity. When some points are beyond the scope of hand region (see the first row of Figure 3), the mean velocity may have a bias to true velocity. So a distance penalty in (12) is created to eliminate outlines. Here, we only consider points when their displacements are less than a given threshold τ_1 as

$$d\left(f_{nm}^{t}, f_{nm}^{t-1}\right) = \left\| f_{nm}^{t} - f_{nm}^{t-1} \right\| < \tau_1. \quad (12)$$

In order to capture fast hand motion, we can set a large value to τ_1. In our experiments, $\tau_1 = 80$ is well suitable for all test videos.

As time goes on, the number of interest points may goes down via optical flow (see the second row in Figure 3). For the sake of avoiding the degeneracy problem, the second strategy is to resample interest points. When the number of interest points is below a given threshold τ_2, we can redetect new interest points using good-features-to-track [21]. After these two strategies, the detected interest points are shown in the third row of Figure 3.

3.2. Error of Prediction. In this work, we accept the idea of the work [12] to handle the error prediction problem. The prediction error is the distance between the predicted centroid $\vec{c}_n^t$ and the actual centroid c_n^t at time t. The scaling function is defined as

$$\alpha\left(\vec{c}_n^t, c_n^t\right) = \exp\left(\frac{-\left(\vec{c}_n^t - c_n^t\right)^2}{\rho^2}\right). \tag{13}$$

Here, ρ is a threshold based on empirical motion, which controls the change rate of penalty α. If ρ is large, α will slowly change. As mentioned in [12], in practice, $\rho = 3.5$ is quite robust to our model. When the actual sets o_n^t are off $o_n^{t|t-1}$, α is lowered to hopefully still capture motion. α can automatically rise when prediction errors decrease by (13).

3.3. Augmented Graph Cuts. Now we explain how to define new terms and incorporate them into energy function. Those new terms are the core principle in augmented graph cuts.

3.3.1. Spatial Constraint. Owing to the similar color of human skin, it is difficult to eliminate the effect of each hand by the works [12, 13] as shown in Figure 2. Here, we introduce a new energy term called spatial term ε_C:

$$\varepsilon_C(\lambda, t) = \omega_C \sum_{i\in I}\sum_{n=1}^{N} \psi\left(i, c_n^{t|t-1}\right)\delta(\lambda(i), n), \tag{14}$$

where $c_n^{t|t-1}$ denotes the centroid of the predict set o_n^t. $\omega_C > 0$ is the parameter value. The penalization is made through the function $\psi(\cdot)$:

$$\psi\left(i, c_n^{t|t-1}\right) = \exp\left(\frac{\left\|i - c_n^{t|t-1}\right\|}{\sum_{n=1}^{N}\left\|i - c_n^{t|t-1}\right\|}\right), \tag{15}$$

where $\|i - c_n^{t|t-1}\|$ is the Euclidean distance from the location of a pixel to the centroid $c_n^{t|t-1}$ of $o_n^{t|t-1}$. When a pixel $i \in I^t$ is close to $c_n^{t|t-1}$, the value ε_C becomes a small value which indicates that the pixel i is encouraged to assign the nth object.

As illustrated in Figure 4(a), when hands are visible ($o_1 \cap o_2 = \phi$), then $\psi(i, c_1) < \psi(i, c_2)$ which means that the pixel i is inclined to assign o_1. Therefore, when hands are totally visible in the same scene, spatial term can distinguish each hand. Nevertheless, when hands overlap together ($o_1 \cap o_2 = \phi$), it will be ambiguous to assign the pixel i to o_1 or o_2 in Figure 4(b). This means that spatial term ε_C is suitable for $o_1 \cup o_2 - o_1 \cap o_2$.

3.3.2. Motion Constraint. In (8), the energy function does not consider the situation in which hands pass over other skin-colored objects, such as face. Therefore, a new energy term called motion term ε_M is given to handle this situation:

$$\varepsilon_M(\lambda, t) = \omega_M \sum_{i\in I}\sum_{n=1}^{N} \beta\left(\vec{v}_n^t\right)(1 - \delta(\lambda(i), n)), \tag{16}$$

where $\omega_M > 0$ is a weight parameter. The function β is defined as

$$\beta\left(\vec{v}_n^t\right) = \exp\left(\frac{-\vec{v}_n^t * \vec{v}_n^t}{\rho^2}\right), \tag{17}$$

where ρ is the motion parameter mentioned in (13).

Using the motion information allows to reject some bad segmentations in the case of hands over skin-colored objects. When a pixel i is from $o_{n_1}^{t|t-1}$ with the velocity $\vec{v}_n^t$, it assigns $\varepsilon_M(n_1, t) = 0$ to $o_{n_1}^{t|t-1}$ and the value $\varepsilon_M(n_i, t) > 0$ to the other sets $o_{n_i}, i \neq 1$ according to (16). $\varepsilon_M(n_1, t) < \varepsilon_M(n_i, t)$ means that the pixel i is intended to assign the n_1th object. The motion term can keep good segmentation when hands and other skin-colored objects overlap (e.g., hands over face).

3.3.3. Chamfer Distance. The above defined terms are based on motion information and the prediction set $o_n^{t|t-1}$. However, spatial and motion terms still cannot deal with hand occlusions (see Figure 4(b)). Therefore, a new term called chamfer term $\varepsilon_{\mathrm{Ch}}$ is introduced to deal with hand occlusions. $\varepsilon_{\mathrm{Ch}}$ is defined as

$$\begin{aligned}\varepsilon_{\mathrm{Ch}}(\lambda, t) &\\ &= \omega_{\mathrm{Ch}} \sum_{(p,q)\in w^t, \lambda_p \neq \lambda_q} (1 - \exp(-\zeta(i)))\left(1 - \delta\left(\lambda_p, \lambda_q\right)\right),\end{aligned} \tag{18}$$

where $\zeta(\cdot)$ is the function of chamfer distance transform and $\omega_{\mathrm{Ch}} > 0$ is the weight parameter. Before computing the chamfer distance, we should get the binary image from the frame I^t at time t (e.g., using canny edge detection [23]). Then the value of chamfer distance can be fast calculated in two passes over the frame [24] as shown in Figure 5. $\varepsilon_{\mathrm{Ch}}$ encourages to keep discontinuous in the image boundary. In particular, when hands overlap, we can set a large value to ω_{Ch} for rejecting bad segmentation in the areas of occlusion $o_1 \cap o_2$.

3.4. Final Energy Function. We merge all of the mentioned terms. Therefore, the hand tracking problem consists of six terms to minimizing the following energy function:

$$\begin{aligned}\varepsilon(\lambda, t) = {} & \varepsilon_D(\lambda, t) + \varepsilon_S(\lambda, t) + \varepsilon_T(\lambda, t) \\ & + \varepsilon_C(\lambda, t) + \varepsilon_M(\lambda, t) + \varepsilon_{\mathrm{Ch}}(\lambda, t).\end{aligned} \tag{19}$$

Compared with the energy function equations (4) and (8), our model can handle hand occlusions, hands over face, and fast hand capture. After building the graph by (19), we can apply the α-expansion algorithm [16] to minimize the energy function.

3.5. Overview of the Proposed Method. We have described the principle of our method to track and segment hands in different circumstances. We use four steps to achieve hands tracking and segmentation. At first, initialization segmentations for all tracked hands are provided by manual operation

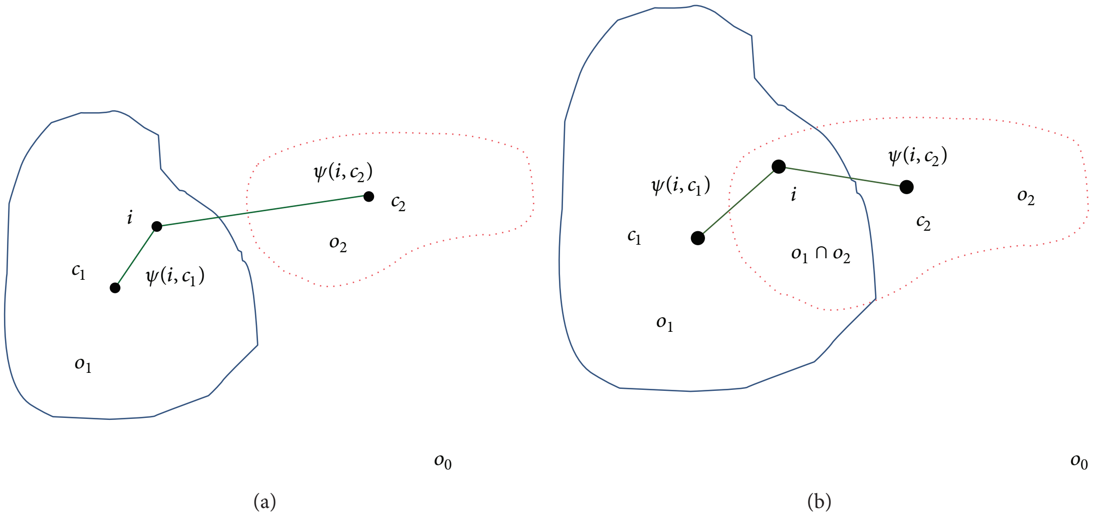


FIGURE 4: Illustration of computing function $\psi(\cdot)$.

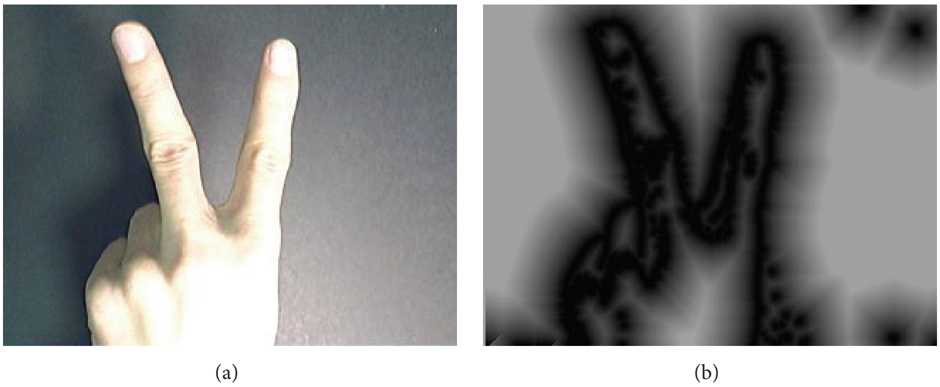


FIGURE 5: (a) Source image and (b) result by chamfer distance transform.

at time $t = 0$. Then at time $t > 0$, the prediction $o_n^{t|t-1}$ can be estimated by the dynamic model. Later on, we construct the graph by the augmented graph cuts and use α-expansion to obtain final segmentation results. Finally, we judge whether the number of interest points is larger than a given threshold. If the number of interest points is below a given threshold, we can resample the interest points. An overview of our algorithm is given in Algorithm 1.

4. Experimental Results

To validate and evaluate the proposed approach, we afford four videos (three videos were captured by our webcam and one video is an American sign language (ASL) video provided by Purdue ASL database [25]). All the videos have the same frame rate with 30 fps. In this paper, we only provided four challenge videos, but more results (e.g., four hand tracking and segmentation) can be found in the website: http://joewan.weebly.com/my-research.html.

4.1. Results. The proposed method is implemented in Microsoft Visual Studio 2008. All the videos we have offered are tested on a Core 2 Duo P8600 Processor with 2 GB RAM. The initialization segmentations (at time $t = 0$), the tracking results, and the different parameters are given in our experiments. Every tracked hand is labelled with different color. Although there are some methods which are similar to us, we only compare the proposed approach with the methods [12]. That is because the methods [13] require a reference background image for background subtraction to obtain external observations. It is not suitable to hand track in dynamic background. Papadakis and Bugeau [1] proposed a framework for object tracking. But the method [1] has a strong assumption that the occluded part of an object is a subset of the prediction of the whole object, which is not appropriate for self-occlusions that commonly happen on hands motion, especially fingers movement. To compare with the methods [12], the parameters ω_C, ω_M, and ω_{Ch} are set to zero, as they can recover the original energy function equation (8).

4.1.1. Hand Occlusions. This video has 141 frames and the frame size is $320 * 240$ pixels, which shows that two hands may be overlapped when both hands are in motion. It is called

```
Step 1. Initialization (at time $t = 0$)
(i) $N$: the number of tracked hands
(ii) $\tau_1$: the displacement of one interesting point from time $t - 1$ to $t$
(iii) $\tau_2$: minimum number of interesting points
(iv) Manually Initialize the sets $O_n^0, n = 1, 2, \ldots, N$ (such as at time $t = 0$ in Figures 6–12)
    $O_{n1}^0 \cap O_{n2}^0 = \phi, n1, n2 = 1, \ldots, N, n1 \neq n2$.
(v) Find interesting points $f_{nm}^0, m = 1, \ldots, m_f$ in $O_n^0$ via good-feature-to-track [7, 21].
    For at time $t = 1, \ldots, T$
Step 2.
(i) Find interesting points $f_{nm}^t$ using $f_{nm}^0$ via optical flow [22].
(ii) If $d(f_{nm}^t, f_{nm}^{t-1}) = \| f_{nm}^t - f_{nm}^{t-1} \| < \tau_1$ obtain the final interest points ${f_{nm}^t}'$.
(iii) Compute the hands mean velocity $v_n^{\bar{t}}, n = 1, \ldots, N$ using ${f_{nm}^t}'$ via (11).
(iv) Predict the sets $\mathrm{o}_n^{t|t-1}, n = l, \ldots, N$ using $v_n^{\bar{t}}$.
Step 3.
    Build the graph and apply $\alpha$-expansion via (19).
Step 4.
    If $m_f < \tau_2$ (the number of interest points below a given threshold $\tau_2$)
    Update interesting points in the region $\mathrm{o}_n^{t|t-1}$ via good-feature-to-track [7, 21].
    If $t < T$
    Return to Step 2.
```

Algorithm 1: An overview of the proposed algorithm.

video 1. The parameters of our method are as follows: $\omega_S = 10$, $\omega_T = 2.8$, $\omega_C = 0.2$, $\omega_M = 1.1$, and $\omega_{\text{Ch}} = 5$. And the method [12] parameters are as follows: $\omega_S = 10$, $\omega_T = 2.8$, $\omega_C = 0$, $\omega_M = 0$, and $\omega_{\text{Ch}} = 0$.

In Figure 6, let me firstly analyze the results which are shown in the first row by the method [12]. We can see that the two hands are labelled green color at $t = 20$, which means that the right hand is wrongly segmented. Additionally, when two hands are partially overlapped at $t = 66$, the left hand fails to track. Nevertheless, the hands are well recovered after hand occlusions by our method as shown in the second row of Figure 6 which shows that our approach is able to solve two principal problems: dealing with hand occlusions and rejecting oversegmentation.

4.1.2. Hand over Face. Now we give an example to demonstrate that our method can achieve hand segmentation even though hands pass over skin-colored objects, such as face. The video called video 2 is recorded outdoors including 106 frames. The frame size is 640 ∗ 480 pixels. Our parameters are as follows: $\omega_S = 5$, $\omega_T = 1.8$, $\omega_C = 1.5$, $\omega_M = 1.8$, and $\omega_{\text{Ch}} = 2$. The parameters of the method [12] are as follows: $\omega_S = 5$, $\omega_T = 1.8$, $\omega_C = 0$, $\omega_M = 0$, and $\omega_{\text{Ch}} = 0$.

As shown in Figures 7 and 8, when the hands move from the left to the right, hand over face occlusion occurs at time $t > 6$. Figure 7 shows the results by the method [12], which reveals the failure to accurately track and segment hand when hands pass over face. In Figure 8, the hand segmentation is quite well achieved along the sequence by our method. Owing to the motion constraint in (19), when the hands pass over the face, our method still can reject the bad segmentations which may occur in the face region.

4.1.3. Fast Hand Tracking in Sign Language. The video called video 3 is from Purdue ASL database [25]. It involves fast hand motion (entire frames), hand over face ($t = 216$), partly hand occlusions ($t = 182, 199$), and hand shape deformation (the entire video). This video includes 265 frames with the frame size of 640 ∗ 480 pixels. Our parameters are $\omega_S = 5$, $\omega_T = 1.8$, $\omega_C = 1.1$, $\omega_M = 0.8$, and $\omega_{\text{Ch}} = 2$. As shown in Figure 9, our method is robust to segment and track hands. From the results in Figures 6–8, the method [12] cannot deal with hand occlusions and hand over face. So we only give the results by our method.

4.1.4. Dynamic Background. In order to further evaluate the effectiveness of the proposed method under complex situations, we test our method in dynamic background. The video called video 4 was captured in lab environment including 174 frames with frame size 320 ∗ 240. The moving pedestrian as the dynamic background walk and happen occlusions when the tracked hand is in motion. The parameters of our method are as follows: $\omega_S = 5$, $\omega_T = 1.8$, $\omega_C = 0.1$, $\omega_M = 1.0$, and $\omega_{\text{Ch}} = 3.9$. Final results are given in Figure 10 which displays that good performance can be achieved in dynamic background by our method.

4.2. Discussion and Adjusting Parameters. The energy minimizing function in (19) is composed of six different terms. It has eight parameters to be tuned (three dynamic model parameters, five graph cuts parameters). However, most of them can be fixed in our experiments. For dynamic model parameters, the three parameters are given constant value because those parameters are not sensitive to our model. We set $\tau_1 = 80$, $\tau_2 = 140$, and $\rho = 3.5$.

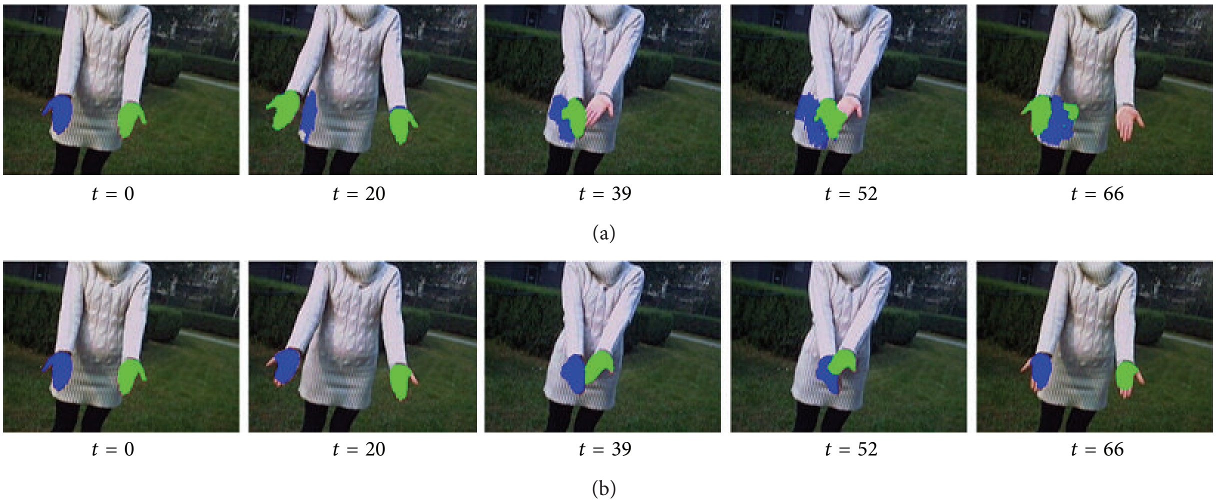


Figure 6: (a) Results by the method [12]. (b) Results by the proposed method (initialization at time $t = 0$).

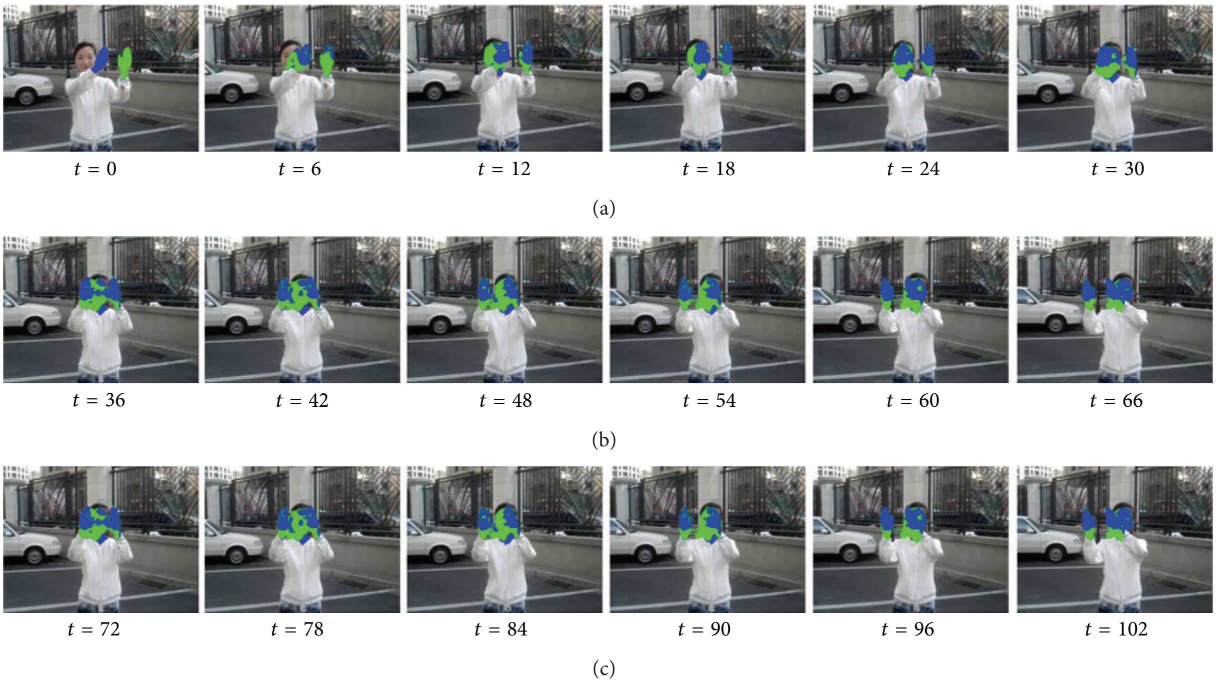


Figure 7: Result by [26] (initialization at time $t = 0$).

Here, we give some hints about adjusting parameters to help use the proposed method. The spatial parameter ω_C denotes the weight value to every pixel of image, which measures the distance from the location of a pixel to the centroid of every tracked hand. When hands are disjoint, ω_C can be set to a large value. If ω_C is set to infinity, the model will only consider the separated areas of the tracked objects. The motion parameter ω_M represents the weight for handling hands over face. When the hand passes over skin-colored object, a large value can be set to ω_M (see Section 4.1.2). The chamfer parameter ω_{Ch} indicates the weight for hand occlusions. When hands overlap, a large value can be set to ω_{Ch} (see Section 4.1.2). These three parameters ($\omega_C, \omega_M, \omega_{\text{Ch}}$) often vary from 0 to 6.

The parameter ω_S makes everywhere more smoothly everywhere [16]. If the value of ω_S is set to be large, it will lead to poor results at object boundaries. In most circumstances, ω_S usually varies from 5 or 10 which is well suitable for the model according to the experimental results. It is certain that the small value ω_T allows [12] to search farther from the prediction and well track the object. When a small value is set to ω_T, oversegmentations may occur. For instance, as

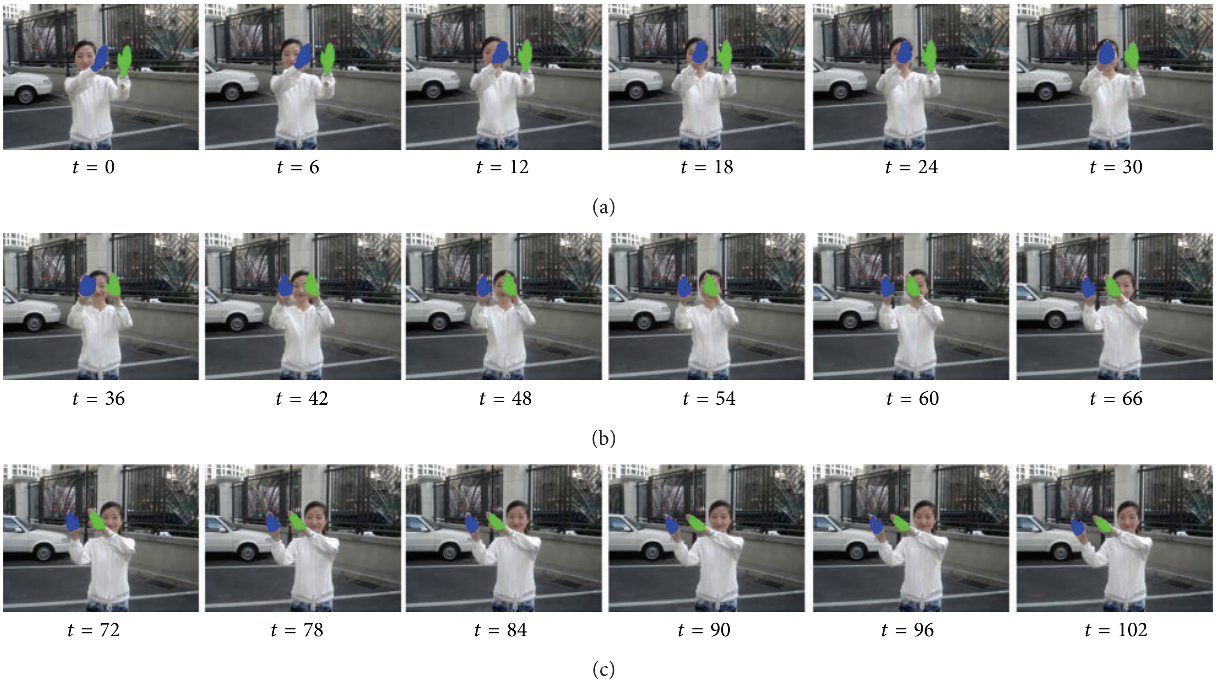


Figure 8: Result by our method (initialization at time $t = 0$).

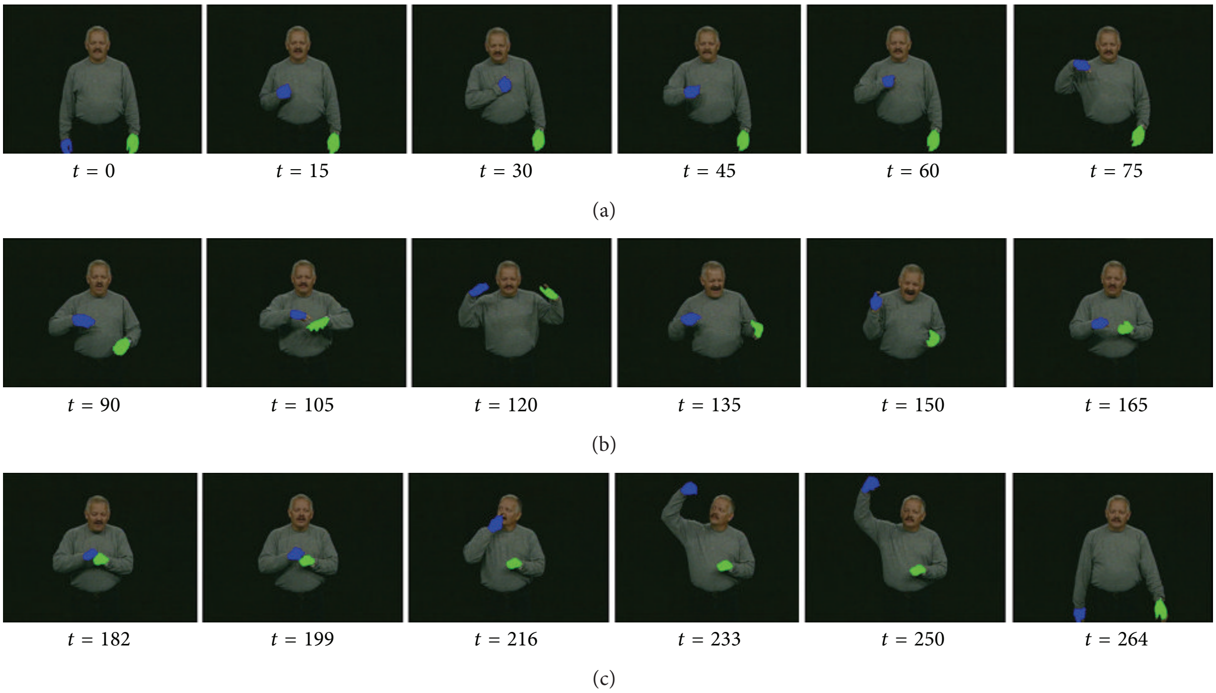


Figure 9: Result by our method (initialization at time $t = 0$).

illustrated in Figure 11, when we set $\omega_T = 0.8$ and other parameters are the same as Section 4.1.3, we can see that oversegmentations happen at time $t = 118, 219$.

4.3. Evaluation and Complexity Analysis. In order to perform objective comparison, we first manually segment hand mask (ground truths) for every frame in our test videos. Then we calculate mean percentage error (MPE) [27] between ground truths and segmentation results. MPE p_ε is defined as

$$p_\varepsilon = \frac{1}{n}\sum_{i=1}^{n}\frac{N_\varepsilon}{F}, \qquad (20)$$

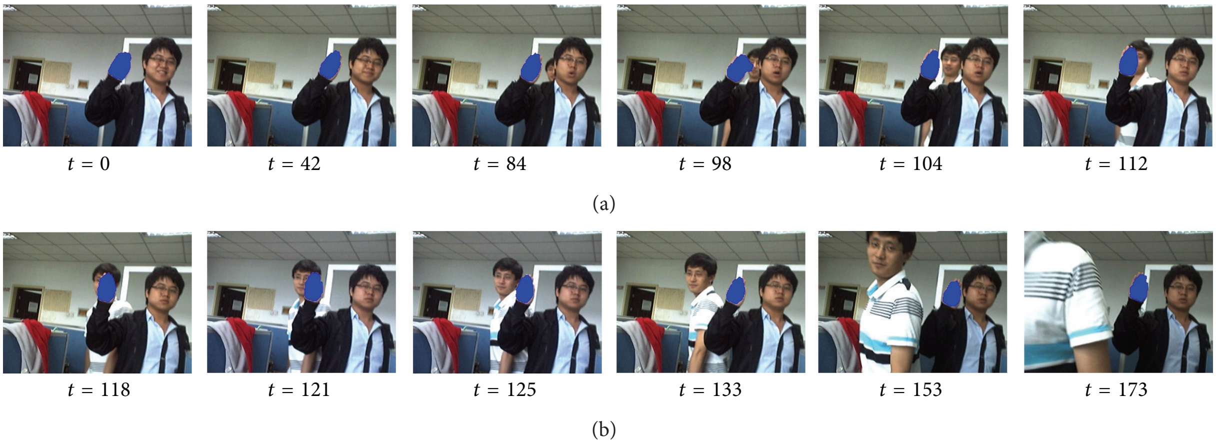


Figure 10: Result by our method (initialization at time $t = 0$).

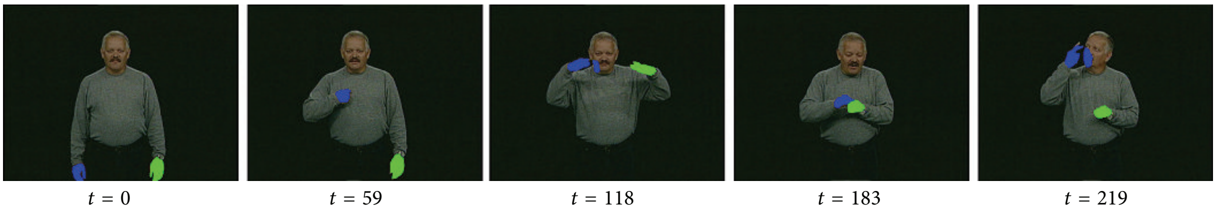


Figure 11: Results with a small value ω_T (initialization at time $t = 0$).

where N_ε denotes the number of false detected pixels, F represents the frame size, and n is the number of frames in a video. Note that the false detection happens in two situations: (1) pixels in the background are detected as the hand region; (2) pixels in one hand is treated as other hands or background. Figure 12(a) shows MPEs for four videos. As shown in Figure 12(a), we can get two conclusions as follows.

(i) When hands and other skin-colored objects are in the same scene (hand over face, hand occlusions), MPEs by our method are much lower than the method [12]. In particular, the MPEs of videos 3 and 4 by the method [12] are very high (>10%) due to wrong labeled face region, when hands and the face are overlapped.

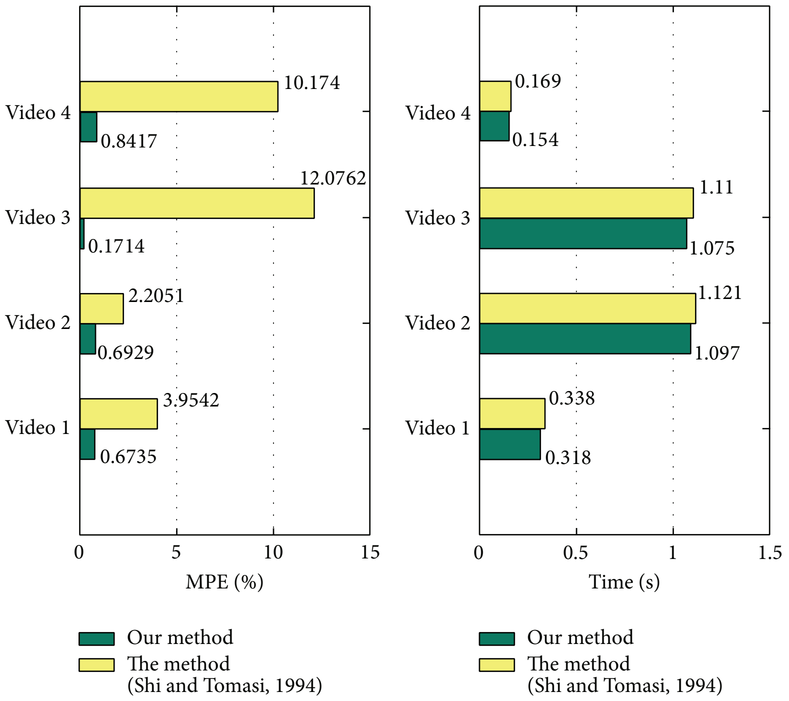
(ii) The MPE (0.1714%) of video 4 by our method is close to ground truth (0%), which proves that the proposed method is well suitable for hands tracking and segmentation in sign language video.

Next, we give the running times of both the proposed method and the method [12] as shown in Figure 12(b) where the average execution time (AET) for every frame in all test videos is given. We can see that the AETs by our method are approximate to the method [12], although the truth is AETs by the proposed approach is slight higher that the method [12] about 20 to 30 milliseconds pre frame. That is because additional terms ($\omega_C, \omega_M, \omega_{\text{Ch}}$) are incorporated into energy function, which leads to a slight high complexity. Meanwhile, the proposed method can successfully track and segment hands when the face and hands are partly occluded. AET depends on the frame size and the number of tracked number (see Figure 12(b)). In our future research, we will consider a narrow band around the prediction sets [10] to decrease the computational cost. The study of this band will be the subject of future works for real-time purpose.

5. Conclusion

In this paper, we present a method based on augmented graph cuts and the dynamic model for hand tracking and segmentation in different environments. The proposed algorithm can resolve three problems: fast hand motion capture, hand occlusions, and hand over face. In our method, we reformulate the energy function by adding some new energy terms which are more robust to hand tracking and segmentation. Additionally, the new terms can deal with occlusions and obtain accurate segmentation.

Meanwhile, there are a lot of perspectives that can be improved. At first, we can develop a method to automatically extract hand region instead of manually segmented hands in initialization step. For instance, we can apply AdaBoost algorithm [26] to detect the region of interest (ROI) of hands and use grab cut [28] in ROI to achieve hand segmentation. Second, some prior knowledge can be incorporated into the proposed method to handle totally occlusion. Moreover, another important point is the tuning of the parameters in energy function. In our future research, we will focus on these problems.



(a)

(b)

Figure 12: Comparisons of the methods on MPE and AET.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported partly by the National Natural Science Foundation of China (61172128), National Key Basic Research Program of China (2012CB316304), New Century Excellent Talents in University (NCET-12-0768), the Fundamental Research Funds for the Central Universities (2013JBZ003), Program for Innovative Research Team in University of Ministry of Education of China (IRT201206), Beijing Higher Education Young Elite Teacher Project (YETP0544), and Research Fund for the Doctoral Program of Higher Education of China (20120009110008).

References

[1] N. Papadakis and A. Bugeau, "Tracking with occlusions via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 1, pp. 144–157, 2011.

[2] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Computing Surveys*, vol. 38, no. 4, pp. 1–45, 2006.

[3] V. I. Pavlovic, R. Sharma, and T. S. Huang, "Visual interpretation of hand gestures for human-computer interaction: a review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 677–695, 1997.

[4] J. Han, G. Awad, and A. Sutherland, "Automatic skin segmentation and tracking in sign language recognition," *IET Computer Vision*, vol. 3, no. 1, pp. 24–35, 2009.

[5] M.-H. Yang, N. Ahuja, and M. Tabb, "Extraction of 2D motion trajectories and its application to hand gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1061–1074, 2002.

[6] B. Stenger, A. Thayananthan, P. H. S. Torr, and R. Cipolla, "Model-based hand tracking using a hierarchical bayesian filter," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1372–1384, 2006.

[7] M. Kolsch and M. Turk, "Fast 2d hand tracking with flocks of features and multi-cue integration," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, pp. 158–165, 2004.

[8] M. Isard and A. Blake, "Condensation-conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[9] C. Shan, T. Tan, and Y. Wei, "Real-time hand tracking using a mean shift embedded particle filter," *Pattern Recognition*, vol. 40, no. 7, pp. 1958–1970, 2007.

[10] N. Xu and N. Ahuja, "Object contour tracking using graph cuts based active contours," in *Proceedings of the International Conference on Image Processing (ICIP'02)*, pp. III/277–III-280, September 2002.

[11] D. Freedman and M. W. Turek, "Illumination-invariant tracking via graph cuts," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 10–17, June 2005.

[12] J. Malcolm, Y. Rathi, and A. Tannenbaum, "Multi-object tracking through clutter using graph cuts," in *Proceedings of the IEEE*

11th International Conference on Computer Vision (ICCV '07), October 2007.

[13] A. Bugeau and P. Pérez, "Track and cut: simultaneous tracking and segmentation of multiple objects with graph cuts," *Eurasip Journal on Image and Video Processing*, vol. 2008, Article ID 317278, 2008.

[14] M. Stamm and K. J. R. Liu, "Live video object tracking and segmentation using graph cuts," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '08)*, pp. 1576–1579, October 2008.

[15] J. Wan, Q. Ruan, G. An, and W. Li, "Hand tracking and segmentation via graph cuts and dynamic model in sign language videos," in *Proceedings of the IEEE 11th International Conference on Signal Processing*, pp. 1135–1138, October 2012.

[16] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proceedings of the 8th International Conference on Computer Vision (ICCV '01)*, vol. 1, pp. 105–112, Vancouver, Canada, July 2001.

[17] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[18] L. Yatziv, A. Bartesaghi, and G. Sapiro, "O(N) implementation of the fast marching algorithm," *Journal of Computational Physics*, vol. 212, no. 2, pp. 393–399, 2006.

[19] H. Li, K. N. Ngan, and Q. Liu, "FaceSeg: automatic face segmentation for real-time video," *IEEE Transactions on Multimedia*, vol. 11, no. 1, pp. 77–88, 2009.

[20] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the 7th IEEE International Conference on Computer Vision (ICCV '99)*, vol. 2, pp. 1150–1157, Kerkyra, Greece, September 1999.

[21] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 593–600, June 1994.

[22] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *International Joint Conference on Artificial Intelligence*, vol. 81, pp. 674–679, 1981.

[23] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

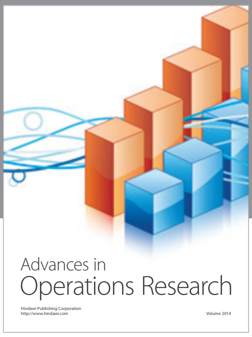
[24] G. Borgefors, "Hierarchical Chamfer matching: a parametric edge matching algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 6, pp. 849–865, 1988.

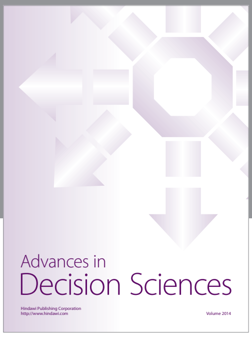
[25] A. M. Martinez, R. B. Wilbur, R. Shay, and A.C. Kak, "Purdue RVL-SLLL ASL database for automatic recognition of American Sign Language," in *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces (ICMI '02)*, pp. 167–172, 2002.

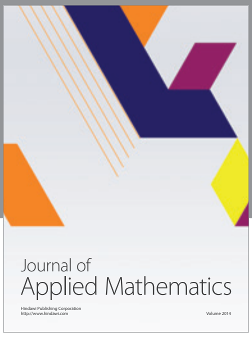
[26] R. Lienhart and J. Maydt, "An extended set of Haar-like features for rapid object detection," in *Proceedings of the International Conference on Image Processing (ICIP '02)*, vol. 1, pp. 900–903, September 2002.

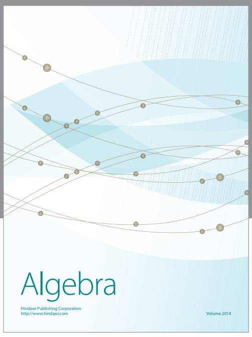
[27] D. L. Waller, *Operations Management: A Supply Chain Approach*, Cengage Learning Business Press, 2003.

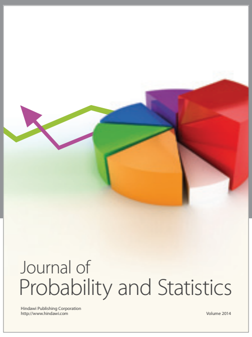
[28] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.


Advances in
Operations Research

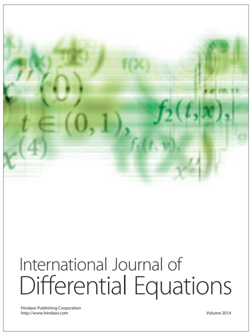

Advances in
Decision Sciences


Journal of
Applied Mathematics

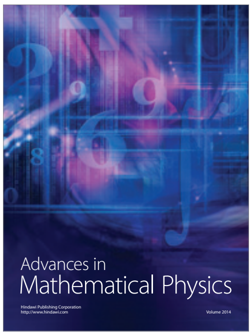

Algebra

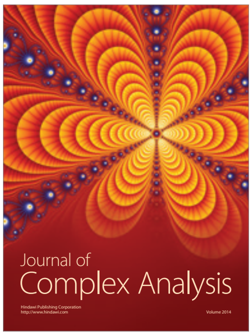

Journal of
Probability and Statistics

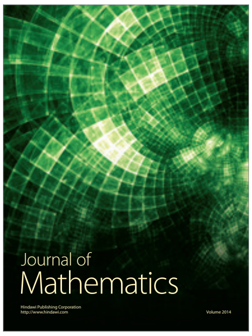
The Scientific
World Journal

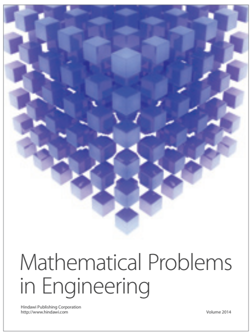

International Journal of
Differential Equations

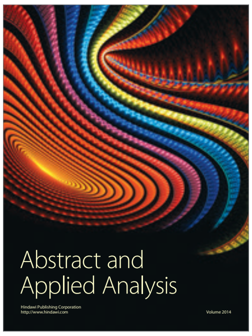
International Journal of
Combinatorics

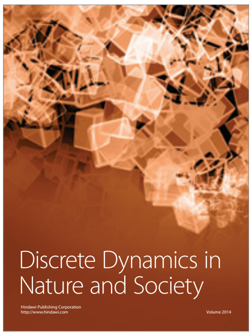

Advances in
Mathematical Physics

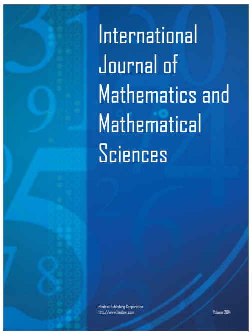

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

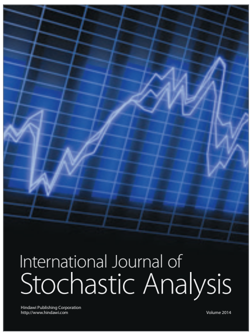

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization