Parsing the Hand in Depth Images

Hui Liang, Junsong Yuan, *Member, IEEE*, and Daniel Thalmann

***Abstract*—Hand pose tracking and gesture recognition are useful for human-computer interaction, while a major problem is the lack of discriminative features for compact hand representation. We present a robust hand parsing scheme to extract a high-level description of the hand from the depth image. A novel distance-adaptive selection method is proposed to get more discriminative depth-context features. Besides, we propose a Superpixel-Markov Random Field (SMRF) parsing scheme to enforce the spatial smoothness and the label co-occurrence prior to remove the misclassified regions. Compared to pixel-level filtering, the SMRF scheme is more suitable to model the misclassified regions. By fusing the temporal constraints, its performance can be further improved. Overall, the proposed hand parsing scheme is accurate and efficient. The tests on synthesized dataset show it gives much higher accuracy for single-frame parsing and enhanced robustness for continuous sequence parsing compared to benchmarks. The tests on real-world depth images of the hand and human body show the robustness to complex hand configurations of our method and its generalization power to different kinds of articulated objects.**

***Index Terms*—Depth-context feature, hand parsing, Markov random field.**

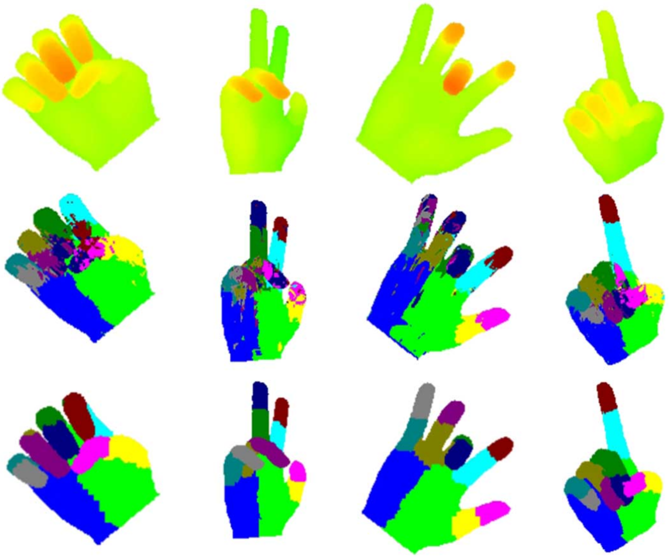


Fig. 1. Comparison of the hand parsing results using per-pixel classification (middle row) and the proposed SMRF based hand parsing scheme combined with temporal reference (lower row). The upper row shows the input depth images.

I. Introduction

VISION-based hand pose tracking and gesture recognition have been research focus in recent years due to their importance in human computer interaction scenarios such as virtual reality and sign language recognition [1]. Such systems greatly improve the users' interaction experience by providing a natural and convenient way for interaction between human beings and computers, compared to the intrusive counterparts, *e.g.* the data-glove [7] and the optical-marker based methods [32]. While a lot of work has been done [2]–[5], [24]–[26], robust hand pose tracking and gesture recognition with visual inputs remains a challenging problem and the performance achieved cannot compare to the alternative methods, such as the data-glove based approaches [22].

One important reason for the difficulty in vision-based hand pose tracking and gesture recognition is the lack of discriminative features. In many previous studies the optical camera is used as the input [4], [5]. As the hand is quite homogeneous in color, the commonly used features, *e.g.* edge and silhouette, are sensitive to lighting condition variations and cluttered background, and are inherently ambiguous for matching [24]–[26]. The depth sensors can provide more discriminative visual features and various methods have been proposed, *e.g.*, the fast point feature histograms [27], the geodesic distance map [33], and the depth kernel descriptors [34]. Although these low-level features prove effective in rigid object recognition, they still lack the discriminative power to provide a compact representation for the articulated objects such as the hand.

The parsed hand parts are very useful high-level features for both hand pose tracking and gesture recognition tasks, *i.e.* segmentation of the hand region into different parts. Similar work has been done in both the fields of full-body tracking [8], [29], [40] and hand parsing [28], [30]. For the depth image input, the existing hand parsing schemes are mostly based on per-pixel classification, and the labeled results are quite noisy, especially when the input hand pose is not covered by the training dataset.

In this paper we propose a unified framework to utilize both temporal and spatial constraints for hand parsing. Our contribution mainly lies in two aspects. First, we develop a depth-context feature and further improve its performance by a distance-adaptive sampling scheme, which proves more effective for hand parsing compared to the binary depth comparison feature in [8]. Second, we propose a novel Superpixel Markov Random Field

Manuscript received August 21, 2013; revised November 28, 2013; accepted January 21, 2014. Date of publication February 13, 2014; date of current version July 15, 2014. This work, which was carried out at BeingThere Centre, was supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Chong-Wah Ngo.

H. Liang and D. Thalmann are with Being There Centre, Institute of Media Innovation, Nanyang Technological University, Singapore 637553 (e-mail: hliang1@e.ntu.edu.sg; danielthalmann@ntu.edu.sg).

J. Yuan is with the School of Electrical & Electronics Engineering, Nanyang Technological University, Singapore 639798 (e-mail: jsyuan@ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2014.2306177

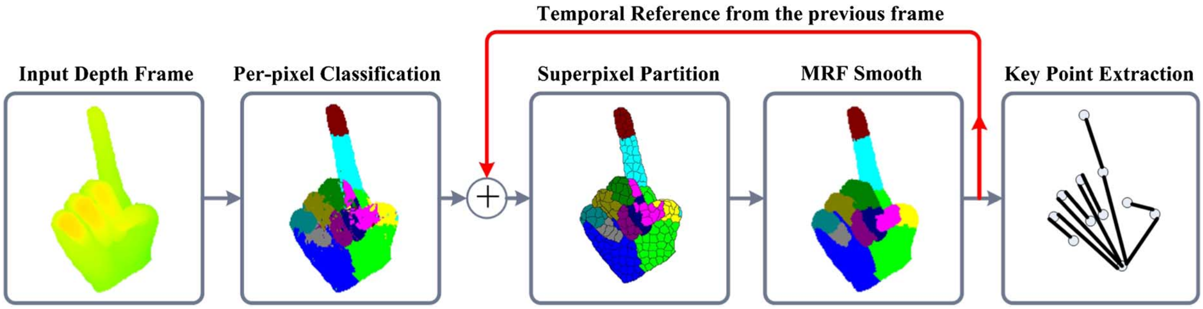


Fig. 2. The pipeline of the proposed hand parsing scheme.

(SMRF) framework that can model the spatial constraints to improve the hand parsing performance considerably based on the per-pixel classification results. Compared to the benchmark [8] and the state-of-the-art methods [28], [40], our method achieves much higher accuracy while maintaining linear computational complexity. Several examples of the experimental results are shown in Fig. 1 to illustrate the advantages of SMRF.

We present the pipeline of the proposed hand parsing algorithm in Fig. 2. The main idea of the depth-context feature is to describe each pixel of the depth image using a sampling grid centered at the current pixel. In our distance-adaptive scheme, the grid is sampled with non-fixed steps, *e.g.* more densely sampled near to the center and vice versa. The random decision forest classifier is used for per-pixel classification based on the depth-context feature. The temporal constraints are enforced by learning the 3D position distribution of the hand parts from the previous frame, and combining it with the RDF classifier to form an ensemble of classifiers. As to the spatial constraints, the neighborhood smoothness of the pixel labels is enforced in the SMRF framework to suppress the per-pixel classification error. By using the superpixel, the misclassified isolated regions can be represented as an atomic element and the computational complexity of Markov Random Field (MRF) based smoothing can be largely reduced. Finally, the parsed hand parts are further processed to get a set of key points as a high-level representation of the hand.

The remainder of this paper is organized as follows: Section II provides a literature review of related hand parsing techniques, and the main modules of our proposed scheme are briefly introduced in Section III. From Section IV to Section VI we present the detailed techniques of each module. Section VII presents the experimental results. At last, Section VIII concludes the paper with some discussions.

II. Related Work

Hand parsing refers to the classification of the hand region into a pre-defined set of different hand parts, *e.g.* different fingers and the palm. Since the hand is homogeneous in color and lacks stable features against hand articulation in the depth image, robust hand parsing remains a challenging task. The parsed hand parts are important features for hand pose tracking and gesture recognition, which are more discriminative than the raw color or depth images of the hand.

To handle the color homogeneity of the hand, the color markers or gloves are often the choices for this task with color camera inputs. In [31] a glove with color-coded rings is used for sign language recognition, and the rings correspond to the joints of each finger. In [14] a color glove with specially designed patterns is used to segment each hand part. A Hausdorff-like distance metric is used for template matching and the database is indexed by similarity sensitive coding to accelerate nearest neighbor search.

A lot of related work for unmarked inputs has been done in both full-body and human hand parsing. In [35] an iterative parsing scheme is proposed to parse the body parts in unconstrained color images by finding the optimal body pose. The initial pose is first determined by using only the edge feature, and it is then used to compute the new appearance model to determine the new parse. This procedure is repeated until convergence. In [36] the AND/OR graph is utilized to represent the body part configuration, the parameters of which are learned by the max-margin algorithm for structure learning. Some shape analysis techniques, *e.g.* convex shape decomposition [37], have been applied to hand parsing. In [37] the individual fingers and palm are segmented by decomposing the hand contour into separated near-convex polygons. As these methods mostly rely on the low-level color features, they have difficulty in parsing the overlapping hand parts due to their similarity in color.

The depth cameras are more powerful tools for body and hand parsing. The occluding parts with similar colors can be separated with their depths, and the pixels can be better described based on the depth value contrast. In [6] a labeled contour model is proposed for gesture recognition. The depth image is first classified into hand parts with the position feature, and the results are then used to label the contour points to provide extra clue for matching. In [8] the binary depth comparison feature is proposed to describe the pixels in the depth image. This feature is represented by a set of pairs of relative offsets from the pixel to be classified. For each dimension of the feature, the value is set to be the depth difference between the two neighboring points defined by the pair of offsets. The RDF classifier is used for classification based on the feature. This framework is also utilized for hand parsing in depth images in [30]. In [39] the authors propose to use ICP to build the temporal correspondence between the previous frame and the current frame for hand parsing, and extract the hand part edges as additional constraints to refine the parsing results. While this method largely relies on the temporal reference for parsing, it is inherently sensitive to tracking failure due to its assumption of small hand shape deformation between successive frames.

In [28] the authors propose the Multi-layered Random Decision Forests to parse the hand. The hand configuration parameters of the templates in the training dataset are first clustered by spectral clustering and divided into K different classes. During training, a RDF classifier is learned for hand shape classification (SCF), which classifies the whole depth image of the hand into the K classes. An individual RDF classifier (GEN) is trained on each of the K subsets of the original dataset to parse the hand into different finger and palm parts with the depth feature in [8]. During testing, the SCF classifier is first applied to the input hand depth image, and its result is used to pick up the GEN classifiers for hand parsing. While considerable improvements are reported over the method in [8], the multi-layered RDFs framework implicitly assumes that the dataset consists of many templates with similar hand configuration parameters, which does not hold true for hand parsing in full degree-of-freedom hand motion scenarios.

In [40], the authors propose to refine the per-pixel classification results obtained with the method in [8] by applying graph cut optimization to a pixel-level graph built on the current and previous frames, and produces 5.96% increase of accuracy compared to [8]. Though not reported, the time performance of such pixel-level graph cut optimization based method is still open to doubt for efficient parsing.

III. System Overview

We present a Superpixel Markov Random Field (SMRF) framework to parse the depth image of the bare hand into individual parts, in which both the temporal and spatial dependencies of the labeled parts are exploited. The input of the framework is a sequence of depth images, where we assume that only one hand is visible and the hand is the nearest object to the camera. The output of the hand parsing scheme is the part label images and key point sets corresponding to the hand parts.

Given the inputs, the processing pipeline of our hand parsing scheme is shown in Fig. 2. The functions of the modules are listed as follows:

Per-pixel classification: to assign an initial part label to each pixel by classification with the learned RDF classifier and the temporal position classifier. A distance adaptive feature selection scheme is combined with a depth-context feature to describe each pixel for classification.

Superpixelpartition: to partition the initial hand part segmentation into a set of superpixels to reduce the computational complexity involved in MRF smoothing.

MRF inference: to reassign the hand part labels via superpixel-level inference. The depth discontinuity, superpixel boundary shapes and hand label co-occurrence are all handled to construct the MRF graph for efficient inference.

Key point extraction: to infer the center of each labeled hand part based on their distributions in 3D space.

IV. Hand Modeling

A 3D hand model is needed to generate the training samples for hand part classification as well as to measure the compatibility between the image features and a hypothesized hand pose ϕ. We build a fully deformable model with 3D closed mesh to simulate the real hand. The model consists of a skeleton system and a skin surface mesh. The skeleton has 27 degrees of freedom (DOF), including 6 DOFs of global motion and 21 DOFs of local motion [11]. It is modeled as a kinematic chain of 20 joints connected by bones in a tree structure with the root at the wrist, as shown in Fig. 3(a). A set of static and dynamic constraints [11], [12] are adopted to limit the parameter space of the hand motion. The skeleton thus has an equivalent of 18 degrees of freedom. The skin mesh consists of about 5000 vertices, which form approximately 7000 triangles. Each vertex and triangle is assigned a label $l \in L$ to indicate which hand part they belong to. Give a hypothesized hand pose ϕ, the joint positions of the skeleton are first computed using forward kinematics, and the positions of the vertices on the skin mesh are then updated using the skeleton subspace deformation method [13].

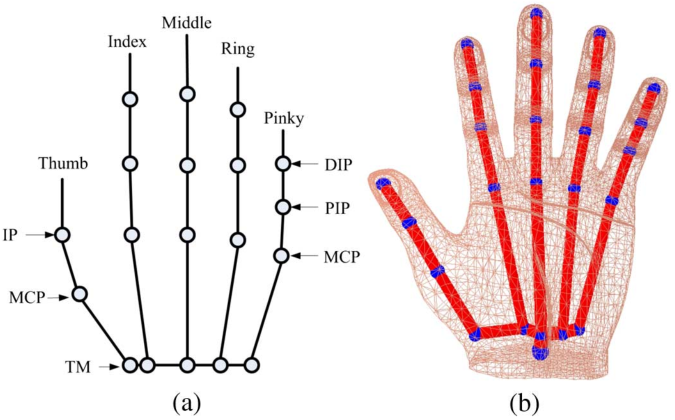


Fig. 3. The kinematic chain (Left) and the 3D hand model (Right).

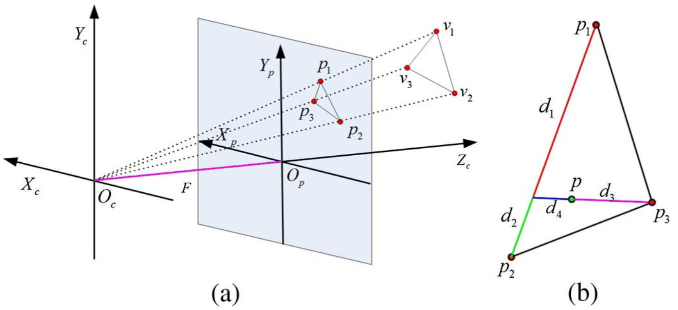


Fig. 4. The pin-hole camera model (Left) and the depth image interpolation scheme (Right).

In order to generate the depth image and label image to train the hand part classifier, we adopt the pin-hole camera model [9] to project the hand model onto the image plane. This is done by calculating the projection of all the triangles of the skin mesh, as shown in Fig. 4(a). The pixel coordinate $\mathbf{p}_i$ of a 3D point $\mathbf{v}_i = [a_i, b_i, c_i]^T$ is projected by:

$$\mathbf{p}_i = \Psi_P(\mathbf{v}_i) = \frac{F}{c_i} \times \begin{bmatrix} D_x a_i \\ D_y b_i \end{bmatrix} + \begin{bmatrix} a_o \\ b_o \end{bmatrix}, \qquad (1)$$

where F is the focal length; D_x and D_y are the coefficients to define the metric units to pixels; a_o and b_o are the principal point of the image plane. These parameters are intrinsic parameters of the camera and can be obtained by camera calibration techniques.

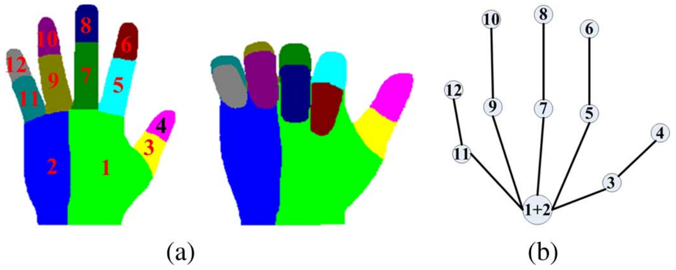


Fig. 5. The hand partition scheme. (a) The label distributions for different hand parts. (b) The tree-structured hierarchy of the positions of the labeled hand parts.

Following this projection model, the three vertices of each triangle are projected onto the image plane to get $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$. The pixels with the projected triangle in the label image are assigned the corresponding label of the triangle $(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$. Besides, we use a linear interpolation method to fill the pixels within $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ in the depth image, as shown in Fig. 4(b). The 3D point $\mathbf{v}$ corresponding to $\mathbf{p}$ is a linear combination of the three vertices:

$$\mathbf{v} = \frac{(d_1 d_3 \mathbf{v}_2 + d_2 d_3 \mathbf{v}_1)}{(d_1 + d_2)(d_3 + d_4)} + \frac{d_4 \mathbf{v}_3}{d_3 + d_4}, \tag{2}$$

where (d_1, d_2, d_3) are the distances from $\mathbf{p}$ to the vertices, and d_4 is the length from $\mathbf{p}$ to $(\mathbf{p}_1, \mathbf{p}_2)$ along the extension of d_3. Besides, since the skin mesh is closed, multiple triangles on the mesh can be projected to the same pixel on the depth and label images. In this case, the pixel will take values from the triangle that gives the minimum depth value.

V. Per-Pixel Classification

The task of hand part classification is to assign a label $l \in L$ to each pixel in the depth image of the hand region. Fig. 5(a) shows our hand label partition scheme, where the whole hand is classified into twelve non-overlapping parts. These hand parts are not independent from each other, and their positions follow a tree-structured hierarchy, as shown in Fig. 5(b). To perform per-pixel classification, we propose a depth-context feature with a novel distance-adaptive sampling scheme to describe the 3D context for each pixel, based on which a trained RDF classifier is used to assign the part labels. In case that the temporal reference is available, the position distribution of each labeled part will also be learned and used in combination with the RDF classifier to form an ensemble of classifiers for per-pixel classification. Details of the temporal constraints will be discussed in Section V-D.

A. Depth-Context Feature

We propose a depth-context feature for classification of each pixel in the depth image and improve its performance with a distance-adaptive sampling scheme. The feature describes the 3D context of each pixel using the relative depths between the pixel and its neighboring pixels as shown in Fig. 6(a). The red circle denotes the current point for classification, and the green circles denote the context points to calculate the feature values. The 3D relative position between each context point and the current

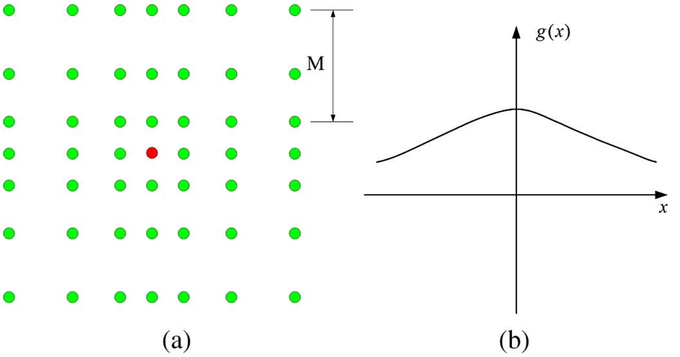


Fig. 6. The distance-adaptive scheme for selecting depth image feature candidates, in which the red circle indicates the current pixel, and the green circles indicate the candidate offset features (Left). The sampling density function along each axis (Right).

pixel is defined as a 3D offset $\mathbf{v}_d = [a_d, b_d, 0]^T$. Given the pixel $\mathbf{p}$ and its corresponding 3D point $\mathbf{v}$, the pixel coordinate $\mathbf{p}_c$ of the context point can thus be obtained by projecting the 3D position of the context point to the image plane $\mathbf{p}_c = \Psi_P(\mathbf{v} + \mathbf{v}_d)$. This projection procedure ensures the relative pixel coordinate of the context point is adaptively adjusted by the depth value of the current pixel and their 3D relative positions are thus conserved. This is similar to the normalization of the feature offsets in [8] and the depth context feature is thus depth-invariant. The feature value is obtained by the depth difference between the current pixel and the projected pixel of the context point:

$$f_F(\mathbf{p}, \mathbf{v}_d) = d_z(\mathbf{p}) - d_z\left[\Psi_P(\mathbf{v} + \mathbf{v}_d)\right], \tag{3}$$

where d_z is the depth value at the given pixel in the depth image. In addition, the feature value f_F is restricted to the range $[-\varepsilon_d, \varepsilon_d]$, where ε_d is a constant value. This is because the background pixels in the synthesized training images are usually assigned a large constant depth value, which is generally different from the background depth values in the test images. Thus, given the same hand configuration, the feature values on the context points that lie on the background can be very different on the training and testing depth images. Such inconsistency can be eliminated by threshold with the constant ε_d. Meanwhile, for a point on the hand, ε_d should be big enough to capture the depth difference between the point and any other context point lying on the hand. In our implementation we set $\varepsilon_d = 0.3$ m to parse the hand.

Another issue is the selection of the context point positions. In previous work [8], [30], the context points are randomly generated within certain ranges. However, the nearer points should be more important to describe the context than the faraway points. Especially, most of the faraway context points of the finger parts lay either on the background or the palm part, both of which are quite homogeneous regions. To this end, we propose a distance-adaptive sampling scheme to generate the context points, as shown in Fig. 6(b). The points nearer to the current pixel are more densely sampled, and vice versa. For simplicity, we focus the discussion on one dimension as the sampling scheme is symmetric. The sampling density function $g(x)$ is adopted to determine the location of each context point. Let the feasible range of the context points be $[-h, h]$. $g(x)$ is a non-increasing function, and satisfies $\int_0^h g(x)dx = M$. M is a parameter to determine the grid size. In our implementation we choose a linear function for $g(x)$. The coordinate of the context point can be obtained by solving the following equation for x_d, $i = 1, \ldots, M$:

$$\int_0^{x_d} g(x)dx = i. \tag{4}$$

To validate the effectiveness of the proposed depth-context feature and the distance-adaptive sampling scheme, we compare its classification accuracy to that of the binary depth comparison feature in [8]. The results show it achieves 12.2% increase of accuracy, and the details of the experiments are given in Section VII-A.

B. RDF Classification

Based on the depth-context feature, we adopt the Random Decision Forest classifier [10] to label the hand parts by per-pixel classification. The training procedure of the random decision forest is essentially the same as that in [8]. To generate the training samples, we randomly select ten pixels for each hand part and the background from each training image, and each sample consists of the label of the pixel and the corresponding depth-context feature value.

During the test stage, an input pixel $(\mathbf{p}, \mathbf{v})$ is first processed by each tree in the random decision forest. For each tree, the posterior probability $P_i(l|\mathbf{v})$ is obtained by starting at the root and recursively assigned to the left or the right child based on the tree node test result until it finally reaches a leaf node. The final posterior probability $P(l|\mathbf{v})$ is obtained by fusing the results of all the trees in the random forest:

$$P(l|\mathbf{v}) = \frac{1}{N}\sum_{i}^{N} P_i(l|\mathbf{v}), \tag{5}$$

where N is the number of trees in the random decision forest. The label of the pixel can be directly determined by MAP estimation: $l^* = \arg\max_l P(l|\mathbf{v})$. However, the final label decision is not made here, and the posterior $P(l|\mathbf{v})$ will be further processed by the Superpixel-MRF framework to give the refined results.

C. Key Point Extraction

The key points are compact representations of the hand parsing results. Instead of calculating the 3D centroid of each labeled hand part, we calculate the expectation positions of each hand part based on the label distribution given by hand parsing. Let the 3D point set on the input hand be H. Let the label distribution given by the RDF classifier be $P(l|\mathbf{v})$, $\mathbf{v} \in H$. The 3D position distribution of a hand part l can be calculated by:

$$P(\mathbf{v}|l) = \frac{P(l|\mathbf{v})P(\mathbf{v})}{\sum_{\mathbf{v}_i \in H} P(l|\mathbf{v}_i)P(\mathbf{v}_i)} = \frac{P(l|\mathbf{v})}{\sum_{\mathbf{v}_i \in H} P(l|\mathbf{v}_i)}, \tag{6}$$

where we assume uniform prior of the 3D positions of the depth points. Thus, the expected position of the hand part l is given by:

$$\mathbf{v}_c^l = \sum\nolimits_{\mathbf{v}_i \in H} \mathbf{v}_i P(\mathbf{v}_i|l). \tag{7}$$

Note for the two hand parts 1 and 2 that belong to the palm in Fig. 5(a), the 3D points within them and the corresponding label distribution are merged to get the palm position. Let the resulting positions of each hand part be $U_I = \{v_c^l | l = 1, \ldots, L\}$. These points are further arranged into a tree hierarchy based on their relationship in the hand skeleton, as illustrated in Fig. 5(b), where the root corresponds to the palm position.

D. Temporal Constraints

The temporal references are useful when parsing the hands in successive images. Given that the hand parsing result in the previous frame is available, we propose to use it as an auxiliary classifier for the RDF classifier for per-pixel classification in the current frame. This classifier forms an ensemble of classifiers with the RDF classifier. To this end, we first use the 3D Gaussian distribution to approximate the point distribution within each hand part in the previous frame, *i.e.* $P_T(\mathbf{v}|l) \sim \mathcal{N}(\mu_l, C_l)$, and the parameters μ_l and C_l are learned from the points that belong to each hand part in the previous frame. The temporal classifier is thus given by:

$$P_T(l|\mathbf{v}) \propto P_T(\mathbf{v}|l)P_T(l), \tag{8}$$

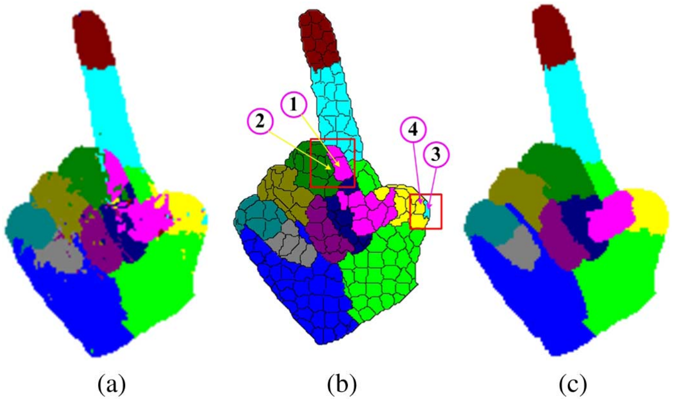
where $P_T(l)$ is proportional to the number of points within each part. For clarity we denote the RDF classifier as $P_R(l|\mathbf{v})$. The previous per-pixel classification scheme can thus be accomplished by the ensemble of P_T and P_R to incorporate the temporal reference, that is:

$$P(l|\mathbf{v}) = \eta P_R(l|\mathbf{v}) + (1-\eta)P_T(l|\mathbf{v}). \tag{9}$$

Here η is the coefficient to control the relative significance between the temporal classifier and the RDF classifier. As the variation of hand motion speed is large and the performance of P_T can degrade a lot when the hand moves very fast, we make the significance of P_R outweigh the temporal term P_T so that their combination is robust to tracking failure, *i.e.* $\eta > 0.5$. In practice we find with a value of 0.5 the method seldom fails when parsing successive depth image sequences.

VI. Superpixel-MRF Hand Parsing

The hand parsing results produced by the per-pixel classification method in Section V are quite noisy since dependencies between neighboring pixels are not fully utilized. Simply applying the pixel-level filtering techniques, *e.g.*, the median filter, to the labeling results will not work well since many misclassified pixels form small isolated regions surrounded by other parts, as shown in Fig. 7. The Markov Random Fields [16] can be used to refine the classification results, which can well model the constraints from the neighboring states and the image observations [17]. However, the traditional pixel-based MRF is time consuming for real-time HCI applications. Besides, the isolated misclassified regions are more suitable to be represented as a whole rather than a set of pixels for MRF inference. Therefore, we partition the hand region into superpixels, and combine them and MRF inference to refine the parsed hand parts based on the per-pixel parsing results.



(a) (b) (c)

Fig. 7. Effectiveness of MRF inference based on superpixel partition and label co-occurrence. (a) Per-pixel classification result. (b) Superpixel partition result. (c) SMRF inference result.

The proposed superpixel-MRF framework is built based on both the posterior probability $P(l|\mathbf{v})$ given by per-pixel classification and the depth image. The superpixels are constructed with two criteria. First, the depth discontinuity must be conserved when determining the borders of neighboring superpixels. Second, pixels within one superpixel should have similar posterior probability. These principles are incorporated into the SLIC superpixel partition algorithm [18], and the resulting superpixel partition is compact in terms of 3D space distribution and posterior probability. The MRF network is then constructed for the superpixels with the similar way in [19].

Given the compact representation of the superpixels, they still behave differently from the pixels. Some superpixels have quite irregular shapes in order to conserve the depth discontinuity during superpixel partition. Also, the relative sizes of neighboring superpixels are sometimes uneven, as shown in Fig. 7(b). This suggests that some superpixels can have larger influence on their neighbors than others. Therefore, in the MRF framework, we model the interaction energies between superpixels based on their common borders, relative sizes and label co-occurrence distribution when determining the pairwise term. That is, if two neighboring superpixels have many common borders, they are quite likely to have the same label. Also, a small superpixel can be more easily smoothed by a big neighbor than opposite. We also take the idea of label co-occurrence from [38] so that the unsmoothness between the neighbors with low co-occurrence rate is high.

A. Superpixel Partition

Though the per-pixel classification result can be noisy, the labeled parts are mostly locally homogeneous. Besides, a large portion of the wrongly classified pixels form isolated small regions. Therefore, the labeled image is suitable to be represented by a set of super pixels for further processing, which can well model the misclassified regions as well as reduce the computational cost involved in the MRF energy minimization process. To this end, we modify the simple linear iterative clustering (SLIC) algorithm to get the superpixel partition [17]. The original SLIC method is developed for color images, while we need to get the superpixel partition that conforms to both the hand part classification results and the depth discontinuity. To be specific, the criteria for our superpixel partition scheme are:

1) Depth continuity: the depth difference between neighboring pixels within the superpixel is smaller than a threshold d_T.
2) Similar posterior probability: the pixels within a superpixel should have similar $P(l|\mathbf{v})$.

According to these requirements, it is not reasonable to apply the SLIC method directly to the posterior distribution $P(l|\mathbf{v})$, as the Euclidean distance in $P(l|\mathbf{v})$ does not make much sense. Thus, we perform superpixel partition in the space of posterior distributions and adopt the Kullback-Leibler divergence to measure the difference between each pixel and the superpixel cluster center. Let the set of superpixel partition be $S = s_1 \cup s_2 \cup \ldots \cup s_K$. The posterior probability and depth of each superpixel are taken as the average of all the pixels within the partition, that is:

$$d_{sk} = \frac{1}{|s_k|} \sum_{\mathbf{v}_j \in s_k} c_j \tag{10}$$

$$P(l|s_k) = \frac{1}{|s_k|} \sum_{\mathbf{v}_j \in s_k} P(l|\mathbf{v}_j), \tag{11}$$

where $d_(S_k)$ and $P(l|s_k)$ are the depth and posterior probability of the superpixel s_k. To perform superpixel clustering, we first define the distance metric in the posterior probability to measure the difference between the pixels and the superpixel cluster to be:

$$D_{kl}(s_k, \mathbf{p}_j) = \sum_{l=1}^{L} P(l|s_k) \log \frac{P(l|s_k)}{P(l|\mathbf{p}_j)}. \tag{12}$$

Besides, in order to preserve the depth discontinuity in the superpixel partition, the pixel $\mathbf{p}_j$ can be assigned to s_k only if $|c_j - d_{s_k}| \leq d_T$. In the implementation we set $d_T = 6$ mm. Also, the superpixel should be compact in the 2D image coordinate space, as in the original SLIC method. Therefore, we define the distance metric for superpixel partition as:

$$D = \begin{cases} \sqrt{D_{kl} + \left(\frac{D_S}{M_S}\right)^2 \gamma^2} & if|c_j - d_{s_k}| \leq d_T \\ \infty & otherwise \end{cases} \tag{13}$$

where M_S is the regular grid step on the image plane to determine the superpixel size; D_S is the pixel distance between $\mathbf{p}_j$ and the superpixel center; γ controls the relative importance of the two terms. Based on the distance metric D, the superpixel partition is performed by the clustering scheme in the SLIC method. The posterior probability of these superpixels forms the observation data for the following MRF inference stage. Note that the sizes of the superpixels are mostly quite close, except that the wrongly labeled regions are generally small or have irregular shapes. This indicates such misclassified parts are more likely to be neutralized by their neighboring superpixels.

B. Superpixel-MRF Inference

Given the set of superpixels $S = \{s_k\}$, the goal for MRF inference is to assign each superpixel a new label $l \in L$. Let the associated labels for the superpixels be $Y = \{y_k\}$. The task for MRF inference of the labels is to get the MAP solution of $Y^* =$

$\arg\max_Y P(Y|S)$, which is equivalent to the minimization of the following energy function:

$$E = E_d + E_S = \sum_{i \in S} \phi_i(y_i) + \sum_{i \in S, j \in N_i} \psi_{i,j}(y_i, y_j). \tag{14}$$

Here E_d is the unary term to measure the discrepancy between the inferred label and the per-pixel classification results, and we set $\phi_i(y_i) = -\log P(y_i|s_i)$.

The pairwise term E_s is used to measure the smoothness between neighboring superpixels, and we utilize the idea of label co-occurrence [38] to define $\psi_{i,j}$. The label co-occurrence represents the conditional probability $P(y_i|y_j)$, which indicates how likely a superpixel with state y_j will have a neighbor with a state y_i. Since some hand parts are more likely to be adjacent than others, *e.g.* the chances that the part 1 and 2 are adjacent are higher than the part 1 and 6, the label co-occurrence can be useful during inference by punishing the unlikely adjacent states. However, unlike [38] in which a superpixel in the color images is equally affected by all its neighbors, we take the depth discontinuity and the irregular shapes of the superpixels into consideration to model the pairwise interaction energy. First, the depth discontinuity between the superpixels is used to define the adjacency of a pair of nodes. The nodes (y_i, y_j) are adjacent only if the following two criteria are met:

$$\Omega_i \cap \Omega_j \neq \emptyset, \tag{15}$$

$$|d_{s_i} - d_{s_j}| \leq d_T \tag{16}$$

where Ω is the set of border pixels of the superpixel. With these criteria, there is a pairwise energy term between two nodes only if they are neighboring superpixels and have similar depths. In addition, some superpixels can have quite irregular shapes, which make them interact with their neighbors in an uneven way. Specifically, a superpixel is more likely to take the same label with the neighbors that share more borders than others, and small superpixels are more likely to take the same label with its big neighbors than the opposite way. This is especially significant for the wrongly labeled regions, which are usually isolated and small. For two adjacent nodes i and j, the uneven influences resulting from these factors should be reflected in the pairwise energy term, in addition to their state differences (y_i, y_j). Thus, we define a weight coefficient $\alpha_{i,j}$ for the pairwise term of adjacent nodes, which is given by:

$$\alpha_{i,j} = \frac{|\Omega_i \cap \Omega_j|}{|\Omega_i|} \times \frac{|s_j|}{|s_i|}. \tag{17}$$

A big value of $\alpha_{i,j}$ indicates the superpixel i is more likely to be affected by its neighbor j.

Fig. 7 shows an example to illustrate the effectiveness of modeling the pairwise energy based on the superpixel partition results. Fig. 7(b) shows the superpixel partition of the per-pixel classification results in Fig. 7(a). As the pixels within each superpixel are assigned the same state, the small and scattered misclassified points are largely suppressed even without MRF inference. However, the larger misclassified regions still cannot be removed, *e.g.* the regions labeled with 1 and 3 in the red rectangles. Region 1 is misclassified to No. 4 hand part and region 3 is misclassified to No. 5 hand part. According to the co-occurrence probability of different hand parts in Fig. 8, both these misclassified regions result in a large pairwise energy. First consider region 3. It is small in relative size compared to its surrounding superpixels, *e.g.* region 4, thus assigning it with the label of its neighbors will produce a big energy decrease, which is favored during inference. For region 1, note it is only adjacent to the superpixels on the middle finger since depth discontinuity is handled to build the MRF framework, thus it will be influenced by only one misclassified superpixel and multiple correctly classified regions in the middle finger. By comparison, its adjacent region 2, which is correctly classified, has much more correctly classified neighbors. Again, region 1 is more inclined to be neutralized by region 2. Fig. 7(c) shows the results given by MRF inference, in which the misclassified regions are successfully removed.

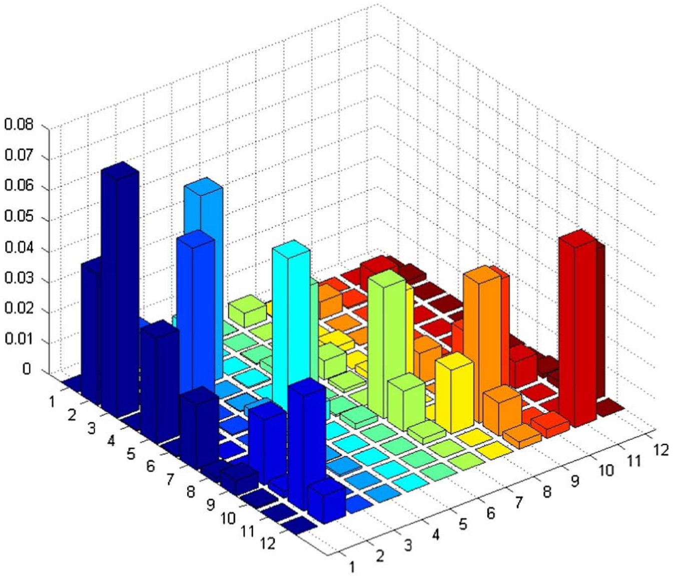


Fig. 8. The co-occurrence probability distribution of the labels of the neighboring superpixels for $M_S = 4$.

Combing the label co-occurrence, node adjacency and weight coefficient $\alpha_{i,j}$, the pairwise potential function $\psi_{i,j}$ for two adjacent node i and j takes the following form:

$$\psi_{i,j}(y_i, y_j) = -\alpha_{i,j} \times [1 - \delta(y_i, y_j)] \times \log\left[\frac{P(y_i|y_j) + P(y_j|y_i)}{2}\right] \tag{18}$$

where δ is the Kronecker delta function to indicate a zero pairwise energy if the node i and j have the same state. $P(y_i|y_j)$ is the conditional probability learned from the training dataset by counting the co-occurrence of the labels of neighboring superpixels. The co-occurrence distribution of the labels is shown in Fig. 8. Note here all the $P(y_i|y_i)$ terms are set to zero as they have no effect on the inference results. By incorporating the co-occurrence probability distribution in SMRF inference, we encode the prior to eliminate the unlikely neighboring labels to smooth the results given by per-pixel classification. Based on the above formulation, we adopt the iterated conditional modes (ICM) algorithm [20] to minimize the energy function to get Y^*, and the initial label for each superpixel is taken to be $y_k =$

$\arg\max_l P(l|s_k)$. The ICM algorithm is a greedy search algorithm, and is guaranteed to converge fast.

C. Computational Complexity Analysis

Overall, the computational cost involved in the proposed hand parsing scheme consists of four parts: per-pixel classification, superpixel partition, MRF network construction and MRF inference. Let the number of pixels in the hand region be n. The RDF classifier is known to have a computational cost of $O(nND_R)$, where D_R is the tree depth. The temporal classifier has a complexity of $O(n)$ as it only involves calculation of the twelve Gaussian terms for each pixel. For superpixel partition with the SLIC algorithm, the complexity is $O(nK_{SLIC})$. We explicitly write out the number of iterations K_{SLIC} for SLIC to converge rather than $O(n)$ [18], as K_{SLIC} is important for the final partition quality.

On average the number of partitioned superpixels is n/M_S^2. To construct the MRF network, the weight coefficient $\alpha_{i,j}$ needs to be calculated for each pair of adjacent nodes, and the average number of neighbors for each node is a constant approximately between four and eight. This results in a complexity of $O(n/M_S^2)$. Finally, the complexity of the ICM algorithm for MRF inference is known to be linear to the number of nodes, *i.e.* $O(nK_{ICM}/M_S^2)$, where K_{ICM} is the number of iterations needed to converge. Thus the overall complexity of the proposed parsing scheme is:

$$C_{Total} = O\left\{n(ND_R + K_{SLIC})\right\} + O(nK_{ICM}/M_S^2). \quad (19)$$

Note that the parameters except n can all be predefined, thus the overall complexity is indeed linear with the number of the pixels.

VII. Experimental Results

In this section we present the results to validate the effectiveness of the proposed DCA feature and the SMRF framework by comparison to the state-of-art methods [8], [28], [40]. For hand parsing, the experiments include the tests on single-frame datasets, continuous hand motion sequences and real-world hand motion sequences. To further test the generalization power of the proposed methods to other articulated objects such as the human body, we also present the results to fulfill the human body segmentation task on a public body part annotation dataset [40]. The performances of the proposed method are compared to the state-of-art methods. The whole program was coded in C++/OPENCV without parallelization, and tested on a PC with Intel i5 750 CPU and 4G RAM.

A. Quantitative Evaluation for Single Frames

We synthesized a dataset of 22.5k templates to quantitatively evaluate the performance of our method on the single frames. The resolution of the images is 320 × 240. Each template consists of a pair of depth image and the ground truth hand part labels. To generate the dataset, we capture a set of hand articulation parameters by the CyberGlove II [7]. The captured hand articulation parameters are combined with the 3D global rotation parameters in certain ranges to handle the viewpoint variation. Here we define the ranges to be $(-20°, 20°)$ for global rotation around the X and Y axes, i.e. the axes parallel to the image plane of the camera, and $(-35°, 35°)$ around the Z axis, i.e. the axis perpendicular to the image plane. These hand motion parameters are used to drive the 3D hand model in Section IV to generate the templates. To evaluate the classification accuracy on the synthesized images, we use 80% of the templates in the dataset for training and the rest 20% for testing.

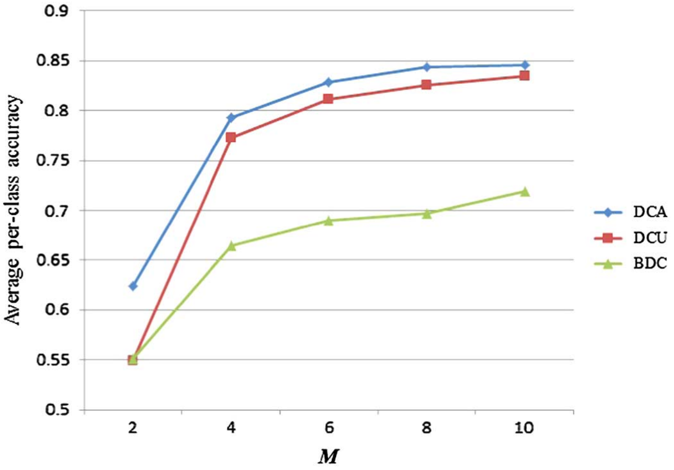


Fig. 9. Comparison of the average per-class accuracy of the three depth features with different values of M.

First, we illustrate that the depth context feature with the adaptive sampling scheme (DCA) achieves considerably higher accuracy than the benchmark binary depth comparison feature (BDC) [8]. The feature value in BDC is calculated by the binary depth comparison of two neighboring context points, *i.e.* $f_F = d_z(\mathbf{p} + \mathbf{u}/d_z(\mathbf{p})) - d_z(\mathbf{p} + \mathbf{v}/d_z(\mathbf{p}))$, where $\mathbf{u}$ and $\mathbf{v}$ are a pair of relative offsets of the context points. As in [8], the offset pairs of $\mathbf{u}$ and $\mathbf{v}$ are randomly sampled. In the experiments we set the number of offset pairs in BDC and the number of context points in DCA to be the same so that the resulting features have the identical dimension. Besides, we also test the performance of the depth-context feature with uniform sampling (DCU), *e.g.* $g(x)$ is a constant function. Fig. 9 shows the per-class classification accuracy of the three methods with respect to different number of M, in which the RDF consists of three trees with a maximum depth of 20. The DCA feature outperforms the benchmark BDC feature by 12.2%, in which the distance-adaptive sampling scheme contributes 2.8% improvement. Especially, the DCA method performs quite well even with a very small value of M, *i.e.* $M = 2$. This proves the capability of the distance-adaptive sampling scheme to capture the 3D context of the pixels in depth image. Fig. 10 shows the per-class classification accuracy with different depth of the RDF at fixed value of $M = 10$. We can see that the DCA method again provide better classification results for all choices of the tree depths.

To validate the effectiveness of the SMRF framework compared to per-pixel classification, we have combined it to the RDF classifier with both the BDC feature and the DCA feature and test the performances on the same dataset. The results are presented in Table I. The label "SMRF-#" represents our method running with different superpixel grid step M_S. The feature grid

TABLE I
COMPARISON OF THE RESULTS OF PER-PIXEL CLASSIFICATION WITH RDF, SMRF INFERENCE WITH DIFFERENT SUPERPIXEL SIZES AND THE MULTI-LAYERED RDFs

Method	Avg. Accuracy	Avg. RDF Time (ms)	Avg. SP Time (ms)	Avg. MRF Time (ms)	Avg. Total Time (ms)
Per-pixel RDF with BDC [8]	71.94%	69.3	-	-	69.3
SMRF-1 with BDC	74.51%	69.1	-	386.4	483.4
SMRF-4 with BDC	**77.40%**	69.2	198.2	20.0	291.0
SMRF-8 with BDC	77.31%	68.9	179.4	6.8	257.1
SMRF-12 with BDC	75.62%	69.9	163.4	4.8	239.8
Per-pixel RDF with DCA	**84.53%**	72.9	-	-	72.9
SMRF-1 with DCA	87.30%	72.7	-	380.3	480.8
SMRF-4 with DCA	**89.29%**	72.8	181.2	19.6	277.1
SMRF-8 with DCA	88.82%	73.1	166.4	7.0	248.5
SMRF-12 with DCA	87.01%	73.0	151.2	4.6	230.6
Multi-layered RDF with BDC [28]	72.14%	235.2	-	-	235.2

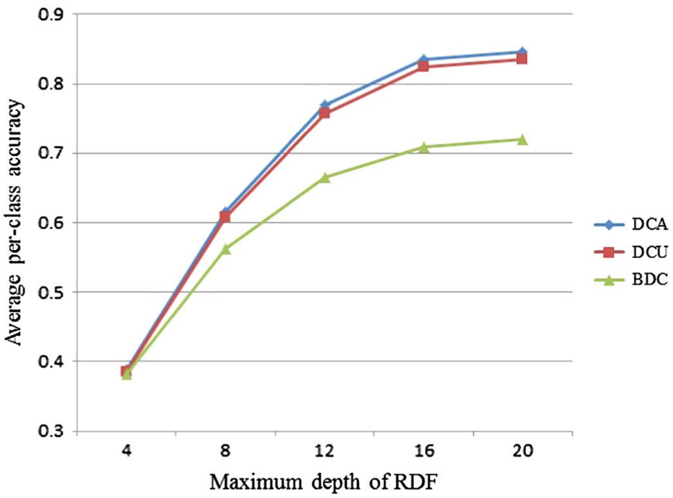


Fig. 10. Comparison of the average per-class accuracy of the three depth features with different values of the maximum depth of RDF.

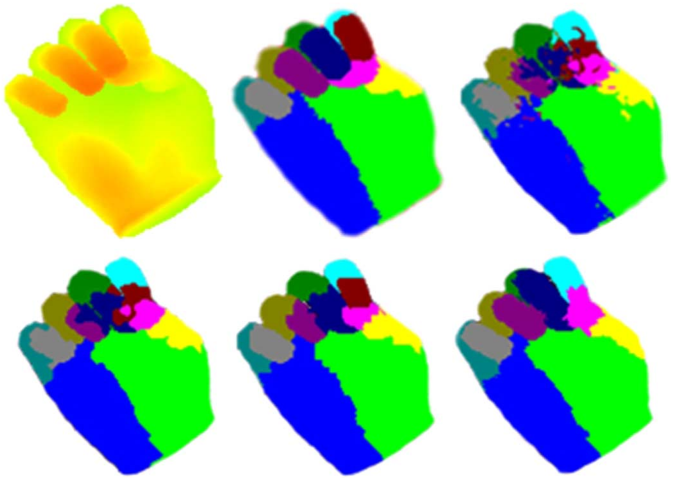


Fig. 11. Illustration of SMRF performance with different superpixel sizes. Upper row: the input depth image (Left), the ground truth hand part labels (Middle) and per-pixel classification with DCA (Right). Lower row: SMRF results with $M_S = 1$ (Left), SMRF with $M_S = 4$ (Middle) and SMRF with $M_S = 12$ (Right).

size is $M = 10$ for DCA, and equivalently the dimension of the BDC feature is 440. Table I also shows the per-frame time cost for per-pixel classification and the superpixel partition (SP) and MRF inference (MRF) modules of SMRF. Overall, the SMRF method provides the highest increase of 5.5% with $M_S = 4$ for the BDC feature at an extra time cost of 218.2 ms, and 4.8% with $M_S = 4$ for the DCA feature at an extra time cost of 200.8 ms. Compared to the benchmark per-pixel classification with RDF and the BDC feature [8], we achieved an overall 17.4% higher accuracy for the twelve hand part classification for $M_S = 4$.

The results for the SMRF method with different M_S also validates that superpixel is a more appropriate representation to remove the misclassified regions than the pixel. Note that "SMRF-1" is equivalent to per-pixel inference with MRF, and the result is not as good as the superpixel-level counterparts, even at a higher extra time cost of about 380 ms per frame. This is because at the pixel-level, a pixel within the isolated misclassified region will be influenced equally by its surrounding pixels, among which there are also the misclassified pixels. By comparison, at the superpixel-level, a misclassified superpixel is mostly surrounded by the superpixels with the correct labels, and thus are more likely to be converted by its neighbors. Besides, according to the results in Table I, the size of the superpixel cannot be too large, as it produces over-smoothing effects. Fig. 11 shows an example of applying SMRF with different values of M_S. Not the finger regions, we can see the isolated misclassified regions are not well suppressed with $M_S = 1$, while the thumb and index fingers are wrongly smoothed for $M_S = 12$. The size $M_S = 4$ produces the result that best conforms to the ground truth.

We also implement the Multi-layered Random Decision Forests [28] to parse the hand based on the hand part partition scheme in Fig. 5. In this approach, the hand configuration parameters of the training data are first clustered into K classes by spectral clustering. During training, a RDF classifier is learned for hand shape classification (SCF), which classifies the whole depth image of the hand into the K classes. An individual RDF classifier (GEN) is trained for hand parsing on each of the K subsets of the original dataset. During testing, the SCF classifier is first applied to the input hand depth image, and its result is used to pick up the GEN classifiers for hand parsing. The BDC feature is used for classification in both the SCF and GEN classifiers. In [28] they report that this method improves the classification accuracy to 91.2% on their own dataset, compared

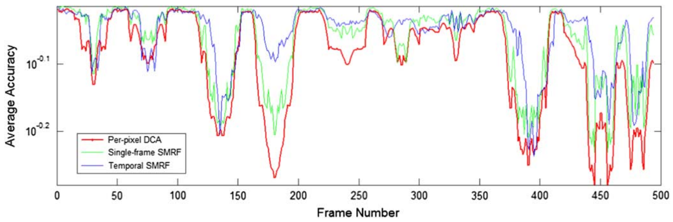


Fig. 12. Comparison of the hand parsing results using per-pixel classification with DCA, single-frame SMRF and SMRF with temporal reference on the first synthesized continuous hand motion sequence.

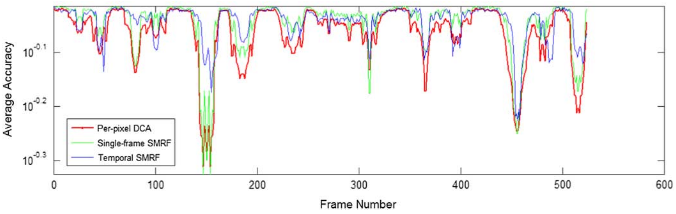


Fig. 13. Comparison of the hand parsing results using per-pixel classification with DCA, single-frame SMRF and SMRF with temporal reference on the second synthesized continuous hand motion sequence.

to 68.0% obtained by a single RDF classifier [8]. As in [28], we set $K = 25$, and the SCF classifier is used to determine the three most probable classes of the test image. The corresponding three GEN classifiers are picked up and their results are weighted to give the final parsing. All the RDF classifiers consist of three trees with a maximum depth of 20. The result obtained with the Multi-layered RDF classifier on our dataset is presented in Table I, which shows very little improvement compared to [8], *i.e.* 72.14% vs. 71.94%. This is not surprising as our dataset consists of a large number of natural hand configurations and the hand pose parameters do not form specific clusters. Therefore, the hand pose parameters within each individual pose cluster still contain large variations, and thus the SCF/GEN framework in [28] cannot work well. By contrast, our proposed SMRF framework shows better generalization capability and gives 5.5% improvement compared to [8].

B. Quantitative Evaluation for Continuous Sequences

To evaluate how the temporal reference can help to improve the hand parsing results, we synthesized two continuous hand motion sequences with the same procedure in Section VII-A, both of which are approximately 500 frames long and contain complex combinations of global hand motion and local finger articulations. The first sequence includes the continuous motion of the hand when it is posing digit gestures, including the transition motion to change from one gesture to another. The 3D hand rotation and transition are also included. The second sequence contains different single/multiple finger motions combined with 3D hand rotation. Especially, these two testing sequences contain some clips in which the hand motions are not covered by the training dataset in order to test the robustness of the proposed parsing scheme.

TABLE II
COMPARISON OF THE CLASSIFICATION RESULTS ON THE TWO SYNTHESIZED HAND MOTION SEQUENCES

	Per-pixel BDC [8]	Per-pixel DCA	Single-frame SMRF	Temporal SMRF
Seq. 1	77.2%	82.4%	87.1%	89.6%
Seq. 2	81.7%	86.8%	90.8%	91.5%

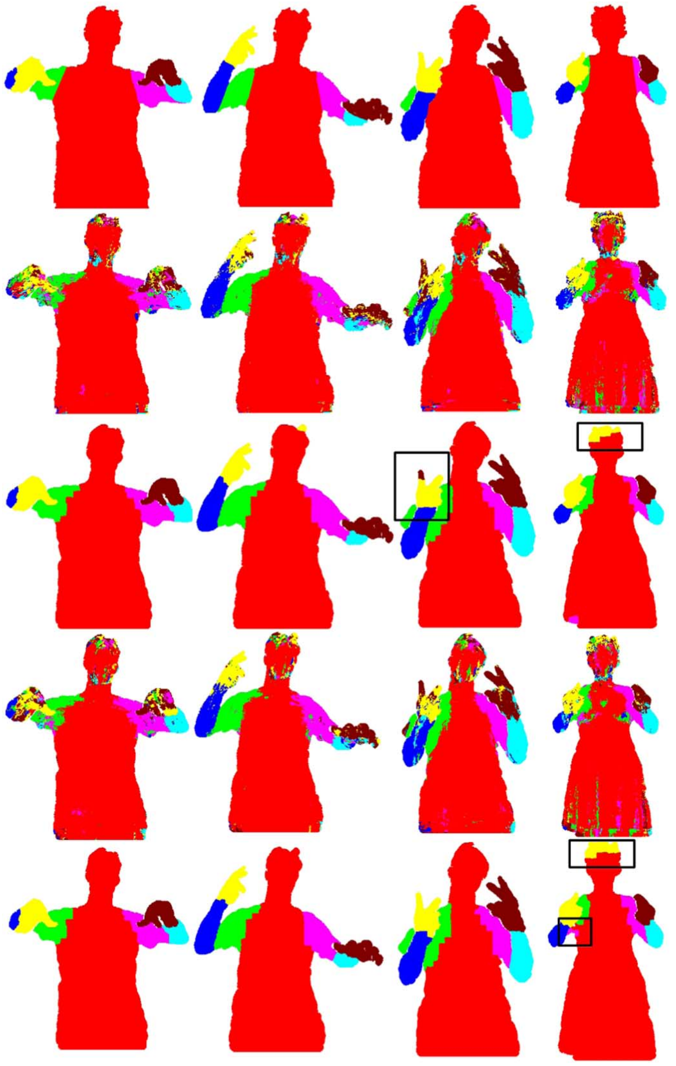


Fig. 14. Comparison of the human body part segmentation results. Ground truth (first row), per-pixel classification with BDC and RDF (second row), SMRF with BDC (third row), per-pixel classification with DCA and RDF (fourth row), SMRF with DCA (fifth row). The black rectangles show the failure cases of SMRF.

In this experiment we set the weighting parameter to fuse the RDF classification and the temporal reference to be $\eta = 0.5$, and the superpixel grid size is $M_S = 4$. The per-frame accuracy curves for the two sequences are shown in Fig. 12 and Fig. 13, in which we compared three methods: per-pixel classification with the proposed DCA feature, the proposed single-frame SMRF with the DCA feature and the temporal SMRF with the DCA feature. Note the average accuracy is shown in log-scale for better illustration. The average accuracies on the whole sequences are summarized in Table II, in which the per-pixel classification results with the BDC feature [8] are also included. For the two sequences, the singer-frame SMRF improves the per-pixel classification results with DCA by 4.7% and 4.0% respectively. The temporal SMRF achieves further

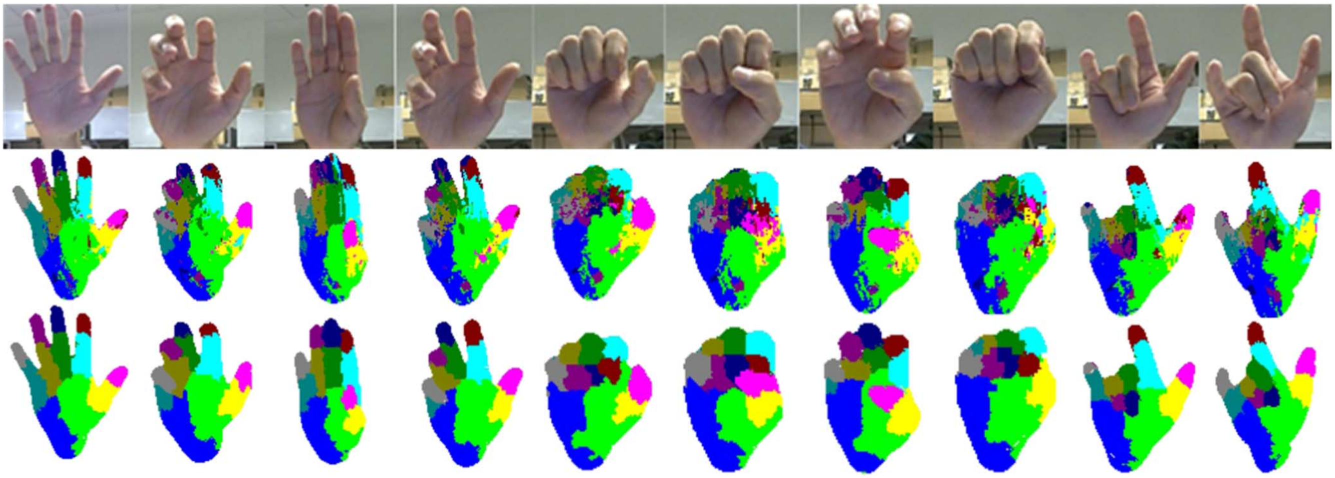


Fig. 15. Comparison of the hand parsing results using per-pixel classification with DCA and the RDF classifier (middle row), and the proposed Temporal SMRF framework (lower row) on real-world hand motion sequences.

TABLE III
COMPARISON OF BODY PART SEGMENTATION RESULTS BETWEEN OUR METHODS AND [40]

Method	Torso	LU arm	LW arm	L hand	RU arm	RW arm	R hand	Avg. Accuracy
Per-pixel RDF with BDC [40]	94.06%	79.81%	78.69%	76.59%	81.18%	83.10%	80.23%	81.95%
Frame-by-Frame Graph Cut [40]	98.86%	75.03%	83.36%	92.41%	77.54%	87.67%	94.20%	87.01%
Temporally Coherent Graph Cut [40]	98.44%	78.93%	84.38%	88.32%	82.57%	88.85%	93.86%	87.91%
Per-pixel RDF with BDC (Ours)	86.52%	88.85%	84.70%	71.42%	87.40%	81.33%	69.78%	81.43%
SMRF-12 with BDC (Ours)	94.10%	93.57%	90.43%	87.31%	93.13%	87.84%	85.79%	**90.31%**
Per-pixel RDF with DCA (Ours)	87.22%	90.11%	87.33%	72.77%	89.43%	84.18%	70.49%	83.08%
SMRF-12 with DCA (Ours)	94.36%	93.27%	91.71%	85.45%	93.33%	88.86%	82.38%	89.91%

1.5% and 0.7% improvement respectively on the two sequences, compared to the single-frame SMRF.

While the improvement obtained by the temporal reference seems not striking in terms of the average accuracy, it is important to see that it largely increases the overall robustness. Since some parts of the testing sequences are either not covered by the training dataset or are going through complex hand motion and self-occlusion, the performances by per-pixel classification with the pre-trained classifier and the SMRF method that built upon it can suffer from drastic degradation, *e.g.* frames between 150 and 200 in Seq. 1 and frames between 130 and 160 in Seq. 2. In these parts we see the system performance is dramatically improved by fusing the temporal information, and thus the resulting overall classification accuracy remains much more stable than the single-frame based counterparts.

C. Overall Qualitative Evaluation

We further test the parsing performance of the SMRF framework on real-world input sequences captured by a SoftKinetic DS325 depth camera, which is about 900 frames long, and the result is illustrated in Fig. 15. The parsed hand parts are shown with different colors, which is consistent with the labeling scheme in Fig. 5(a). The results show the effectiveness of the SMRF method. From the figure we can see that the results of per-pixel parsing are very noisy. In addition, its performance for small finger parts get even worse for the challenging cases such as tightening fingers or when some fingers are occluded. By comparison, the SMRF method produces more meaningful parsing in such cases.

D. Human Body Part Segmentation

We adapt the proposed hand parsing scheme to human body part segmentation to demonstrate its generalization power to different kinds of articulated objects, and test it on the body part annotation dataset in [40]. The idea of the body part segmentation method in [40] is close to ours in that they also improve the RDF classification results by utilizing the spatial-temporal context in a graph cut optimization framework. The dataset consists of 500 frames of annotated human body parts and the corresponding depth and color images. The resolution of the images is 640×480. The body partitions consist of seven parts: the torso, the left/right upper arms (LU/RU), the lower arms (LW/RW) and the hands (LH/RH). In this experiment we compare the per-pixel classification results with the RDF classifier using the BDC and DCA features separately, based on which the SMRF framework is also tested. We use the same setting as [40], *i.e.* we perform 5-fold cross-validation on the dataset, and select 1000 pixels per image for training. The RDF consists of 3 trees and each tree has a maximum depth of 20. Due to the larger resolution of the images, the superpixel size is set to be $M_S = 12$ for SMRF inference. For this experiment we do not utilize the temporal references.

The performances of the proposed parsing scheme are compared with [40] in Table III, in which we also cite their best results with three different methods. The first is their implementation of RDF classification with the BDC feature. The second is frame-by-frame graph-cut optimization, which uses the single-frame information. The third is temporal-coherent graph-cut optimization, which incorporates multiple frames for inference and reports the highest accuracy of 87.91%. Note that our implementation of the RDF classification using the BDC feature achieves similar average accuracy to [40], *i.e.* 81.43% vs. 81.95%, while the classification accuracies are different for the individual classes. This can be a result of our different selection strategy of the training samples. In our implementation to train the RDF classifier, we select the 1000 sample pixels from each training image by choosing the same number of pixels from each class, and thus the training samples are balanced for all the classes. In [40], the 1000 samples are selected uniformly in the whole image. As the torso part occupies the largest area and the number of torso samples in the training data is thus more than the others, their trained RDF classifier gives much higher accuracy for the torso over the other parts.

Overall, the results in Table III show the effectiveness of the proposed SMRF method. With the BDC feature, the SMRF method improves the per-pixel results by 8.88% from 81.43% to 90.31%. By comparison, the frame-by-frame graph cut method in [40] achieves 5.06% increase compared to per-pixel classification. By inference in multiple frames, their temporal-coherent graph cut method achieves 5.96% increase. The results with the DCA feature are quite close to that of the BDC feature though. The reason may be due to the difference between the hand and the body, *e.g.* the hand has more serious self-occlusion and can rotate freely around all three axes. Several body segmentation examples are given in Fig. 14. The first row is the ground truth annotations. The second and fourth rows are the per-pixel classification results with the BDC and DCA features respectively, and the third and fifth rows are the improved results with SMRF respectively. In the black rectangles we show some failure cases of the SMRF method. In these cases the per-pixel results are too noisy to recover the correct label.

VIII. Conclusion

In this paper we developed a novel hand parsing scheme with the input of depth images. Our contributions mainly include a depth context feature with the distance-adaptive sampling scheme (DCA) which proves more accurate for hand parsing compared to the binary depth comparison feature (BDC) [8], and a Superpixel-MRF framework (SMRF) to enforce both the spatial and temporal constraints to improve the per-pixel classification results. The results on both the synthesized depth images and real-world test sequences demonstrate the good performance of the proposed hand parsing scheme compared to the state-of-the-art methods [8], [28]. The tests on the human body annotation dataset further demonstrate the generalization power of our method to different kind of articulated objects. The results show the proposed SMRF framework outperforms the pixel-level graph cut optimization framework [40] that also enforces the spatial and temporal constraints.

The future work will focus on enlarging the training dataset for the RDF classifier to enable the hand to move in larger viewpoint variations. Our preliminary results have shown the proposed hand parsing scheme's capability to allow the user to rotate their hand freely in the plane parallel to the camera's image plane while maintaining high accuracy. Yet we still need to add considerably more training samples to allow free 3D hand rotation. In addition, we will also focus on how to recognize the hand gestures and restore the full DOF hand motion based on the parsed hand parts.

References

[1] J. P. Wachs, M. Kolsch, H. Stern, and Y. Edan, "Vision-based hand-gesture applications," *Commun. ACM*, vol. 54, no. 2, pp. 60–71, Feb. 2011.

[2] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, "Robust part-based hand gesture recognition using Kinect sensor," *IEEE Trans. Multimedia*, vol. 15, no. 5, pp. 1110–1120, Aug. 2013.

[3] Z. Li and R. Jarvis, "Real time hand gesture recognition using a range camera," in *Proc. ACRA*, 2009.

[4] M. Van den Bergh, E. Koller-Meier, F. Bosche, and L. Van Gool, "Haarlet-based hand gesture recognition for 3D interaction," in *Proc. WACV*, 2009.

[5] X. Shen, G. Hua, L. Williams, and Y. Wu, "Dynamic hand gesture recognition: An exemplar-based approach from motion divergence fields," *Image Vision Comput.*, vol. 30, no. 3, pp. 227–235, Mar. 2012.

[6] Y. Yao and Y. Fu, "Real-time hand pose estimation from RGB-D sensor," in *Proc. ICME*, 2012.

[7] CyberGlove [Online]. Available: http://www.cyberglovesystems.com.

[8] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, and M. Finocchio, "Real-time human pose recognition in parts from single depth images," in *Proc. CVPR*, 2011.

[9] J. Heikkila, "A four-step camera calibration procedure with implicit image correction," in *Proc. CVPR*, 1997.

[10] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[11] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly, "A Review on vision-based full DOF hand motion estimation," in *Proc. CVPR*, 2005.

[12] J. Lin, W. Ying, and T. S. Huang, "Modeling the constraints of human hand motion," in *Proc. HUMO*, 2000.

[13] J. P. Lewis, M. Cordner, and N. Fong, "Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation," in *Proc. SIGGRAPH*, 2000.

[14] R. Y. Wang and J. Popovic, "Real-time hand-tracking with a color glove," *ACM Trans. Graph.*, vol. 28, no. 3, Aug. 2009.

[15] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 5, pp. 698–700, May 1987.

[16] R. Kindermann and J. L. Snell*, Markov Random Fields and Their Applications*. Providence, RI, USA: Amer. Math. Soc., 1980.

[17] M. Fauvel, J. Chanussot, and J. A. Benediktsson, "SVM- and MRF-based method for accurate classification of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 4, pp. 736–740, Oct. 2010.

[18] A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[19] X. Wang, "A new localized superpixel Markov random field for image segmentation," in *Proc. ICME*, 2009.

[20] J. Besag, "On the statistical analysis of dirty pictures," *J. Roy. Statist. Soc., ser. B*, vol. 48, no. 3, pp. 259–302, 1986.

[21] M. K. Bhuyan, V. V. Ramaraju, and Y. Iwahori, "Hand gesture recognition and animation for local hand motions," *Int. J. Mach. Learn. Cyber.*, Mar. 2013.

[22] H. Wang, M. C. Leu, and C. Oz, "American sign language recognition using multi-dimensional hidden Markov models," *J. Inf. Sci. Eng.*, 2006.

[23] J. Choi, J. Park, H. Park, and J. Park, "iHand: An interactive bare-hand-based augmented reality interface on commercial mobile phones," *Opt. Eng.*, vol. 52, no. 2, Feb. 2013.

[24] C. Kerdvibulvech and H. Saito, "Model-based hand tracking by chamfer distance and adaptive color learning using particle filter," in *Proc. 2009 EURASIP*.

[25] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[26] S. Lu, D. Metaxas, D. Samaras, and J. Oliensis, "Using multiple cues for hand tracking and model refinement," in *Proc. 2003 CVPR*.

[27] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. ICRA*, Kobe, Japan, 2009.

[28] C. Keskin, F. Kirac, Y. E. Kara, and L. Akarun, "Hand pose estimation and hand shape classification using multi-layered randomized decision forests," in *Proc. ECCV*, 2012.

[29] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari, “2D articulated human pose estimation and retrieval in (almost) unconstrained still images,” *Int. J. Comput. Vision*, vol. 99, no. 2, pp. 190–214, 2012.

[30] C. Keskin, F. Kirac, Y. E. Kara, and L. Akarun, “Real-time hand pose estimation using depth sensors,” in *Proc. ICCV*, 2011.

[31] B. Dorner, “Chasing the color glove: Visual hand tracking,” Master’s thesis, Simon Fraser Univ., Burnaby, BC, Canada, 1994.

[32] A. Aristidou and J. Lasenby, “Motion capture with constrained inverse kinematics for real-time hand tracking,” in *Proc. ISCCSP*, 2010.

[33] A. Baak, M. Muller, G. Bharaj, H. P. Seidel, and C. Theobalt, “A data-driven approach for real-time full body pose reconstruction from a depth camera,” in *Proc. ICCV*, 2011.

[34] L. Bo, X. Ren, and D. Fox, “Depth kernel descriptors for object recognition,” in *Proc. IROS*, 2011.

[35] D. Ramanan, “Learning to parse images of articulated bodies,” in *Proc. NIPS*, 2007.

[36] L. Zhu, Y. Chen, Y. Lu, C. Lin, and A. Yuille, “Max margin AND/OR graph learning for parsing the human body,” in *Proc. CVPR*, 2008.

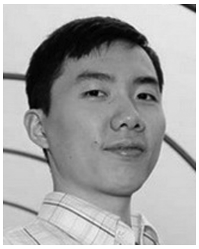
[37] Z. Ren, J. Yuan, C. Li, and W. Liu, “Minimum near-convex decomposition for robust Shape Representation,” in *Proc. ICCV*, 2011.

[38] J. Tighe and S. Lazebnik, “SuperParsing: Scalable nonparametric image parsing with superpixels,” in *Proc. ECCV*, 2010.

[39] H. Liang, J. Yuan, D. Thalmann, and Z. Zhang, “Model-based hand pose estimation via spatial-temporal hand parsing and 3D fingertip localization,” *Visual Comput. J.*, vol. 29, no. 6-8, pp. 837–848, Jun. 2013.

[40] A. Hernandez-Vela, N. Zlateva, A. Marinov, M. Reyes, P. Radeva, D. Dimov, and S. Escalera, “Graph cuts optimization for multi-limb human segmentation in depth maps,” in *Proc. CVPR*, 2012.

Hui Liang received the B.S. and M.S. degrees in Electronics and Information Engineering from Huazhong University of Science & Technology (HUST), Wuhan, China, in 2008 and 2011, respectively. He is currently pursuing the Ph.D. degree at Nanyang Technological University, Singapore. His research interests include computer vision and hand-based human computer interaction.



Junsong Yuan is a Nanyang Assistant Professor at Nanyang Technological University, leading the video analytics program at School of EEE. He obtained his PhD from Northwestern University, M.Eng. from National University of Singapore, and B.Eng. from special class for the gifted young at Huazhong University of Science and Technology, China. He has co-authored over 100 technical papers and filed 3 US patents and 2 provisional US patents. He received Outstanding EECS Ph.D. Thesis award from Northwestern University and Doctoral Spotlight Award from IEEE Conf. Computer Vision and Pattern Recognition Conference (CVPR’09). He is Organizing Chair of Asian Conf. on Computer Vision (ACCV’14), and co-chairs workshops at CVPR’12’13 and ICCV’13. He also serves as Area Chair for WACV’14, ACCV’14, ICME’14, and is an associate editor of Visual Computer Journal and Journal of Multimedia. He gave tutorials at IEEE ICIP’13, FG’13, ICME’12, SIGGRAPH VRCAI’12, and PCM’12.

Daniel Thalmann is with the Institute for Media Innovation at the Nanyang Technological University in Singapore. He is a pioneer in research on Virtual Humans. He has been the Founder of VRlab at EPFL, Switzerland, Professor at The University of Montreal and Visiting Professor/Researcher at CERN, University of Nebraska, University of Tokyo, and National University of Singapore. He is coeditor-in-chief of the Journal of Computer Animation and Virtual Worlds, and member of the editorial board of 6 other journals. Daniel Thalmann was Program Chair and CoChair of several conferences including IEEE VR, ACM VRST, ACM VRCAI, CGI, and CASA. Daniel Thalmann has published more than 500 papers in Graphics, Animation, and Virtual Reality. He is coeditor of 30 books, and coauthor of several books including ‘Crowd Simulation’ (second edition 2012). He received his PhD in Computer Science in 1977 from the University of Geneva and an Honorary Doctorate from University Paul- Sabatier in Toulouse, France, in 2003. He also received the Eurographics Distinguished Career Award in 2010 and the 2012 Canadian Human Computer Communications Society Achievement Award.