Vis Comput (2013) 29:837–848
DOI 10.1007/s00371-013-0822-4

ORIGINAL ARTICLE

Model-based hand pose estimation via spatial-temporal hand parsing and 3D fingertip localization

Hui Liang · Junsong Yuan · Daniel Thalmann · Zhengyou Zhang

Published online: 8 May 2013
© Springer-Verlag Berlin Heidelberg 2013

Abstract In this paper we present a novel vision-based markerless hand pose estimation scheme with the input of depth image sequences. The proposed scheme exploits both temporal constraints and spatial features of the input sequence, and focuses on hand parsing and 3D fingertip localization for hand pose estimation. The hand parsing algorithm incorporates a novel spatial-temporal feature into a Bayesian inference framework to assign the correct label to each image pixel. The 3D fingertip localization algorithm adapts a recently developed geodesic extrema extraction method to fingertip detection with the hand parsing algorithm, a novel path-reweighting method and K-means clustering in metric space. The detected 3D fingertip locations are finally used for hand pose estimation with an inverse kinematics solver. Quantitative experiments on synthetic data show the proposed hand pose estimation scheme can accurately capture the natural hand motion. A simulated water-oscillator application is also built to demonstrate the effectiveness of the proposed method in human-computer interaction scenarios.

Keywords Fingertip detection · Geodesic distance · Hand pose estimation · Human computer interaction

H. Liang (✉)
Institute for Media Innovation & School of EEE, Nanyang Technological University, 50 Nanyang Drive, Singapore 637553, Singapore
e-mail: hliang1@e.ntu.edu.sg

J. Yuan
School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798, Singapore
e-mail: jsyuan@ntu.edu.sg

D. Thalmann
Institute for Media Innovation, Nanyang Technological University, 50 Nanyang Drive, Singapore 637553, Singapore
e-mail: danielthalmann@ntu.edu.sg

Z. Zhang
Microsoft Research, Redmond, WA 98052, USA
e-mail: zhang@microsoft.com

1 Introduction

Hand pose estimation is an important research topic which has a variety of applications in human-computer interaction (HCI) scenarios, such as gesture recognition, animation synthesis and virtual object manipulation. However, capturing the hand motion and full articulation is quite a challenging task due to its high flexibility. In previous researches, many sensor-based and vision-based methods have been proposed to fulfill the task. In sensor-based systems, specialized hardware is used for hand motion capture. The electromechanical or magnetic sensing devices, such as data-gloves [1] and optical sensors [2] are commonly used to measure the hand locations and finger articulation in such systems. Although they provide quite accurate measurements and can achieve real-time performance, such systems are cumbersome and expensive to use.

Vision-based methods are cheap alternatives to the sensor-based ones, and provide more naturalness in human computer interaction. However, they have their own challenges. First, the dimensionality of the hand motion parameter space is about 30, thus searching for a matched pose for an input image is computationally intensive due to the high-dimensional search space. Second, the hand is highly articulated, which results in self-occlusion in its projected images. The estimated pose can be ambiguous as we have no clue of the occluded hand parts. Third, the environment for the

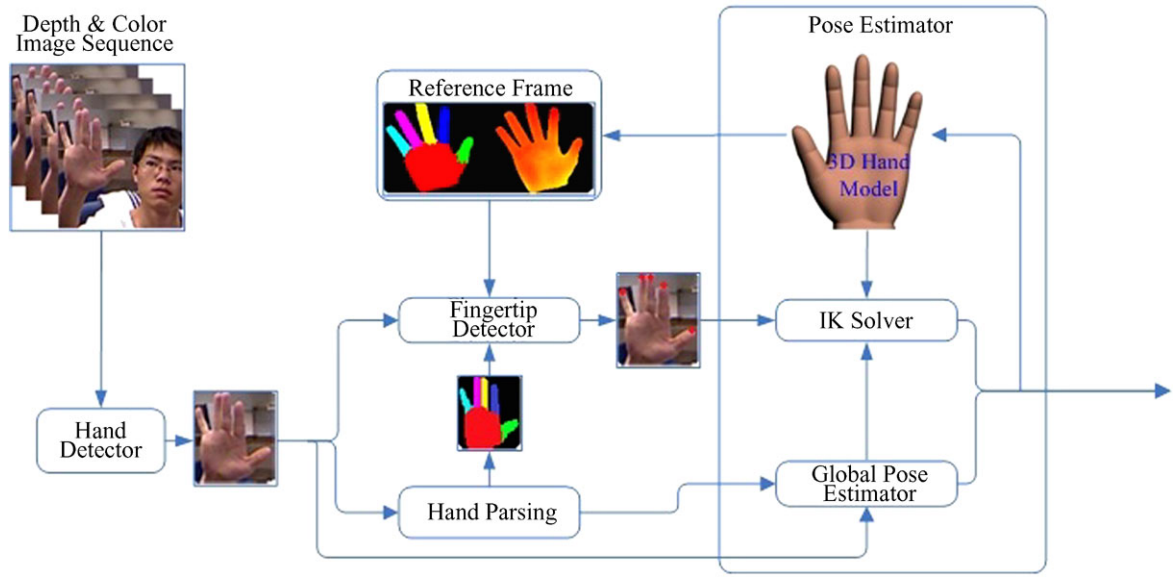


Fig. 1 Overview of the proposed hand pose estimation framework

HCI applications is usually uncontrolled, e.g., illumination variation, cluttered background, etc. This incurs additional difficulty in hand detection and feature extraction, such as the calculation of edge map.

Generally, vision-based hand pose estimation methods can be divided into two categories: appearance-based methods [3–12] and model-based methods [13–15]. In appearance-based approaches, various features are extracted from the input images to estimate the hand pose. Usually a lot of training samples are used to train a mapping function from the features to the hand poses in advance. Given the learned mapping function, the hand pose can be estimated efficiently. However, due to the nonlinearity of the mapping, these methods suffer from the problem of ambiguous poses without applying temporal continuity. In model-based approaches the hand pose is estimated by aligning a projected 3D hand model to the extracted hand features in the inputs. The pose parameters are found by minimizing the discrepancy between the model and hand features. As the dimension of the parameter space is high, such methods are slow and can easily be trapped into local minima without good initial pose estimation.

In this paper we propose a novel pixel-level hand parsing algorithm and 3D fingertip localization algorithm for full degree-of-freedom (DOF) hand pose estimation, which are inspired by the concepts of geodesic extrema [16] and inverse kinematics (IK)-based hand pose estimation approaches [2]. Based on the two algorithms, we implement a vision-based markerless hand pose estimation system with the input of depth image sequences. Figure 1 shows the main parts of the proposed framework, including 3D hand model, hand detector, hand parsing module, fingertip detector, global motion estimator and IK solver. The single-frame processing routine starts with hand detection in the depth image, followed by pixel-level hand parsing and 3D fingertip localization, and ends in global and local hand motion estimation. The 3D hand model is used for generating the hand part label image and depth image given a hand pose hypothesis as well as for IK-based pose estimation. The hand parsing algorithm incorporates a novel spatial-temporal feature into a Bayesian inference framework to assign the correct label to each image pixel. This feature accounts for both the 2D coordinate and relative positions to the finger edges of each pixel. The 3D fingertip localization scheme adapts the geodesic extrema extraction algorithm [16] to fingertip detection with the hand parsing results, a novel path-reweighting method and K-means clustering in metric space. The global motion estimator utilizes the identified palm part, the middle finger part and the corresponding depth image part to estimate global translation and rotation. The hand model is then aligned to input depth image according to the global motion estimation. By modeling each finger as a single kinematics chain, the IK solver estimates local hand pose with the detected fingertips. Finally, the 27 DOF hand pose estimation is used to drive the 3D hand model to generate the reference hand part label image and depth image.

Compared to previous researches, the proposed framework can provide several advantages. First, the hand parsing algorithm builds good correspondence between input depth images and the five fingers as well as the palm of a 3D hand model. This algorithm can be integrated into existing model-based hand tracking systems, such as [8], to achieve better performance by evaluating a hand hypothesis with the correspondence information, although it is used to improve the fingertip detection results in our framework. Second, the 3D

fingertip localization algorithm can be directly used for HCI interaction, such as in the proposed virtual instrument playing scenario. Third, the large number of DOFs of hand is decomposed in our framework. That is, we estimate global motion and local pose of the hand in separate phases. By using the five detected fingertips for local hand pose estimation, we further divide the parameter space of local hand pose into five non-overlapping subsets. This leads to more efficient pose estimation compared to searching for the candidate pose in the full parameter space of hand pose. Experimental results show the proposed hand tracking scheme provides reasonable accuracy at the real-time speed.

2 Related work

In this section we focus on previous researches in full DOF hand pose estimation, in which all the kinematic parameters of the hand, including global hand motion and local finger articulations, are recovered from the inputs. While the hand is chromatically homogeneous in appearance, some previous work adopts colored markers for pose estimation. In [2], colored markers are placed on the hand, and the 3D hand pose is retrieved with the 2D projection of these markers. The hand motion constraints are analyzed to reduce the 27 hand parameter space to 12 to enhance the time performance. In [17], a color glove with specially designed pattern is adopted in the hand tracking system, and hand pose estimation is performed by nearest-neighbor search in a large database. The method can capture the hand articulation quite accurately and self-occlusion can be handled. However, the method cannot estimate the exact depth position of the palm.

Markerless methods are still the mainstream for vision-based hand pose estimation and there is large room for improvement. These methods can be categorized into model-based method [3–12, 30] and appearance-based methods [13–15]. In [3], the hand motion is decoupled into global motion and local finger motion, which is recovered in a series of iterations. In each iteration, the global motion is formulated as a pose determination problem and solved by least median of square. The local finger motion is estimated by inverse kinematics using the fingertips as end-effectors. The method is not robust as extraction of fingertips is difficult and sensitive to self-occlusion. A similar framework is adopted in [4], in which the contour and silhouette differences are minimized between the input and model using two-step optimization. The frame rates are 11 Hz and 5 Hz on two sequences of seven and 18 DOF hand motions, respectively. In [30], the positions of the fingertips are tracked with the particle filter, which are used for pose estimation by combining an Inverse Kinematic solver. This fingertip tracking scheme is further utilized in [31], and is combined with the articulated iterative closest point algorithm to estimate the hand pose.

In [10], an Unscented Kalman filter is used to track the hand pose in a model-based framework. The likelihood of a hypothesized pose is evaluated in terms of the similarity between the image and model contours. A frame rate of 3 Hz is reported on a seven DOF hand motion sequence. The idea is extend in [11], in which a hierarchical filtering scheme is proposed for hand tracking to combine a tree-based object detector and a Bayesian Filter to eliminate the ambiguity in single-frame pose estimation.

In [9], the iterative closest point (ICP) algorithm is generalized for registration of articulated structures. This method is applied to track the motion of a hand with nine degree-of-freedom. The experiments show a frame rate of 3 Hz can be achieved and small occlusion can be tolerated. In [12] the feasible hand configuration is constructed by indexing the training samples using a KD-tree, and combines the Nelder–Mead simplex search and particle filtering to search for the hypothesized pose that best match the input, in terms of edge and silhouette similarities. However, no quantitative results are reported. Oikonomidis et al. [8] presents another model-based framework, and minimization of the difference between the input and model projection is solved using a variant of particle swarm optimization (PSO) algorithm. With GPU acceleration, the method can achieve a frame rate of about 15 Hz.

In [5], a multi-view method is proposed for 3D hand pose estimation. The hand motions are captured in both frontal and side view to overcome the self-occlusion problem. To handle the high dimensional parameter space of hand pose, it adopts a separable state-based particle filter (SSBPF) to reduce the computational complexity. The results show the average hand joint angle estimation error is about 11 degrees.

In [6], a hand motion capture system with the inputs of eight HD cameras is presented. It is capable of capturing the full DOF hand motion, and can handle the self-occlusion and interaction between two hands. Some salient points on the fingers are detected via pre-trained classifiers, and they are used in combination with edges and optical flow to build correspondence between the input images and the hand model. Hand pose is restored by minimizing the distance between the model projection and the corresponding points in all eight input images.

In [7], the randomized decision trees are trained on a dataset of synthetic hand depth images with labeled parts. During hand pose estimation, per-pixel classification is first performed to assign each pixel in the extracted hand region to a hand part, and a mean-shift is performed on the labeled results to find the joint locations for the hand skeleton. Hand pose is finally estimated by fitting the skeleton to the joint locations.

In [13], a non-parametric method is proposed to track the hand grasping motion. The extracted hand region is expressed with the Histogram of Oriented Gradients (HOG).

A dataset of the hand grasping motion in different viewpoints and illumination conditions is built. The hand pose is estimated by first retrieving several candidates with the input HOG feature using Local-sensitive Hashing, and then applying temporal continuity to the retrieved results. This method cannot handle quick hand motion due to its simple motion model.

In [14], an initialization scheme is proposed to use the hand silhouette for hand tracking. The whole parameter set of the hand pose is decomposed into many overlapping subsets. LSH-based nearest neighbor search is used to get the partial estimation for each subset, and the results are further integrated by a simulated annealing EM algorithm to give the complete pose estimation results.

In [15], 20 hand shape prototypes are rendered from 4128 views to generate a depth image dataset. Hand pose is recovered by retrieving the best match from the database that minimizes the chamfer distance and pixel-wise Euclidean distance in the depth images. However, the retrieval accuracy is still low and real-time performance is not achieved.

3 Hand pose estimation framework

The proposed hand pose estimation framework belongs to the model-based category. It aims at recovering all 27 degrees of freedom (DOF) of hand motion from a depth camera, without using any markers. Let the parameter space of hand motion be described by a feature vector $\phi = (\phi_g, \phi_l)$, where ϕ_g is the global hand motion and ϕ_l is the local hand motion. ϕ_g consists of 3D translation and rotation of the hand and ϕ_l corresponds to the 21 DOF of local hand pose. We adopt a right-handed coordinate system with origin at the center of camera projection to describe the hand motion. The positive X and Y axes point right and up parallel to the image plane of the camera, and the positive Z axis points out of the image plane along the optical axis. Global translation of the hand is then defined as the palm center position $T_g = (x_g, y_g, z_g)$ in the above coordinate system, and global rotation is defined as the Euler angles of palm rotation $\theta_g = (\theta_x, \theta_y, \theta_z)$ with the ZYX convention.

The task for the proposed scheme is to restore all 27 parameters of hand motion from the input depth image sequence, which is very challenging. We decompose this task into several sub-tasks. Let F_V be the 3D point cloud of the input hand region, and $F_V^m(\phi)$ be the point cloud generated by a 3D hand model. We further define U_H as the pixel set within the hand region, $\Pr(l|\psi(p))$ as the probability that pixel p has the label l, v_f^i as the detected fingertips, and v_i as the model fingertips. Their definitions will be given in detail in the following sections. The sub-tasks are then described as:

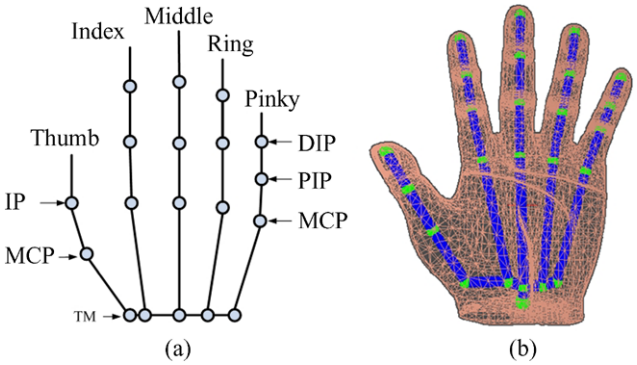


Fig. 2 The kinematic chain (*left*) and the 3D hand model (*right*)

1. Hand parsing: $\forall p \in U_H$, assign a label l_{opt} to p such that $\Pr(l_{\text{opt}}|\psi(p))$ is maximized;
2. Global motion estimation: find ϕ_g such that the error between F_V and $F_V^m(\phi)$ is minimized in the palm region. Here $\phi = (\phi_g, 0)$ means that no local motion is considered;
3. Local motion estimation: find ϕ_l so that the distance $\|v_i(\phi_l, \phi_g) - v_f^i\|_2$ is minimized for each finger;

Task 1 is fulfilled with the hand parsing algorithm. Tasks 2 and 3 are fulfilled with the hand detector and global motion estimator, and the 3D fingertip localization algorithm and IK solver, respectively.

4 3D hand model

The framework adopts a fully deformable hand model, which consists of the skeleton and the skin surface mesh, as shown in Fig. 2. The skeleton has 27 degrees of freedom (DOF), including 6 DOFs of global motion and 21 DOFs of local motion [19]. It is modeled as a kinematic chain of 20 joints. The joints are connected by bones in a tree structure, with root at the wrist. A set of static and dynamic motion constraints [19, 20] are adopted to limit the parameter space of hand pose and the skeleton has an equivalent of 15 DOFs of local motion. A label $l \in A_L = \{palm, thumb, index, middle, ring, pinky\}$ is pre-assigned to each vertex in the skin surface mesh. The elements in A_L correspond to the hand palm and five fingers. Given a pose vector ϕ and projection matrix P, the hand model can generate a depth image F_D^m, 3D point cloud F_V^m and hand part label image F_L^m. In our framework, F_D^m, F_V^m and F_L^m are used as reference frames for hand parsing and fingertip localization.

5 Hand detection

Robust hand detection itself is a difficult problem [21–24]. Our hand detector works on the input depth frames and sim-

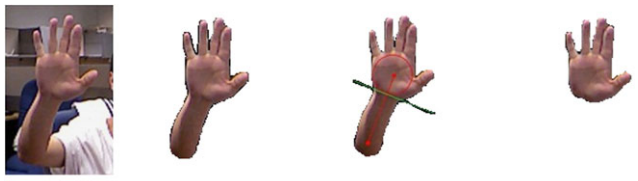


Fig. 3 Illustration of hand detection. (**a**) Is the original image, (**b**) is the foreground region, (**c**) shows the palm circle, orientation vector (*red line*) of the forearm and the segmentation line (*green line*), and (**d**) shows the final segmentation results

plifies this problem with several assumptions. First, we assume the hand is the nearest object to the camera and constrain global hand rotation by

$$-15^\circ \le \theta_x \le 15^\circ, \qquad -15^\circ \le \theta_y \le 15^\circ, \qquad -90^\circ \le \theta_z \le 90^\circ \tag{1}$$

Second, the maximum of absolute difference of depth values between points within forearm region and hand region is confined within certain threshold c, i.e. $c = 0.2$ m. Third, based on the morphology of the hand, we assume that the hand palm forms a globally largest blob in the hand and forearm region in the depth image when $\theta_x \approx \theta_y \approx 0^\circ$, and forms a locally largest blob when the hand rotates within ranges defined in (1). The palm region can thus be approximated with a circle $C_p = (p_p, r_p)$, where p_p is the palm center and r_p is the radius.

In the proposed system we use only the depth frames as the input, and each depth frame is preprocessed to generate a 3D point cloud. Based on the above assumptions, hand detection consists of three steps: foreground segmentation, palm localization and hand segmentation. It starts with threshold to the depth frame to get the foreground F. F is given by

$$F = \{(i, j) | z_{i,j} < z_0 + c\}, \tag{2}$$

where $(i, j, z_{i,j})$ denotes a pixel in the depth image at coordinate (i, j) and with depth value $z_{i,j}$, z_0 is the minimum depth value. This ensures that both hand and forearm regions are extracted from the depth frame. The contour of F is approximated by a polygon B. C_p then equals the largest inscribed circle of B. To reduce the computational complexity of palm localization, the center of C_p is tracked with a Kalman filter so that it is searched locally according to a prior prediction in each intermediate frame. Finally the hand and forearm regions are separated by a tangent line of C_p. This line is approximately perpendicular to the orientation vector of the forearm, which is defined as the Eigenvector that corresponds to the largest Eigenvalue of the covariance matrix of the contour pixel coordinates of F. Let U_H be the pixel set within the hand region. Figure 3 presents the hand detection results. Let the extracted hand regions in depth frame and 3D point cloud be F_D and F_V. Global hand translation T_g is obtained by finding the point corresponding to C_p in F_V.

6 Hand parsing

Hand parsing refers to the procedure during which each pixel within U_H is assigned a part label $l_i \in A_L$. At an intermediate frame k, we perform this task using a spatial-temporal-based scheme, with the input of reference hand pose ϕ^{k-1} in the previous frame, the extracted 3D point cloud F_V and a set of feature lines A_F. The palm is identified first, and the task to label remaining pixels is formulated as a classification problem, which is solved with a naive Bayesian classifier. Especially, we design a novel feature descriptor for each pixel to better solve the problem to exploit the temporal reference and spatial information.

We first use the Iterative Closest Point (ICP) algorithm [18] to coarsely align the 3D hand model to F_V based on the reference pose estimation ϕ^{k-1}, and then generate the reference depth image F_D^{ref}, 3D point cloud F_V^{ref} and label image F_L^{ref}. Besides, the palm is first identified according to its several properties. First, the palm can be assumed rigid and the 3D coordinates of palm pixels in F_V are continuous. Second, the geodesic distances from the palm center to palm border pixels show little variation. To label the palm part, we calculate the geodesic distance from palm center to each pixel to be labeled using the original method in [16]. A pixel is assumed to belong to the palm if the following three criteria are met:

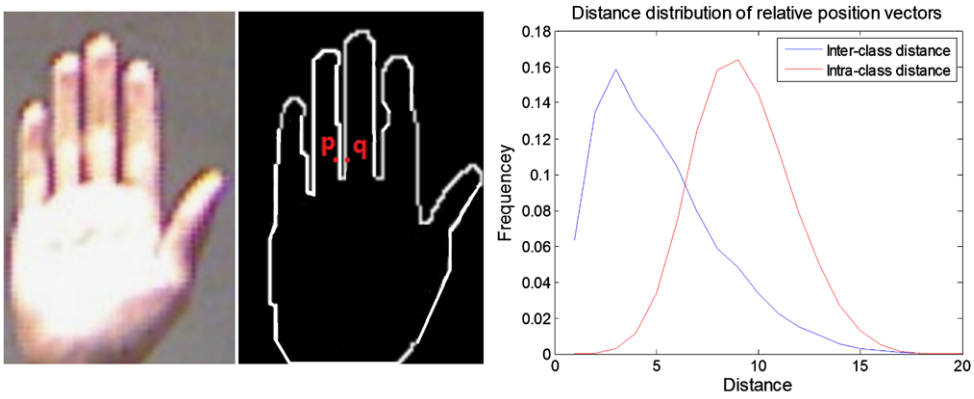
1. Corresponding pixel in F_L^{ref} has the palm label;
2. Its absolute depth difference from the corresponding pixel in F_D^{ref} is within certain threshold;
3. Its geodesic distance is within a fixed threshold;

We assign a feature vector $\psi(p)$ to each remaining pixel p to perform classification. $\psi(p)$ consists of two parts: the 2D coordinate p of the pixel and its relative position vector $D_F(p)$ to the feature lines A_F. $A_F = \{fl_1, fl_2, \ldots, fl_M\}$ is a set of M line segments extracted from the current frame that mainly correspond to edges of the projection of the five fingers on the image plane. $D_F(p)$ is defined as:

$$D_F(p) = \left(d_p^1, d_p^2, \ldots, d_p^M\right)^T, \tag{3}$$

where $d_p^m = sign(dist(p, fl_m))$ is the sign of the signed distance from pixel p to the line segment fl_m. The distance metric for the relative position vectors of two pixel p and q is given by

$$Dist_F(p, q) = \sum_{m=1}^{M} \left\| d_p^m - d_q^m \right\|_2 \tag{4}$$



(a) (b)

Fig. 4 The illustration for the feature $D_F(p)$. (**a**) Example where $D_F(p)$ is needed to differentiate two close-by pixels. (**b**) Inter-class and intra-class distance distribution of the distance metric $Dist_F$

Given that the finger edges are ideally detected and extracted into A_F, $D_F(p)$ can serve as a useful local descriptor to differentiate pixels that belong to different fingers. Figure 4(a) gives an example. The two pixels p and q have very close coordinates and thus can be easily classified to the same finger if only the coordinate information is used. In this case, their relative positions to the edge lines between them help to give the right classification. Figure 4(b) presents the quantitative evaluation of the effectiveness of $D_F(p)$. We estimate the inter-class and intra-class distances $Dist_F(p,q)$ on six synthetic sequences, where the ground truth information of pixel labels and finger edges of which are available. Details of the sequences will be given in Sect. 8. The inter-class samples contain 95 600 pairs of pixels and the intra-class samples contain 143 400 pairs of pixels. The results verify our previous assumption.

We now introduce the Bayesian framework to label each pixel p. The task is to find the label l_{opt} for p such that

$$\begin{aligned} l_{\text{opt}} &= \arg\max_i \Pr\big(l_i|\psi(p)\big) \\ &\propto \arg\max_i f\big(\psi(p)|l_i\big)\Pr(l_i) \\ &= \arg\max_i f(p|l_i) f\big(D_F(p)|l_i\big)\Pr(l_i) \end{aligned} \tag{5}$$

where $\Pr(l_i)$ is the prior for l_i; $f(p|l_i)$ is the position likelihood and $f(D_F(p)|l_i)$ is the likelihood based on the relative distance vector. We take advantage of the reference label frame F_L^{ref} to approximately estimate the three items for the current frame. $\Pr(l_i)$ can be obtained by calculating the ratio of the number of pixels within each finger to the number of pixels within all five fingers. Besides, we model $f(p|l_i)$ as a 2D Gaussian distribution, namely $f(p|l_i) \sim N(\mu_i, \sum_i)$. The parameters μ_i and $\sum_i$ are estimated with the pixel coordinates within each finger in F_L^{ref}.

$f(\psi(p)|l_i)$ is modeled as a metric exponential distribution [25] based on the relative position feature distance between p and the exemplar p_i of l_i. Note here the exemplar p_i cannot be directly obtained from F_L^{ref} as the extracted set of feature lines has changed in the current frame. Instead, we calculate the center of gravity of pixel coordinates within l_i in F_L^{ref}, from which we then spirally search for a pixel in the current frame so that the absolute depth difference between it and the corresponding pixel in the reference frame is within certain range. The relative position vector of this pixel is chosen as the exemplar p_i of l_i. $f(\psi(p)|l_i)$ is then given as:

$$f\big(D_F(p)|l_i\big) \propto \exp\big(-\lambda \times Dist_F(p, p_i)\big) \tag{6}$$

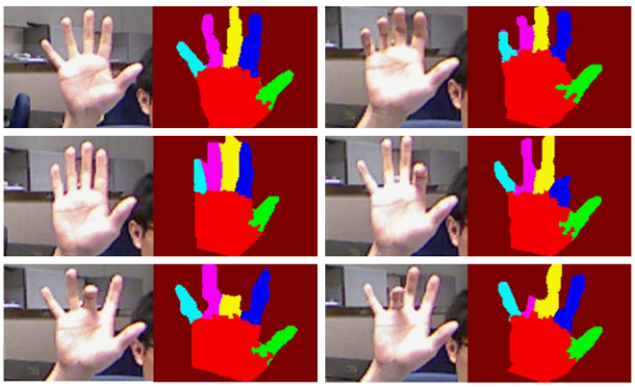


Fig. 5 Hand parsing results

Table 1 Comparison of labeling accuracy between using and not using relative position feature

Mode	Seq. 1	Seq. 2	Seq. 3	Seq. 4	Seq. 5	Seq. 6
Use $D_F(p)$	71.2	81.5	77.6	79.5	86.9	67.6
No $D_F(p)$	70.2	78.6	77.1	77.9	86.3	67.5

In Fig. 5 we present some examples of the hand parsing results. Besides, we also compare the labeling results of using $f(p|l_i)$ alone and using both $f(p|l_i)$ and $f(D_F(p)|l_i)$ on the six synthetic sequences. The percentage of correctly labeled pixels is given in Table 1. It shows that the relative

position feature improves the labeling accuracy, though not very significant. This may be due to several reasons. First, the position likelihood has worked well enough. Note that a large portion of mislabeled pixels result from the labeled palm regions as the finger pixels close to the palm are not very important for the following fingertip localization procedure. Second, we only rely on the depth frame and the feature lines are extracted from the hand contour. This means when two fingers are side-by-side, $D_F(p)$ has no effect in the current implementation.

7 Fingertip localization and labeling

The 3D Fingertip locations can be considered as end effectors for inverse-kinematics-based hand pose estimation [2, 26]. The proposed 3D fingertip detector is largely inspired by the geodesic extrema extraction algorithm [16, 27, 28]. Ideally, the points where the geodesic distances to the palm center are maximized should correspond to the fingertips. However, the original method in [16] cannot be directly applied to multiple fingertip localization for the following reasons. First, the palm center is not always precisely detected and its position inevitably fluctuates during real hand tracking process. The detected geodesic extrema may correspond to false fingertips. Second, multiple fingers can usually be side-by-side, in which case the method in [16] is likely to detect only one of these near-by fingertips since it requires the geodesic extrema to be sparsely located. Therefore, we adapts the geodesic extrema extraction algorithm [16] to multiple fingertip localization by integrating the hand parsing results, a novel path-reweighting scheme and K-means clustering in metric space.

The tasks of fingertip localization are to find the 3D locations of the fingertips and assign a label $l_i \in A_L$ to each of them. As in [16], we first build a graph $G_h = (V_h, E_h)$ using the point cloud F_V. V_h consists of all points within F_V. For each pair of vertices $(p, q) \in V_h$, there is an edge between p and q if and only if they are in the 8-neighborhood of each other and their 3D distance $d(p, q) = \|p - q\|_2$ is within threshold τ. Each edge is assigned a weight $\beta \times d(p, q)$, where $\beta = 1.0$ if p and q are within the same hand part, and $\beta = 10.0$ otherwise. While the resulting graph may not be connected, we search for a set of connected components in G_h using the union-find algorithm. The connected component that contains the palm center is identified and all other connected components are connected to it by finding their nearest vertices and adding an edge with weights equal to the 3D Euclidean distance. In this way all vertices in G_h are connected.

The fingertips are located one-by-one in the proposed scheme. In the first pass, we perform a Dijkstra graph search on G_h to calculate the geodesic distance from palm center p_p for each vertex $p \in V_h$. The one with the largest geodesic distance is taken as a fingertip,denoted as p_g^1. Let the geodesic distance of each vertex p be $d_g(p)$. The shortest path from p_p to p_g^1 is then extracted. Analysis of the topology of the hand shows that this shortest path actually approximates the skeleton of the corresponding finger. Let the point set on the path be $A_P^1 = \{ps_{1,k} | k = 0, 1, \ldots, M_1\}$, where M_1 is the number of the points. We then shrink the weight of the path by:

1. Add an edge of weight $\alpha_1 \times d_g(p_g^1)$ from p_p to p_g^1;
2. Shrink the weights of the edges connecting to each point by a weighting function $f_s(d_g(ps_{1,k}))$;

where $0 \le \alpha_1 \le 1$, and $f_s(d)$ is a non-increasing function defined on $[0, d_g(p_g^1)]$ and satisfies the property $f_s(0) = 1$, $f_s(d_g(p_g^1)) = \alpha_2$, $0 \le \alpha_2 \le 1$. By properly adjusting the parameters α_1 and α_2, the reweighting procedure can decrease the geodesic distances of points along the path A_P^1, and thus reduce the possibility that multiple fingertips are detected in a single finger region. Especially, the definition of $f_s(d)$ indicates that edges close to the palm center p_p are less influenced so that the geodesic distance from p_p to other fingertips will not change significantly. In our implementation, $f_s(d)$ takes the form of a biquadratic function. In each following pass, the geodesic distances to p_p are again calculated for all vertices by Dijkstra search on the modified graph and a new fingertip is located by finding the geodesic extrema. The path-reweighting scheme is then applied. The procedure iterates for $K \ge 5$ times to generate K fingertip candidates p_g^i, $i = 1, 2, \ldots, K$.

We utilize the K-means clustering algorithm to partition all K fingertip candidates into five clusters. To this end, each candidate is assigned a feature vector $\varphi(p_g^i) = \{A_P^i, \Pr(l|p_g^i)\}$. A_P^i is the shortest path points and $\Pr(l|p_g^i)$ is the hand part label distribution of the points on A_P^i. The distance metric for two candidate fingertips p_g^i and p_g^j is given by

$$Dist(p_g^i, p_g^j) = D_{KL}\big(\Pr(l|p_g^i) || \Pr(l|p_g^j)\big) + D_H(A_P^i, A_P^j), \quad (7)$$

where D_{KL} is the K–L divergence between the two distributions and D_H is the Hausdorff distance. In each cluster, the candidate with the largest geodesic distance is selected as the true fingertip. This gives five located fingertips: p_f^i, $i = 1, 2, \ldots, 5$. We denote their corresponding 3D positions as v_f^i, $i = 1, 2, \ldots, 5$. Besides, the label of each finger is determined by finding the label that maximizes $\Pr(l|p_g^i)$.

8 Hand pose estimation

Global and local hand motions are estimated in separate phases in the proposed framework. The global translation

T_g is estimated during hand detection. In this section, the global motion estimator recovers 3D global hand motion θ_g based on the identified palm part and middle finger part. With the estimation of ϕ_g, the hand model is aligned to F_V. Each finger is modeled as a single kinematic chain, and the 21 DOF parameter space of ϕ_l is thus decomposed into five non-overlapping subsets. The IK solver estimates the local pose for each finger with the detected fingertips. Especially, the static and dynamic hand motion constraints [19, 20] are integrated into local motion estimation to reduce the inherent ambiguity of IK solver as well as to improve the speed.

8.1 Global rotation estimation

Since global rotation is defined as the Euler angles of palm rotation, we find that it can be uniquely defined by two vectors $\{\mathbf{v}_n, \mathbf{v}_t\}$, where $\mathbf{v}_n$ is the normal vector of the hand palm and $\mathbf{v}_t$ is the vector from the palm center to Metacarpophalangeal joint of the middle finger. With this notation, we define $\theta_g = (0, 0, 0)$ as $\mathbf{v}_n^0 = (0, 0, 1)$ and $\mathbf{v}_t^0 = (0, 1, 0)$ in the coordinate system given in Sect. 2. Global hand rotation can then be obtained by estimating $\{\mathbf{v}_n, \mathbf{v}_t\}$ from the point cloud F_V.

We utilize the hand parsing results to estimate $\mathbf{v}_n$ and $\mathbf{v}_t$. To estimate $\mathbf{v}_n$, we extract the 3D points in F_V that are identified as palm points, and perform PCA analysis to their coordinates. Based on the 3D shape of hand palm, $\mathbf{v}_n$ is approximated by the Eigenvector that corresponds to the smallest Eigenvalue of the covariance matrix of these 3D point coordinates. Estimation of $\mathbf{v}_t$ requires localization of Metacarpophalangeal joint of the middle finger, which is difficult due to self-occlusion and lack of discriminative features. Note that the middle finger has little adduction/abduction motion and we assume that global hand rotation is limited with (1). Instead of directly estimating $\mathbf{v}_t$, we use the 2D vector from the palm center p_p to the center of gravity of the identified middle finger part p_{mid} to approximate the projection of $\mathbf{v}_t$ on the image plane. Let the 2D vector be $\mathbf{v}_t^- = p_{\text{mid}} - p_p = (x_t^-, y_t^-)$, and the current projection matrix be P. Let the rotation matrix corresponding to $\theta_g = (\theta_x, \theta_y, \theta_z)$ be R_g. We have

$$\begin{aligned}\mathbf{v}_t^- &= P \times \mathbf{v}_t \\ &= P \times R_g \times \mathbf{v}_t^0 \\ &= \begin{bmatrix} \tau(\sin\theta_x \sin\theta_y \cos\theta_z - \sin\theta_z \cos\theta_x) \\ \tau(\cos\theta_x \cos\theta_z + \sin\theta_x \sin\theta_z \sin\theta_y) \end{bmatrix}\end{aligned} \tag{8}$$

Here τ is a scalar constant determined by the projection matrix P. The equation has four unknown variables. However, with the previous assumption in (1), θ_z can be estimated by $\theta_z \approx \arctan(-x_t^-/y_t^-)$. We then solve the equation $R_g \times \mathbf{v}_n^0 = \mathbf{v}_n$ for (θ_x, θ_y) with the constraints in (1).

8.2 Inverse kinematics

Inspired by the instrumented hand motion capture systems [2], we use inverse kinematics to estimate ϕ_l based on the estimation of ϕ_g and detected 3D fingertip locations. A kinematic chain is built for each finger in the hand model. By using the dynamic constraints in [20] the degrees of freedom for each kinematic chain is reduced to three. To perform inverse kinematics, we first rotate the hand model by $\theta_g = (\theta_x, \theta_y, \theta_z)$ and translate the hand model by T_g so that the model palm center coincides with the detected palm center. For each kinematic chain we minimize the difference between the model fingertip locations and detected fingertip locations:

$$\theta_{l,i}^* = \underset{\theta_{l,i}}{\operatorname{argmin}} \left\| v_i(\theta_{l,i}, \phi_g) - v_f^i \right\|_2, \tag{9}$$

where $\theta_{l,i}$ is the joint angle vector for the ith kinematic chain; $v_i(\theta_{l,i}, \phi_g)$ is the ith model fingertip location as a function of $\theta_{l,i}$ and ϕ_g and v_f^i the detected fingertip locations. This minimization problem is solved with the cyclic coordinate descent algorithm [29] in our system.

9 Experiments

In this section we present the experimental results. The whole program was coded in C++, and tested on a PC with Intel i5 750 CPU and 4G RAM. The experimental evaluation of the proposed method includes both quantitative test on synthesized input and real-world test. The resolutions of the input sequences for both tests are 320×240. For quantitative tests we synthesized six sequences which contain the ground truth data of the hand motion parameters. The six sequences include different types of hand motions, with seq. 1 for grasping motion, seq. 2 for adduction/abduction motion, seq. 3 for successive single finger motion, seq. 4 for flexion motion of two fingers, seq. 5 for global rotation and seq. 6 for combination of grasping and global rotation motion. For real-world test, we present a HCI application based on the proposed hand pose estimation scheme with the input of a Kinect sensor. The average frame rate is about 5 Hz.

9.1 Pose estimation accuracy

This part presents the quantitative evaluation results for global rotation and local pose estimation. For local motion, we define the evaluation metric as the mean absolute error between the recovered local joint angles and ground truth data, and the results for all six sequences are presented. For global rotation, we define the evaluation metric as the error between ground truth data and recovered global rotation for all three dimensions, and the results are presented for

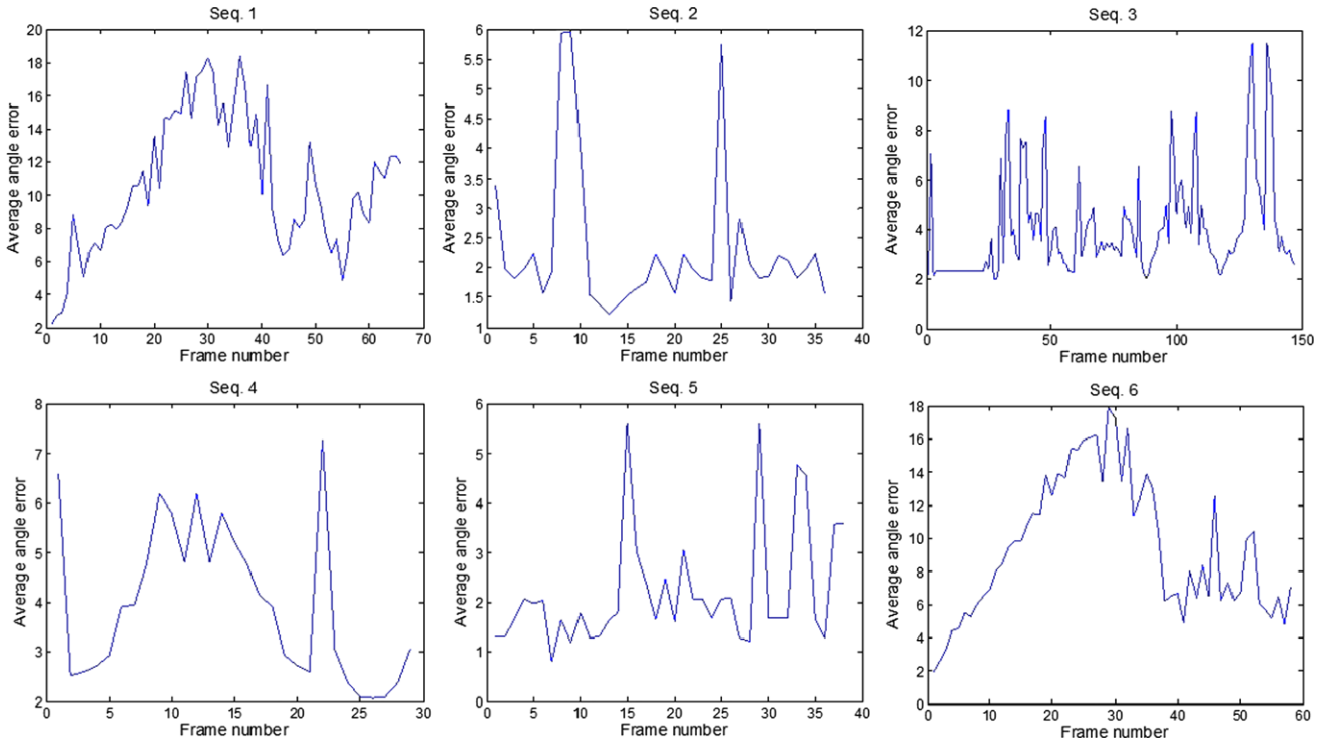


Fig. 6 Quantitative test results for local motion estimation on synthetic sequences

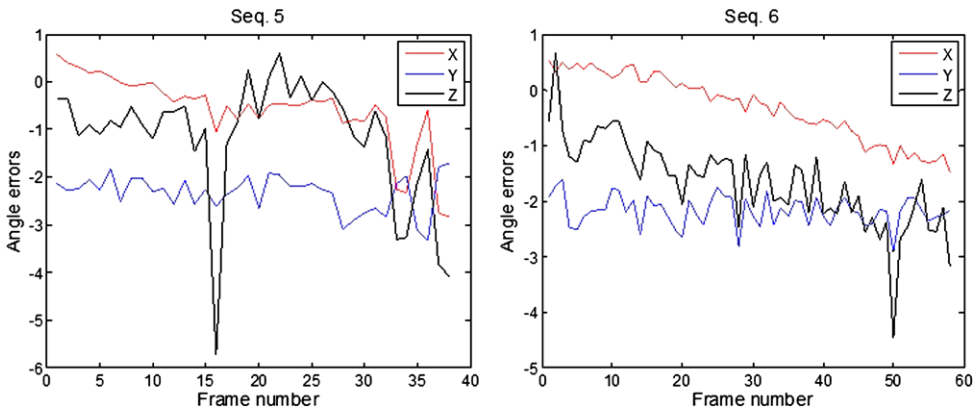


Fig. 7 Quantitative test results for global motion estimation on synthetic seq. 5 and seq. 6

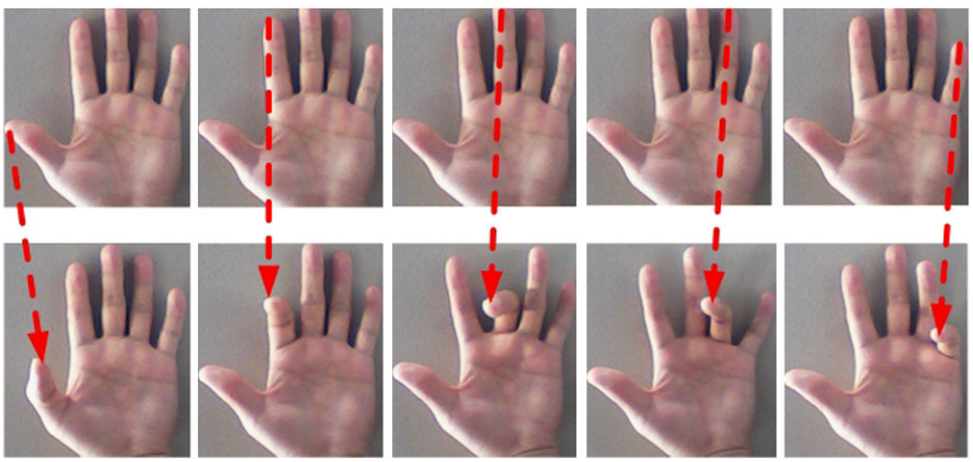


Fig. 8 Dynamic gesture set. Each dynamic gesture is defined by the 3D fingertip motion

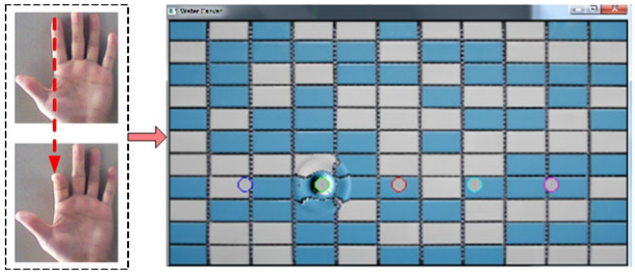


Fig. 9 Playing a simulated water-oscillator instrument

sequences that contain global motion. Figure 6 presents the quantitative test results for local pose estimation. The average local pose estimation errors are 10.58°, 2.28°, 4.04°, 3.93°, 2.21° and 9.42° for the six sequences, respectively. The errors for seq. 1 and seq. 6 are bigger than other sequences as grasping is quite complex hand motion. Figure 7 presents the quantitative test results for global rotation estimation. We can see that global rotation estimation is quite accurate, with estimation error within 3° for most times.

9.2 Virtual instrument playing

In this application we directly utilize the tracked 3D fingertips to play a simulated water-oscillator instrument. We define five dynamic gestures to interact with the instrument and each gesture corresponds to the motion of a single finger. To recognize the gesture, we constantly check whether the corresponding finger is performing a sudden flexion movement based on the fingertip position changes over certain time interval. The gesture set is shown in Fig. 8. When a dynamic gesture is recognized, the corresponding sound is played and a simulated water wave is generated around the oscillator. Figure 9 illustrates the user interface for this application.

10 Conclusions

In this paper we present a vision-based markerless hand pose estimation framework with depth image sequence input. It mainly relies on a novel hand parsing algorithm and 3D fingertip localization algorithm. Quantitative evaluations show the proposed framework can capture natural hand motion quite accurately, and a virtual instrument playing application is developed to demonstrate the use of the system. In the future we plan to combine edge features in color frames for more accurate hand parsing to improve hand pose estimation accuracy.

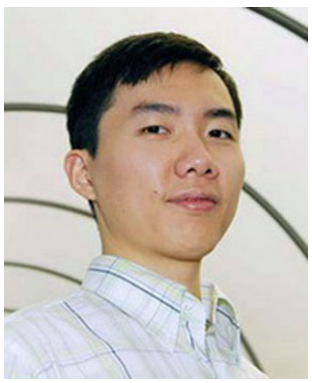
Acknowledgements This research, which is carried out at BeingThere Centre, is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

References

1. Cyberglove 2. http://www.cyberglovesystems.com
2. Aristidou, A., Lasenby, J.: Motion capture with constrained inverse kinematics for real-time hand tracking. In: International Symposium on Communications, Control and Signal Processing, pp. 1–5 (2010)
3. Wu, Y., Huang, T.S.: Capturing articulated human hand motion: a divide-and-conquer approach. In: Proceedings of the IEEE International Conference on Computer Vision 1, pp. 606–611 (1999)
4. Henia, O.B., Hariti, M., Bouakaz, S.: A two-step minimization algorithm for model-based hand tracking. In: WSCG (2010)
5. Ho, M., Tseng, C., Lien, C., Huang, C.: A multi-view vision-based hand motion capturing system. Pattern Recognit. **44**(2), 443–453 (2011)
6. Ballan, L., Taneja, A., Gall, J., Gool, L.V., Pollefeys, M.: Motion capture of hands in action using discriminative salient points. In: ECCV, vol. 12, pp. 640–653 (2012)
7. Keskin, C., Kira, F., Kara, Y.E., Akarun, L.: Real time hand pose estimation using depth sensors. In: Proceeding of the IEEE International Conference on Computer Vision Workshops, pp. 1228–1234 (2011)
8. Oikonomidis, I., Kyriazis, N., Argyros, A.A.: Efficient model-based 3D tracking of hand articulations using kinect. In: Proceedings of the British Machine Vision Conference (2011)
9. Pellegrini, S., Schindler, K., Nardi, D.: A generalisation of the ICP algorithm for articulated bodies. In: Proceedings of the British Machine Vision Conference (2008)
10. Stenger, B., Mendonqa, P.R.S., Cipolla, R.: Model-based 3D tracking of an articulated hand. In: CVPR, vol. 2, pp. 310–315 (2001)
11. Stenger, B., Thayananthan, A., Torr, P.H.S., Cipolla, R.: Model-based hand tracking using a hierarchical Bayesian filter. IEEE Trans. Pattern Anal. Mach. Intell. **28**(9), 1372–1384 (2006)
12. Lin, J.Y., Wu, Y., Huang, T.S.: 3D model-based hand tracking using stochastic direct search method. In: FG 2004, pp. 693–698 (2004)
13. Romero, J., Kjellstrom, H., Kragic, D.: Monocular real-time 3D articulated hand pose estimation. In: IEEE-RAS International Conference on Humanoid Robots, pp. 87–92 (2009)
14. Xu, J., Wu, Y., Katsaggelos, A.: Part-based initialization for hand tracking. In: IEEE International Conference on Image Processing, pp. 3257–3260 (2010)
15. Doliotis, P., Athitsos, V., Kosmopoulos, D., Perantonis, S.: Hand shape and 3D pose estimation using depth data from a single cluttered frame. In: ISVC, vol. 1, pp. 148–158 (2012)
16. Plagemann, C., Ganapathi, V., Koller, D., Thrun, S.: Real-time identification and localization of body parts from depth images. In: IEEE International Conference on Robotics and Automation, pp. 3108–3113 (2010)
17. Wang, R.Y., Popovic, J.: Real-time hand tracking with a color glove. ACM Trans. Graph. **28**(3) (2009). doi:10.1145/1531326.1531369
18. Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2), 239–256 (1992)
19. Erol, A., Bebis, G., Nicolescu, M., Boyle, X.R.D.: A review on vision-based full DOF hand motion estimation. In: CVPR 05, pp. 75–82 (2005)
20. Lin, L.J., Ying, W., Huang, T.S.: Modeling the constraints of human hand motion. In: Proceedings of the Workshop on Human Motion, pp. 121–126 (2000)
21. Mo, Z., Neumann, N.: Real-time hand pose recognition using low-resolution depth images. In: CVPR 06, pp. 1499–1505 (2006)
22. Panin, G., Klose, S., Knoll, A.: Real-time articulated hand detection and pose estimation. In: Proceedings of the International Symposium on Advances in Visual Computing, pp. 1131–1140 (2009)

23. Kolsch, M., Turk, M.: Robust hand detection. In: FG 2004, pp. 614–619 (2004)
24. Kolsch, M., Turk, M.: Hand tracking with flocks of features. In: CVPR (2005)
25. Toyama, K., Blake, A.: Probabilistic tracking with exemplars in a metric space. Int. J. Comput. Vis. **48**(1), 9–19 (2002)
26. Chua, C.S., Guan, H., Ho, Y.K.: Model-based 3d hand posture estimation from a single 2d image. Image Vis. Comput. **20**(3), 191–202 (2002)
27. Baak, A., Muller, M., Bharaj, G., Seidel, H.P., Theobal, C.: A data-driven approach for real-time full body pose reconstruction from a depth camera. In: Proceedings of the IEEE International Conference on Computer Vision (2011)
28. Schwarz, L., Mkhitaryan, A., Mateus, D., Navab, N.: Estimating human 3d pose from time-of-flight images based on geodesic distances and optical flow. In: FG 2011, pp. 700–706 (2011)
29. Wang, L.C.T., Chen, C.C.: A combined optimization method for solving the inverse kinematics problem of mechanical manipulators. IEEE Trans. Robot. Autom. **7**(4), 489–499 (1991)
30. Liang, H., Yuan, J., Thalmann, D.: 3D fingertip and palm tracking in depth image sequences. In: ACM MultiMedia, pp. 785–788 (2012)
31. Liang, H., Yuan, J., Thalmann, D.: Hand pose estimation by combining fingertip tracking and articulated ICP. In: VRCAI, vol. 12, pp. 87–90 (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 vision-based human computer interaction.



Junsong Yuan is a Nanyang Assistant Professor at Nanyang Technological University (NTU), Singapore. He is currently the Program Director of Video Analytics at Infocomm Center of Excellence, School of EEE, NTU. He received Ph.D. from Northwestern University and M.Eng. from National University of Singapore. Before that, he graduated from the Special Class for the Gifted Young of Huazhong University of Science and Technology and received his B.Eng. in 2002. Dr. Yuan's research interests include computer vision, video analytics, large-scale visual search and mining, human computer interaction, biomedical image analysis, etc. He is the co-chair of IEEE CVPR 2012 and 2013 Workshop on Human action understanding from 3D data (HAU3D'12'13), and the co-chair of CVPR 2012 Workshop on Large-scale video search and mining (LSVSM'12). He received the Outstanding EECS Ph.D. Thesis award from Northwestern University, and the Best Doctoral Spotlight Award from IEEE Conf. Computer Vision and Pattern Recognition Conference (CVPR'09). He has filed three US patents and two provisional US patents.

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. His current research interests include Real-time Virtual Humans in Virtual Reality, crowd simulation, and 3D Interaction. Daniel Thalmann has been the Founder of The Virtual Reality Lab (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. Until October 2010, he was the President of the Swiss Association of Research in Information Technology and one Director of the European Research Consortium in Informatics and Mathematics (ERCIM). 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 member of numerous Program Committees, Program Chair and CoChair of several conferences including IEEE VR, ACM VRST, and ACM VRCAI. 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) and 'Stepping Into Virtual Reality' (2007), published by Springer. He received his PhD in Computer Science in 1977 from the University of Geneva and an Honorary Doctorate (Honoris Causa) 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.

Zhengyou Zhang received the B.S. degree in electronic engineering from Zhejiang University, Hangzhou, China, in 1985, the M.S. degree in computer science from the University of Nancy, Nancy, France, in 1987, and the Ph.D. degree in computer science and the Doctorate of Science (Habilitation à diriger des recherches) from the University of Paris XI, Paris, France, in 1990 and 1994, respectively.

He is a Principal Researcher with Microsoft Research, Redmond, WA, USA, and manages the multimodal collaboration research team. Before joining Microsoft Research in March 1998, he was with INRIA (French National Institute for Research in Computer Science and Control), France, for 11 years and was a Senior Research Scientist from 1991. In 1996-1997, he spent a one-year sabbatical as an Invited Researcher with the Advanced Telecommunications Research Institute International (ATR), Kyoto, Japan. He has published over 200 papers in refereed international journals and conferences, and has coauthored the following books: *3-D Dynamic Scene Analysis: A Stereo Based Approach* (Springer-Verlag, 1992); *Epipolar Geometry in Stereo, Motion and Object Recognition* (Kluwer, 1996); *Computer Vision* (Chinese Academy of Sciences, 1998, 2003, in Chinese); *Face Detection and Adaptation* (Morgan and Claypool, 2010), and *Face Geometry and Appearance Modeling* (Cambridge University Press, 2011). He has given a number of keynotes in international conferences. Dr. Zhang is a Fel-

low of the Institute of Electrical and Electronic Engineers (IEEE), the Founding Editor-in-Chief of the IEEE Transactions on Autonomous Mental Development, an Associate Editor of the International Journal of Computer Vision, and an Associate Editor of Machine Vision and Applications. He served as Associate Editor of the IEEE Transactions on Pattern Analysis and Machine Intelligence from 2000 to 2004, an Associate Editor of the IEEE Transactions on Multimedia from 2004 to 2009, among others. He has been on the program committees for numerous international conferences in the areas of autonomous mental development, computer vision, signal processing, multimedia, and human-computer interaction. He served as a Program Co-Chair of the International Conference on Multimedia and Expo (ICME), July 2010, a Program Co-Chair of the ACM International Conference on Multimedia (ACM MM), October 2010, and a Program CoChair of the ACM International Conference on Multimodal Interfaces (ICMI), November 2010. He is serving a General Co-Chair of the IEEE International Workshop on Multimedia Signal Processing (MMSP), October 2011.