Occluded Human Body Segmentation and its Application to Behavior Analysis

Jun-Wei Hsieh[1], Shin-Yu Chen[2], Chi-Hung Chuang[3], Chang-Yu Huang[2]

Dept. of Computer Science and Engineering, National Taiwan Ocean University, Taiwan
shieh@notu.edu.tw

Dept. of Electrical Engineering, Yuan Ze University, Taiwan
s958507@mail.yzu.edu.tw

Dept. of Learning and Digital Tech., Fo Guang University, Taiwan
chi_hung_chuang@hotmail.com

***Abstract*—This paper addresses the problem of occluded human segmentation and then uses its results for human behavior recognition. To make this ill-posed problem become solvable, a novel clustering scheme is proposed for constructing a model space for posture classification. To construct the model space, we use a triangulation-based method to divide a posture into different triangular meshes from which a posture descriptor "centroid context" is then extracted for posture recognition and model selection. Then, a model-driven approach can be proposed for separating an occluded region to individual objects from the model space. Due to partial occlusions, the task of model selection is very challenging. For reducing the model space, a particle filtering technique is then used for locating possible positions of each occluded object. Then, from these positions, the best model of each occluded object can be then selected using its distance maps. Then, a novel template re-projection technique is proposed for repairing an occluded object to a complete one. Then, each action sequence can be converted to a series of symbols through posture analysis. Since occluded objects are handled, there will be many posture symbol converting errors in this representation. Instead of using a specific symbol, we code a posture using not only its best matched key posture but also its similarities among other key postures. Then, recognition of an action taken from occluded objects can be modeled as a matrix matching problem. With the matrix representation, different actions (even caused by occluded persons) can be more robustly and effectively matched by comparing their Kullback–Leibler distance. Experimental results show the effectiveness and superiority of the proposed method in classifying human behaviors from occlude objects.**

***Index Terms*—Occluded Object Segmentation, Behavior analysis, K-L distance .**

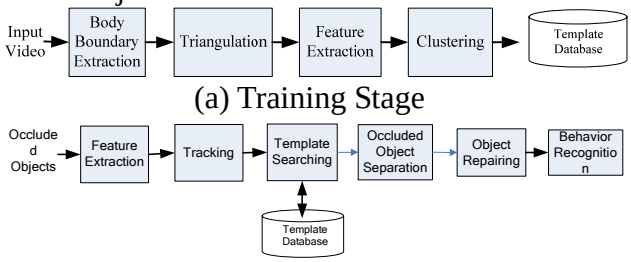
I. INTRODUCTION

Human behavior analysis can be applied in a variety of application domains such as video surveillance, video retrieval, human-computer interaction systems, and medical diagnoses. In the past, many approaches [5] , [7] , [8] have been proposed for video-based human movement analysis. A visual surveillance system to model and recognize human behavior using HMMs (Hidden Markov Models) in [7] and a trajectory feature. Rosales and Sclaroff [8] proposed a trajectory-based recognition system to detect pedestrians in outdoor environments and recognize their activities from multiple views based on a mixture of Gaussian classifiers. In [9] , Pynadath *et al.* have considered human actions as a complex hierarchy of events ranked where lower levels contain shorter actions that combine temporally to form higher level events/actions which are more abstract. In [10] , Wren *et al.* proposed a Pfinder system for tracking and recognizing human behavior based on a 2-D blob model. The challenge of incorporating 2-D posture models into the analysis of human behavior is the possibility of ambiguity between the adopted models and real human behavior, which may have mutual occlusions between body parts or lose clothes. In these circumstances, although it is well-known that the cardboard model [11] is good for modeling articulated human motions, the prerequisite that body parts must be well segmented makes this model inappropriate for real-time analysis of human behaviors.

In addition to the above approach, some papers discuss human behavior analysis in the condition of occlusion. The common approach to track objects is to use background subtraction and establish correspondence from frame to frame to find the track of the object[14] . An alternative approach

to background subtraction is to find the transformation of the object which is modeled using simple geometric models, e.g., ellipse or rectangle. Using the mean-shift approach to compute the translation of a circular region was addressed in[15] . Peursum *et al.* [13] present a method for finding and classifying objects within real-world scenes by using the activity of humans interacting with these objects to infer the object's identity.

In this paper, we address the problem of occluded human segmentation and then use its results for human behavior recognition. Since this segmentation problem is still ill-posed, this paper assumes that the objects have been observed some periods before they are occluded. The detailed components of the system are shown in Fig. 1. First of all, before objects are occluded, the training stage (shown in Fig. 1 (a)) is applied for constructing the model space. Then, the model spaces of these objects can be constructed for well separating the occluded object to different ones. To construct the model space, we use a triangulation-based method to divide a posture into different triangular meshes. THen, a posture descriptor "centroid context" is then extracted for posture recognition. With this descriptor, a novel key posture selection scheme is then proposed for constructing a model space. To select the best model from this model space for guiding the segmentation process, a tracking technique is then adopted for roughly detecting possible locations of each occluded object. Then, the distance transform is used for finely matching occlude objects according to their partial edges. After model selection, a recovering scheme is then proposed for repairing an occluded object to a complete one. Then, each action sequence can be converted to a series of symbols through posture analysis. Since occluded objects are handled, there will be many posture symbol converting errors in this representation. Instead of using a specific symbol, we code a posture using not only its best matched key posture but also its similarities among other key postures. Then, recognition of an action taken from occlude objects can be modeled as a matrix matching problem. With the matrix representation, different actions (even occluded) can be more robustly and effectively matched by comparing their Kullback–Leibler distance. Experimental results show the effectiveness and superiority of the proposed method in classifying human behaviors from occlude objects.



(a) Training Stage

(b) Segmentation Stage

Fig. 1: Flowchart of the proposed system to recognize human behaviors from occluded objects. (a) Training stage. (b) Recognition stage.

The remainder of the paper is organized as follows. Details of posture descriptor are described in Section II. The scheme of model construction is discussed in Section III. Section IV discusses the techniques of object tracking and model selection. Section V describes details of occlude object segmentation. Then, a novel behavior recognition scheme from occluded objects is proposed in Section VI. The experiment results are given in Section VII. We then present our conclusions in Section VIII.

II. Posture Representation Using Centroid Contexts

This paper assumes that all the analyzed video sequences are captured by a still camera. Then, different human postures can be detected through background subtraction. Then, the descriptor "centroid contexts" will be extracted from each foreground object for posture representation and classification. In what follows, details of this technique will be described.

A. Skeleton Extraction Using Triangulation

Assume that P is a binary posture and extracted through background subtraction. This paper uses the technique of constrained Delaunay triangulation [1] to divide P to different triangle meshes. Then, according to the result of triangulation, a

graph can be formed by connecting all the centroids of any two connected meshes in *P*. This section will use the depth-first search technique to find the skeleton that will be used for posture recognition.

Assume Ω_P is the set of triangular meshes extracted from *P*, i.e., $\Omega_P = \{T_i\}_{i=0,1,\ldots,N_{T_P}-1}$. Each triangle mesh T_i in Ω_P has a centroid C_{T_i}. Two triangular meshes, T_i and T_j, are connected if they share one common edge. Then, based on this connectivity, *P* can be converted into an undirected graph G_P, where all centroids C_{T_i} in Ω_P are nodes on G_P; and an edge exists between C_{T_i} and C_{T_j} if T_i and T_j are connected. The degree of a node on the graph is defined by the number of edges connected to it. Thus, based on the above definitions, we perform a graph search on G_P to extract the skeleton of *P*.

To derive a skeleton based on a graph search, we seek a node *H* whose degree is one and whose position is the highest among all the nodes on G_P. *H* is defined as the head of *P*. Then, we perform a depth-first search from *H* to construct a spanning tree in which all the leaf nodes L_i correspond to different limbs of *P*. The branching nodes B_i (whose degree is three in G_P) are the key points used to decompose *P* into different body parts such, as the hands, feet, or torso. Let C_P be the centroid of *P*, and let *U* be the union of *H*, C_P, L_i, and B_i. The skeleton S_P of *P* can be extracted by linking any two nodes in *U* if they are connected. Using the above linking strategy, a path can be easily found from the spanning tree of *P*. Note that, the spanning tree of *P* obtained by depth-first search is also a skeleton.

B. Centroid Context

In this section, we introduce a centroid context-based shape descriptor that can characterize the interior of a shape. This descriptor is used in the fine search process. Since the triangulation results of human postures vary, we calculate the distribution of every posture based on the relative positions of the meshes' centroids. A descriptor of this form guarantees robustness and compactness. Assume all postures are normalized to a unit size. Then, similar to the technique used in shape context [2] , we project a sample onto a log-polar coordinate and label each mesh. We use *m* to represent the number of shells used to quantize the radial axis and use *n* to represent the number of sectors that we would like to quantize each shell. Therefore, the total number of bins used to construct the centroid context is $m \times n$. For the centroid *r* of a triangle mesh of a posture, we construct a vector histogram $h_r = (h_r(1), \ldots, h_r(k), \ldots, h_r(mn))$, in which $h_r(k)$ is the number of triangle mesh centroids in the *k*th bin by considering *r* as the origin, i.e.,

$$h_r(k) = \#\{q \mid q \neq r,\ (q - r) \in bin^k\}, \qquad (1)$$

where bin^k is the *k*th bin of the log-polar coordinate. Then, given two histograms $h_{r_i}(k)$ and $h_{r_j}(k)$, the distance between them can be measured by a normalized intersection:

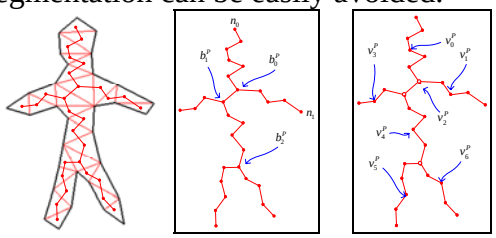
$$C(r_i, r_j) = 1 - \frac{1}{N_{mesh}} \sum_{k=1}^{K_{bin}} \min\{h_{r_i}(k), h_{r_j}(k)\}, \qquad (2)$$

where K_{bin} is the number of bins and N_{mesh} denotes the number of meshes calculated from a posture. Using Eqs.(1) and(2), we can define a centroid context to describe the characteristics of an arbitrary posture *P*.

Fig. 2: Polar Transform of a human posture.

In the previous section, we presented a tree search algorithm that can be used to find a spanning tree T_{dfs}^P from a posture P based on the triangulation result. As shown in Fig. 3, (b) is the spanning tree derived from (a). The tree T_{dfs}^P captures the skeleton feature of P. Here, we call a node a branch node if it has more than one child. By this definition, there are three branch nodes in Fig. 3(b), i.e., b_0^P, b_1^P, and b_2^P. If we remove all

the branch nodes from T_{dfs}^P, it will be decomposed into different branch paths $path_i^P$. Then, by carefully collecting the set of triangle meshes along each path, it is clear that each path $path_i^P$ will correspond to one body component in P. For example, in Fig. 3(b) if we remove b_0^P from T_{dfs}^P, two branch paths will be formed, i.e., one from node n_0 to b_0^P and one from b_0^P to node n_1. The first path corresponds to the head and neck of P, and the second corresponds to the hand of P. In some cases, the path may not correspond to a high-level semantic body component exactly, as shown by the path from b_0^P to b_1^P. However, if the length of a path is further constrained, over-segmentation can be easily avoided.



(a) (b) (c)

Fig. 3: Body component extraction. (a) Triangulation result of a posture. (b) The spanning tree of (a). (c) Centroids derived from different parts (are determined by removing all the branch nodes).

Given a path $path_i^P$, we can collect a set of triangle meshes V_i^P along it. Let c_i^P be the centroid of the triangle mesh closest to the center of this set of triangle meshes. As shown in Fig. 3(c), c_i^P is the centroid extracted from the path that begins at n_0 and ends at b_0^P. Given a centroid c_i^P, we can obtain its corresponding histogram $h_{c_i^P}(k)$ using Eq.(1). Assume that the set of these path centroids is V^P. Based on V^P, the centroid context of *P* is defined as follows:

$$P = \{h_{c_i^P}\}_{i=0,\ldots,|V^P|-1}, \quad (3)$$

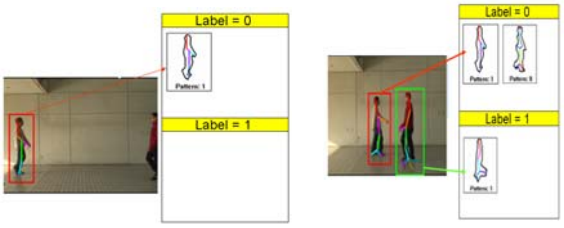
where $|V^P|$ is the number of elements in V^P. Given two postures *P* and *Q*, the distance between their centroid contexts is measured by

$$d_{cc}(P,Q) = \frac{1}{2|V^P|}\sum_{i=0}^{|V^P|-1} w_i^P \min_{0\le j<|V^P|} C(c_i^P, c_j^Q) + \frac{1}{2|V^Q|}\sum_{j=0}^{|V^Q|-1} w_j^Q \min_{0\le i<|V^Q|} C(c_i^P, c_j^Q), \quad (4)$$

where w_i^P and w_j^Q are the area ratios of the *i*th and *j*th body parts residing in *P* and *Q*, respectively.

III. Model Construction

The occlusion problem is still ill-posed for computer vision method. To tackle this problem, this paper assumes that all the objects have been observed some frames before they are occluded or their model space has been constructed. Like Fig. 4, before occlusion, a series of postures can be collected and form a model space. Some postures may be redundant, so they should be removed from the modeling process. To do this, we use a clustering technique to select a set of key postures from a collection of real-world video clips. Then, a model-driven scheme can be proposed for well separating occluded objects to different individuals. In what follows, the scheme for model space construction is described.



(a) (b)

Fig. 4: Posture template construction. (a) Initial posture template. (b) Sets of posture templates selected from different frames.

Assume P_t is the posture extracted from the *t*th frame. Given two adjacent postures, P_{t-1} and P_t, the distance between them, d_t, is calculated using Eq.(4), where *w* is set 0.5. Assume T_d is the average value of d_t for all pairs of adjacent postures. For a posture P_t, if d_t is greater than $2T_d$, we define it a posture-change instance. By collecting all the postures that correspond to posture-change instances, we derive a set of key posture candidates S_{KPC}. However, S_{KPC} may still

contain many redundant postures, which would degrade the accuracy of the sequence modeling. To address this problem, we use a clustering technique to find a better set of key postures.

Initially, we assume each element e_i in S_{KPC} individually forms a cluster z_i. Then, given two cluster elements, z_i and z_j, in S_{KPC}, the distance between them is defined by:

$$d_{cluster}(z_i, z_j) = \frac{1}{|z_i||z_j|} \sum_{e_m \in z_i} \sum_{e_n \in z_j} Dist(e_m, e_n), \qquad (5)$$

where $Dist(.,.)$ is defined in Eq.(4) and $|z_k|$ represents the number of elements in z_k. According to Eq.(5), we can execute an iterative merging scheme to find a compact set of key postures from S_{KPC}. Let z_i^t and Z^t be the ith cluster and the collection of all clusters z_i^t respectively at the tth iteration. At each iteration, we choose a pair of clusters z_i^t and z_j^t whose distance $d_{cluster}(z_i, z_j)$ is the minimum among all pairs in Z^t, i.e.,

$$(z_i, z_j) = \arg\min_{(z_m, z_n)} d_{cluster}(z_m, z_n),$$

$$\text{for all } z_m \in Z^t,\ z_n \in Z^t,\ \textit{and}\ z_m \neq z_n.$$

If $d_{cluster}(z_i, z_j)$ is less than T_d, then z_i^t and z_j^t are merged to form a new cluster and thus constructing a new collection of clusters Z^{t+1}. The merging process is executed iteratively until no further merging is possible. Assume $\overline{Z}$ is the final set of clusters after merging. Then, from the ith cluster $\overline{z}^i$ in $\overline{Z}$, we can extract a key posture e_i^{key} so that

$$e_i^{key} = \arg\min_{e_m \in \overline{z}_i} \sum_{e_n \in \overline{z}_i} Dist(e_m, e_n). \qquad (6)$$

Based on Eq.(6) and checking all clusters in $\overline{Z}$, the set S_{KP} of key postsures, i.e., $S_{KP} = \{e_i^{key}\}$ can be constructed for human action sequence analysis.

IV. Object Tracking And Model Matching

The goal of this paper is to enhance the ability of a behavior analysis system for recognizing the behaviors between two persons even though they are occluded. Since the occlusion problem is still ill-posed in computer vision, a scheme of key posture selection has been proposed in the previous section for constructing a model space. Then, this section will present a novel method for choosing the best template model from the model space for tackling the occlusion problem.

A. Object Tracking Using Particle Filters

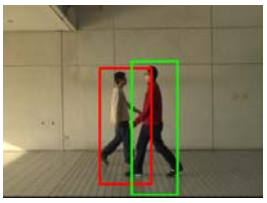
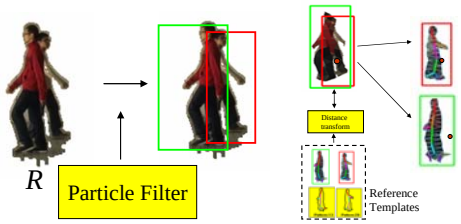


Fig. 5. The region is found by particle filters.



(a) (b)

Fig. 6: Flowchart of template model selection. (a) Tracking using particle filters. (b) Model selection using distance transform.

Let Z denote the set of template models. This paper presents a coarse-to-fine approach for finding the best model from Z. To avoid a full search on the whole image, at the coarse stage, the technique of particle filters is used for tracking each occluded objects. The particle filter algorithm [3] , [4] .is very similar to the mean-shift one but the difference is that the particle filter combines with Monte Carlo rule and condensation algorithm to filter out the new object position frame by frame. The former uses the sample sets to predict the object of candidates and the later involves a stochastic dynamic model to reserve the samples with high probability in the previous state. Then, each pedestrian will be bounded by a rectangle. Like Fig. 6, two rectangles with different colors were shown for denoting the tracked persons. From the tracking result, in what follows, different features (including contour, color, and centroid) will be used and integrated for finding the best model from Z more accurately.

B. Model Matching Using Triangulation

Given an occluded object O_k, the technique of particle filters can roughly locate its position. To obtain its best matching model from *Z*, this section will propose a triangulation-based scheme for tackling the occlusion problem.

(a) (b) (c)

Fig. 7: (a) An occluded region *R*. (b) Triangulation result of *R* using its contour feature. (c) Triangulation result of *R* using its edge feature.

Assume that *R* is the foreground region (extracted through background subtraction) in which different objects are occluded together. This scheme first over-segments *R* to different triangulation meshes using its edge features rather than its contours. Like Fig. 9(b), if the contour feature is used, it does not provide lots of triangular meshes for separating occluded objects to individual ones. However, if the triangulation task is done according to the edge feature, more useful meshes can be extracted for analyzing the inner structures of *R* (like Fig. 9(c)). Then, based on the triangulation results, a novel matching scheme will be proposed for tackling the occlusion task. Then, based on the triangulation results, a labeling technique will be presented for labeling each mesh so that the occluded pedestrians in *R* can be well separated for further behavior analysis. Thus, in this paper, the edges in *R* are used for guiding this over-segmentation process.

Assume that t_k is one triangulated mesh in *R* and there are *K* objects in *R*. Then, a label field $L = \{l \mid l \in [0,...,K\text{-}1]\}$ on *R* can be created. Thus, the segmentation problem can be converted to a labeling algorithm on *G*. Given t_k, its optimal label l_{t_k} can be estimated by maximizing a posteriori probability $P(l \mid t_k)$ as follows:

$$l_{t_k} = \arg\max_{l \in L} P(l \mid t_k).$$

Using the Bayesian rule, the posteriori probability can be further rewritten by

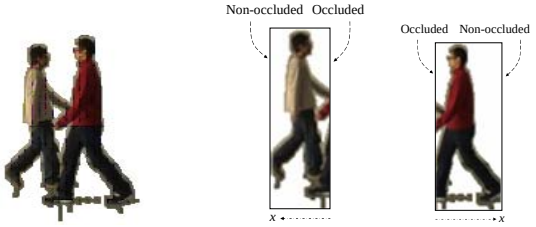
$$P(l \mid t_k) \propto p(t_k \mid l) P(l), \qquad (7)$$

where $P(l)$ is the priori probability of the *l*th object and $p(t_k \mid l)$ is the likelihood probability of t_k belonging to the *l*th object. According to the template and color models of t_k, $p(t_k \mid l)$ can be further decomposed to two components, i.e.,

$$p(t_k \mid l) = p_{template}(t_k \mid l)\, p_{color}(t_k \mid l),$$

where $p_{template}(t_k \mid l) = \exp(-d_{template}(t_k, l)/\sigma^2_{template})$ and $p_{color}(t_k \mid l) = \exp(-d_{color}(t_k, l)/\sigma^2_{color})$.

C. Distance Transform



(a) (b) (c)

Fig. 8: An occluded region and its two objects. (a) Occluded foreground region *R*. (b) and (c): Occluded objects in R.

Assume that *R* is a foreground region and there are *N* occluded objects O_k in it. Like Fig. 8, there are two occluded objects in *R*. For one object O_k in *R*, we can use the technique of particle filter for tracking its position and then extract it using a bounding box. (b) and (c) show the objects extracted by these two boxes. The occluded case shown in Fig. 8 is not serious. When the case shown in Fig. 6 is handled, the best template model should be selected from *Z* for more accurately segmenting each object O_k from *R*. The goal of template selection is achieved by transforming O_k to its corresponding distance map DT_{O_k}.

Given an object O_k, its distance map DT_{O_k} supplies each pixel of O_k with the distance to the nearest edge pixel. More accurately, the value of

a pixel r in DT_{O_k} is the shortest distance between it and all edge pixels in O_k, i.e.,

$$DT_{O_k}(r) = \min_{q \in edge(O_k)} d(r,q), \quad (8)$$

where $edge(O_k)$ denotes the edge map of O_k (obtained from Canny edge detector) and $d(r,q)$ is the Euclidian distance between r and q. The distance between the two distance maps of two objects O_k and D can be defined as follows:

$$d_{edge}(O_k, M_l) = \frac{1}{|O_k|} \sum_{r \in O_k} |DT_{O_k}(r) - DT_{M_l}(r)|, \quad (9)$$

where $|O_k|$ represents the image size of O_k. When calculating Eq.(9), O_k and M_l must be normalized to a unit size and their centers must be set to the origins of O_k and M_l, respectively. Then, the best template can be selected from the model space Z according to this form:

$$M_{O_k} = \arg\min_{M_l \in Z} d_{edge}(O_k, M_l). \quad (10)$$

However, the distance d_{edge} is not accurate when an occluded object is compared. In Eq.(9), all the pixels in O_k are equally set for calculating d_{edge}. When an occluded object is handled, like Fig. 8(b) or (c), the pixels close to the non-occluded side should play more important roles than the ones close to the occluded side in template matching. When considering this effect, we weight a pixel r with the following equation:

$$w(r) = \exp(\kappa |x_r|), \quad (11)$$

where $\kappa = 0.1$ and x_r is the difference between r and the occluded side in the x coordinate (see Fig. 8(b) and (c)). When the weighting function is considered, we modify Eq.(9) as follows:

$$d_{edge}(O_k, M_l) = \frac{1}{C} \sum_{r \in O_k} w(r) |DT_{O_k}(r) - DT_{M_l}(r)|. \quad (12)$$

where $C = \sum_{r \in O_k} w(r)$.

D. Color model

Once each model template is built, the triangulation technique is used for over-segmenting each input model to different regions. Let O_n^l denote the nth template of the lth object. Then, O_n^l will be divided to different body parts $R_{O_n^l}^i$ using the triangulation technique. Assume that $R_{O_n^l}^{i,j}$ is one of triangulation meshes found from $R_{O_n^l}^i$. To tackle the occlusion problem, each mesh $R_{O_n^l}^{i,j}$ is further modeled using a Gaussian function whose parameters are the mean and variance of the major color in $R_{O_n^l}^{i,j}$ as follows:

$$R_{O_n^l}^{i,j} = Gaussian(r_\mu, r_\sigma, g_\mu, g_\sigma, b_\mu, b_\sigma).$$

V. Occluded Object Segmentation

To separate two objects from an occluded region R, the previous section adopts a distance transform for selecting a possible template for guiding the segmentation problem. This section will we use the triangulation technique to over-segment R to different triangulation meshes.

A. Over-segmentation Using Triangulation

To separate two occluded objects to different parts, we first over-segment the occluded region R to different triangulation meshes using its edge features. For this over-segmentation task, the contour feature is not a good choice for capturing the inner structure of R. Like Fig. 9(b), the contour feature does not provide lots of triangular meshes for separating these two occluded objects to individual ones. However, if the edge feature is used for the triangulation task, more useful meshes can be extracted for analyzing the inner structures of R. Like Fig. 9(c), if the edges in R are adopted, more detailed triangulation meshes can be extracted for segmenting R to different parts.

(a) (b) (c)

Fig. 9: (a) An occluded region R. (b) Triangulation result of R using its contour feature. (c) Triangulation result of R

using its edge feature.

Assume that t_k is one triangulated mesh in R and there are K objects in R. Then, a label field $L = \{l \mid l \in [0,...,K\text{-}1]\}$ on R can be created. Thus, the segmentation problem can be converted to a labeling algorithm on G. Given t_k, its optimal label l_{t_k} can be estimated by maximizing a posteriori probability $P(l \mid t_k)$ as follows:

$$l_{t_k} = \arg\max_{l \in L} P(l \mid t_k).$$

Using the Bayesian rule, the posteriori probability can be further rewritten by

$$P(l \mid t_k) \propto p(t_k \mid l) P(l), \qquad (13)$$

where $P(l)$ is the priori probability of the lth object and $p(t_k \mid l)$ is the likelihood probability of t_k belonging to the lth object. According to the template and color models of t_k, $p(t_k \mid l)$ can be further decomposed to two components, i.e.,

$$p(t_k \mid l) = p_{template}(t_k \mid l)\, p_{color}(t_k \mid l),$$

where $p_{template}(t_k \mid l) = \exp(-d_{template}(t_k, l)/\sigma^2_{template})$ and $p_{color}(t_k \mid l) = \exp(-d_{color}(t_k, l)/\sigma^2_{color})$. $d_{template}(t_k, l)$ and $d_{color}(t_k, l)$ denote the template and color distances between t_k and the lth object, respectively. $\sigma^2_{template}$ and σ^2_{color} represent the variances of $d_{template}(t_k, l)$ and $d_{color}(t_k, l)$, respectively. To calculate $d_{template}(t_k, l)$, a re-projection technique will be first described for projecting t_k on the lth object and then estimating their dissimilarity. After that, the color distance $d_{color}(t_k, l)$ will be discussed in Section 6.3.

B. Template Re-projection

(a) (b) (c)

Fig.10: Template re-projection. (a) A mesh shown by a red dot. (b) Projection of a mesh t_k on the selected template model M_{O_l}. (c) Re-projection.

In Section 5, the technique of particle filter is used for tracking each object O_l in an occluded region R (see Fig. 6). Since there are lots of occluded pixels found in R, given a mesh t_k, which object it belongs to is till difficultly determined. In what follows, a novel template re-projection technique will be described for calculating the dissimilarity between t_k and each template model.

Let $c_{M_{O_l}}$ and c_{t_k} dente the central positions of M_{O_l} and t_k in R, respectively. With $c_{M_{O_l}}$ and c_{t_k}, t_k in M_{O_l} can be determined as follows:

$$p_{M_{O_l}}(t_k) = c_{t_k} - c_{M_{O_l}},$$

where the center of $c_{M_{O_l}}$ is the origin. Fig.10 shows an example to illustrate our idea. In (a), a mesh t_k (denoted by a red dot) is given. (b) shows one of template models selected by the distance transform. Let g_i denote one of body parts in M_{O_l} whose center is denoted by a yellow dot like Fig.10(b). Then, with $p_{M_{O_l}}(t_k)$, the relative position between t_k and g_i can be accordingly obtained. Thus, two projection tasks for building the spatial relations between R and M_{O_l} can be performed, i.e., the one from R to M_{O_l} and the other one from M_{O_l} to R. For the first one, we project t_k on M_{O_l} so its corresponding mesh $\bar{t}_k$ in M_{O_l} can be found (see the red dot in Fig.10(b)). For the second one, all the body parts g_i in M_{O_l} are projected on R and then their corresponding meshes t_{g_i} from R can be found using the nearest neighbor criterion. Let n_g denote the number of body parts in M_{O_l}. Then, the distance between t_k and M_{O_l} can be calculated as follows:

$$d_{template}(t_k, M_{O_l}) \cong \frac{1}{n_g+1}[d_\Delta(t_k, \bar{t}_k) + \sum_{i=1}^{n_g} d_\Delta(t_k, t_{g_i})],$$

where $d_\Delta(t_i, t_j)$ is the distance between two triangular meshes t_i and t_j. Let h_t denote the color histogram of a triangular mesh t. Then, the KL distance is used to measure $d_\Delta(t_i, t_j)$:

$$d_\Delta(t_i, t_j) = \sum_{k=0}^{n_{bin}-1} h_{t_i}[k] \left(\log \frac{h_{t_j}[k]}{h_{t_i}[k]} \right). \quad (14)$$

C. Color Similarity

In Section 4.2, a clustering technique to cluster the colors of template models to different clusters. Let C^l denote the set of color models in the *l*th object and C_n^l the *n*th color model in C^l. Then, the color distance $d_{color}(t_k, l)$ between t_k and the *l*th model is defined by:

$$d_{color}(t_k, l) = \min_{C_n^l \in C^l} \frac{(r_{t_k} - r_{C_n^l})^2}{\sigma_{t_k,r}^2} + \frac{(g_{t_k} - g_{C_n^l})^2}{\sigma_{t_k,g}^2} + \frac{(b_{t_k} - b_{C_n^l})^2}{\sigma_{t_k,b}^2}. \quad (15)$$

After normalization, $d_{color}(t_k, l)$ is redefined as follows:

$$d_{color}(t_k, l) = \frac{d_c(t_k, l)}{\sum_l d_c(t_k, l)}. \quad (16)$$

D. Object Recovering

(a) (b) (c) (d) (e) (f)

Fig.11: Repairing results of occluded objects.

Once an occluded region is separated to different objects, the next task is to recognize their related posture and behavior types. Like Fig.11, two rectangle boxes were used to track the observed objects. In (b), we use the blue area to denote the overlapping component between them. Let O_a and O_b denote the extracted objects from the occluded object *R*. In addition, A_R denote this overlapping component of *R*. Then, with A_R, we can an 'union' operation to repair O_a and O_b, respectively, as follow:

$$\bar{O}_a = O_a \cup A_R \text{ and } \bar{O}_b = O_b \cup A_R. \quad (17)$$

Like Fig.11(c) and (d), two objects O_a and O_b are extracted from the occluded object *R* in (a) using the technique proposed in Section 5. Then, we repair O_a and O_b as $\bar{O}_a$ and $\bar{O}_b$, respectively, using Eq.(17). Actually, the repairing task is only performed only when a serious fragmentation happens. To detect this case, a fragmentation ratio is defined as

$$f_{ratio} = \frac{\min(|O_a|, |O_b|)}{\max(|O_a|, |O_b|)},$$

where $|O|$ denotes the number of pixels in O. If f_{ratio} < 30%, a serious fragmentation happens. Under this case, only the smaller object (like O_a) will be repaired and the larger one (like O_b) is kept. In real cases, some holes will exist in each object. Thus, a morphological operation 'closing' is further used to fill the above holes. Fig.11(e) and (f) show the repairing results of O_a and O_b, respectively. After repairing, the posture types of O_a and O_b can be then recognized using Eq.(4), respectively.

VI. Behavior Recognition

In this paper, each movement sequence of a human subject is represented by a continuous sequence of postures which changes over time. This paper tries to present a novel approach to recognize human behaviors directly from videos even though occlusions happen.

Let $A = \{Q_0^A, Q_1^A, ..., Q_t^A, ...\}$ denote an action sequence. We can convert it to a string using Eq.(4) and then analyze throng a string matching scheme. Assume $\bar{K}_t^A$ is the recognized type of a query posture Q_t^A in *A*. In real cases, there should be some key postures in $\Omega_{K_{Q_t}}$ which are very similar to $\bar{K}_t^A$. Thus, the correct type of each posture Q_t^A in *A* will not be always found. To enlarge the difference between $\bar{K}_t^A$ and other

similar postures in $\Omega_{K_{Q_t}}$, we represent Q_t^A using not only $\bar{K}_t^A$ but also the similarities between $\bar{K}_t^A$ and other key postures in $S_{KP}^{90^\circ}$. Thus, a feature vector h_t^A for representing the *t*th posture Q_t^A is constructed using the form:

$$h_t^A(i) = S_{cc}(\bar{K}_t^A, K_i), \tag{18}$$

where $K_i \in S_{KP}^{90^\circ}$. h_t^A keeps the relations between Q_t^A and all the key postures in $S_{KP}^{90^\circ}$. With Eq.(18), we can convert *A* to a matrix representation as:

$$H_A = \{h_t^A\}_{t=0,\ldots,|A|-1}, \tag{19}$$

where |*A*| means the number of postures in *A*. Like Fig. 12, an action matrix H_{A_s} is used to represent a "squatting" action A_s. Each entry $H_{A_s}[t, K_i]$ records the similarity between the *i*th key posture K_i in $S_{KP}^{90^\circ}$ and Q_t^S in A_s.

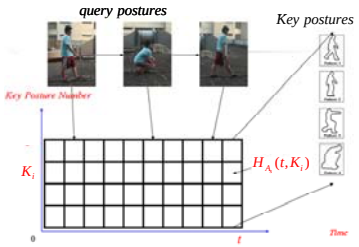


Fig. 12: An action matrix H_{A_s} for representing a "squatting" action.

Given a query action *A* and an action type *D* in the database, they can be converted to two action matrices H_A and H_D, respectively. For the *i*th element h_i^A in H_A and the *j*th element h_j^D in H_D, the KL distance is used to measure their dissimilarity (or cost) as follows:

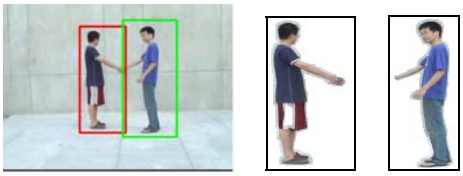
$$cost(i, j) = \sum_{k=0}^{|S_{KP}^{90^\circ}|-1} h_i^A[k] \left(\log \frac{h_i^A[k]}{h_i^D[k]} \right). \tag{20}$$

Usually, an action sequence will different temporal scaling changes, different initial states, and symbol converting errors. To tackle the above problems, a dynamic time warping (DTW) techqniue is used to calcuatle the distance between H_A and H_D. Let $DTW_{H_A,H_D}(i, j)$ denote the warping cost between the two subsequences $\{h_t^A\}_{t=0,\ldots,i}$ and $\{h_t^D\}_{t=0,\ldots,j}$. That is, $DTW_{H_Q,H_D}(i, j)$ is the minimum number of dynamic time warping cost needed to transform the first $(i+1)$ vectors of H_A into the first $(j+1)$ vectors of H_D. The value of $DTW_{A,D}(i, j)$ can be recursively calculated using the following form:

$$\begin{aligned} DTW_{A,D}(i, j) = \min(&DTW_{A,D}(i-1, j) + cost(i, j), \\ &DTW_{A,D}(i, j-1) + cost(i, j), \\ &DTW_{A,D}(i-1, j-1) + cost(i, j)). \end{aligned} \tag{21}$$

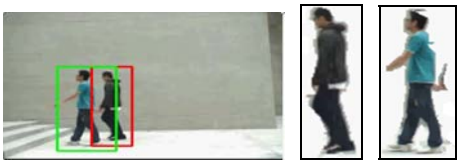
Based on Eq.(21), the distance between *A* and *D* can be measured.

VIII. Experimental Results



(a) (b) (c)

Fig. 13: Occlusion due to the handshaking action. (a) Input frame. (b) and (c): Objects extracted from (a).



(a) (b) (c)

Fig. 14: Object separation when a walking sequence was handled. (a) Input frame. (b) and (c): Objects extracted from (a).

To test the efficiency and effectiveness of the proposed approach, we constructed a large database of more than thirty thousand postures, from which we obtained over three hundred human movement sequences. The first set of experiments was used to examine the ability of our scheme to extract objects from a handshaking sequence. Fig. 13 shows the result of object extraction from an occluded foreground region. Fig. 14 shows the result of object separation when a walking sequence was handled. It is noticed that the trouser colors between them are similar. However, our method still works well to separate them to different parts.

(a) (b) (c)

Fig.15: Object separation when a walking sequence was handled. (a) Input frame. (b) and (c): Objects extracted from (a).

The sequence "Walk-parallel" is two pedestrians walking to the same direction. A key point worthy to be noticed is that they must enter the scene in different time, so we can establish their posture template. Even though they wear blue jeans both in Fig.15(c) and Fig.15(f), we get the good performance.

(a) (b) (c) (d) (e) (f)

Fig.16: Segmentation results of the "turn around" sequence. (a) and (b): Input frames. (c) and (d): Results of (a). (e) and (f): Results of (b).

(a) (b) (c) (d)

Fig.17: Segmentation result for the "outdoor" sequence. (a) and (b): Input frames. (c) and (d): Results of (c) and (d).

Fig.16 shows the results of the sequence "turn-around". The "turn-around" action will often lead to a wrong label for identifying objects. However, our method still works well to separate them to different objects. Fig.17 shows the segmentation results when an outdoor scene was handled. In an outdoor environment, the lighting will change significantly. However, this case is still well tackled using our scheme.

The most difficult case is the people wearing the clothes with similar color. Fig.18 shows the segmentation results when the clothes colors of peoples are similar. However, our method performs well to deal with this case. Table. 1 lists the precision analysis of our method under different test sequences. The superiority of our proposed segmentation method can be proved from this table.

(a) (b) (c) (d) (e) (f)

Fig.18: Segmentation results when the clothes colors are similar. (a) and (b): Input frames. (c) and (d): Results of (a). (e) and (f): Results of (b).

Table. 1: Accuracy analysis of our method under different video types.

Methods / Action types	Proposed
Cross	0.894
walk parallel	0.873
Handshake	0.9599
Turn round	0.892
Complicated	0.714

Table. 2: The true positive of recognition human behavior.

Types	Walk-left	Walk-right	Stretch-left	Stretch-right
Accuracy	0.43	0.52	0.68	0.72

Table. 3: The true positive of recognition human behavior.

Types	Walk-left	Walk-right	Stretch-left	Stretch-right
Accuracy	0.59	0.75	0.9	0.93

Table. 4: Confusion matrix of behavior recognition using our method.

	W-Left	W-Right	Stretch Left	Stretch Right
Walk Left	88	71	27	16
Walk Right	39	72	6	2
Stretch Left	13	8	51	3
Stretch Right	3	10	9	56

Once an occluded region is separated to different objects, we can then recognize their corresponding behavior types. In this paper, four behavior types were recognized, i.e., walk-left, walk-right, stretch-left, and stretch-right. Table 2 shows the accuracy of these four behavior types if two occluded objects are matched using Eq.(4). Due to occlusion, the accuracy is not so high. However, if the matrix representation was used (see Eq.(20)) for matching actions using the K-L distance, the recognition accuracy can be improved significantly. Table 3 lists the accuracy analysis when the matrix representation was used. Table 5 shows the confusion matrix among four behavior

types for illustrating the accuracy of behavior recognition. Clearly, the number of "true-positive" recognition results was significantly improved.

VIIII. CONCLUSION

This paper has proposed a novel segmentation method for segmenting an occluded region into different objects. The contributions of this paper can be summarized as follows:

(a) A triangulation-based method was proposed for extracting important skeletal features and centroid contexts for posture classification and model representation.

(b) A clustering scheme was proposed for key model selection. Then, a model-driven approach was proposed for segmenting the occluded regions into different objects.

(c) A hierarchical model selection method was proposed for tackling the ambiguity problem in model selection when postures' contours are similar.

(d) A matrix representation was proposed for recognizing human behaviors more accurately even though occlusions happen.

REFERENCE

[1] L. P. Chew, "Constrained Delaunay triangulations," Algorithmica, vol. 4, no.1, pp.97-108, 1989.

[2] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," IEEE Transactions on Pattern Recognition and Machine Intelligence, vol. 24, no. 4, pp. 509–522, April 2002.

[3] N. Gordon and D. Salmond, "Bayesian state estimation for tracking and guidance using the bootstrap Filter," Journal of Guidance, Control and Dynamics, vol. 18, no. 6, pp. 1434-1443, 1995.

[4] M. Isard and A. Blake, "CONDENSATION: conditional density propagation for visual tracking," International Journal on Computer Vision, vol. 29, no. 1, pp. 5-28, 1998.

[5] T. B. Moeslund and E. Granum, "A survey of computer vision-based human motion capture," Computer Vision and Image Understanding, vol. 81, no. 3, pp. 231-268, Mar. 2001.

[6] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman, "Human body model acquisition and tracking using voxel data," International Journal of Computer Vision, vol. 53, no. 3, pp. 199-223, 2003.

[7] N. M. Oliver, B. Rosario, A. P. Pentland, "A Bayesian Computer Vision System for Modeling Human Interactions," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 831-843, Aug. 2000.

[8] R. Rosales and S. Sclaroff, "3D Trajectory Recovery for Tracking Multiple Objects and Trajectory Guided Recognition of Actions," Proc. of IEEE Conf. On Computer Vision and Pattern Recognition, vol. 2, pp. 117-123, June 1999.

[9] D. V. Pynadath and M. P. Wellman, "Probabilistic State Dependent Grammars for Plan Recognition," Sixteenth Conference on Uncertainty in Artificial Intelligence, pp. 507-514, 2000.

[10] C. R. Wren, A. Azarbayejani, T. Darrell, and A.P. Pentland, "Pfinder: real-time tracking of the human body," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 780-785, July 1997.

[11] S. X. Ju, M. J. Black, and Y. Yaccob, "Cardboard people: a parameterized model of articulated image motion," International Conference on Automatic Face and Gesture Recognition, pp. 38-44, Killington, Vermont, 1996.

[12] A. Yilmaz, X. Li, and M. Shah, "Contour-Based Object Tracking with Occlusion Handling in video Acquired Using Mobile Cameras," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.26, no. 11, p.1531-1536, Nov. 2004.

[13] P. Peursum, S. Venkatesh, G. A.W. West, and H. Bui, "Object Labeling from Human Action Recognition," Proc. of IEEE Conf. on Pervasive Computing and Communications, pp.399-406, March, 2003.

[14] A. M. Elgammal and L. S. Davis, "Probabilistic framework for segmenting people under occlusion," Proceedings of Eighth IEEE International Conference on Computer Vision, vol. 2, pp. 145-152, 2001.

[15] D. Stauffer and W. Grimson, "Kernel-Based Object Traking," IEEE Trans. on Pattern Analysis and Machine Intellignce, vo25, no.5, pp. 564-575, May 2003.