Robust Part-Based Hand Gesture Recognition Using Kinect Sensor

Zhou Ren, Junsong Yuan, *Member, IEEE*, Jingjing Meng, *Member, IEEE*, and Zhengyou Zhang, *Fellow, IEEE*

***Abstract*—The recently developed depth sensors, e.g., the Kinect sensor, have provided new opportunities for human-computer interaction (HCI). Although great progress has been made by leveraging the Kinect sensor, e.g., in human body tracking, face recognition and human action recognition, robust hand gesture recognition remains an open problem. Compared to the entire human body, the hand is a smaller object with more complex articulations and more easily affected by segmentation errors. It is thus a very challenging problem to recognize hand gestures. This paper focuses on building a robust part-based hand gesture recognition system using Kinect sensor. To handle the noisy hand shapes obtained from the Kinect sensor, we propose a novel distance metric, Finger-Earth Mover's Distance (FEMD), to measure the dissimilarity between hand shapes. As it only matches the finger parts while not the whole hand, it can better distinguish the hand gestures of slight differences. The extensive experiments demonstrate that our hand gesture recognition system is accurate (a 93.2% mean accuracy on a challenging 10-gesture dataset), efficient (average 0.0750 s per frame), robust to hand articulations, distortions and orientation or scale changes, and can work in uncontrolled environments (cluttered backgrounds and lighting conditions). The superiority of our system is further demonstrated in two real-life HCI applications.**

***Index Terms*—Finger-Earth Mover's Distance, hand gesture recognition, human-computer interaction, Kinect system.**

I. Introduction

HAND gesture recognition is of great importance for human-computer interaction (HCI), because of its extensive applications in virtual reality, sign language recognition, and computer games [4]. Despite lots of previous work, traditional vision-based hand gesture recognition methods [5]–[7] are still far from satisfactory for real-life applications. Because of the nature of optical sensing, the quality of the captured images is sensitive to lighting conditions and cluttered backgrounds, thus optical sensor based methods are usually unable to detect and track the hands robustly, which largely affects the performance of hand gesture recognition.

To enable a more robust hand gesture recognition, one effective way is to use other sensors to capture the hand gesture and motion, e.g., through the data glove [8]. Unlike optical sensors, such sensors are usually more reliable and are not affected by lighting conditions or cluttered backgrounds. However, as it requires the user to wear a data glove and sometimes requires calibration, it is inconvenient to use and may hinder the natural articulation of hand gesture. Also, such data gloves are usually more expensive than optical sensors, e.g., cameras. As a result, it is not a very popular way for hand gesture recognition.

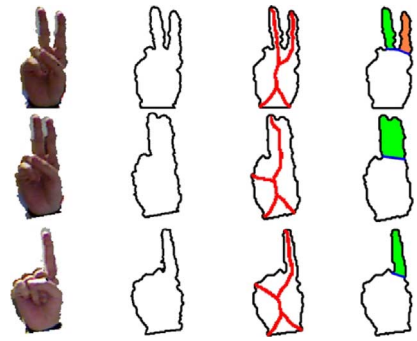


Fig. 1. The first two columns illustrate three challenging cases for hand gesture recognition using Kinect sensor, where the first two hands have the same gesture while the third one confuses the recognition. Using the skeleton representation shown in red in the third column [2], the last two hand gestures lead to very similar skeletons, thus skeleton-based matching algorithm [3] classifies them as the same gesture. In the last column, the part-based representations are illustrated. Using the proposed distance metric, *Finger-Earth Mover's Distance*, we can classify the first two hands as the same gesture and handle the noisy hand shapes obtained by Kinect sensor.

Thanks to the recent development of inexpensive depth cameras, e.g., the Kinect sensor [9], new opportunities for hand gesture recognition emerge. Instead of wearing a data glove, using the Kinect sensor can also detect and segment the hands robustly, thus it provides a valid base for gesture recognition. In spite of many recent successes in applying the Kinect sensor to articulated face recognition [10], human body tracking [11] and human action recognition [12], it is still an open problem to use Kinect for hand gesture recognition. Due to the low-resolution of the Kinect depth map, typically, of only 640×480, although it works well to track a large object, e.g., the human body, it is difficult to detect and segment a small object from an image with this resolution, e.g., a human hand which occupies a very small portion of the image with more complex articulations. In such a case, the segmentation of the hand is usually inaccurate, thus may significantly affect the recognition step.

To illustrate the above problem, the first column of Fig. 1 shows three examples. It can be seen that the contours (in the second column) have significant local distortions in addition to pose variations. Due to the low resolution and inaccuracy of the

Manuscript received July 15, 2012; revised October 17, 2012; accepted October 22, 2012. Date of publication February 25, 2013; date of current version July 15, 2013. A preliminary version of this paper appeared in the ACM International conference on Multimedia [1]. This work was supported in part by the Nanyang Assistant Professorship SUG M4080134 and Microsoft Research gift grant. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Eckehard G. Steinbach.

Z. Ren, J. Yuan, and J. Meng are with Nanyang Technological University, Singapore.

Z. Zhang is with Microsoft Research, Redmond, WA 98052 USA.

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.2013.2246148

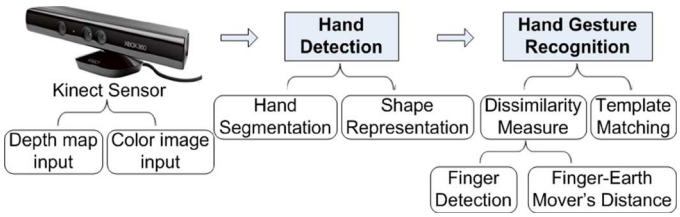


Fig. 2. The framework of our part-based hand gesture recognition system.

Kinect sensor, the two fingers of the second hand are indistinguishable as they are close to each other. Unfortunately, classic shape recognition methods, such as correspondence-based shape matching algorithms [13], [14] and skeleton matching methods [3], [15], cannot robustly recognize shape contour with severe distortions. For example, as shown in the third column of Fig. 1, the red skeletons of the last two hands are very similar. Hence skeleton matching algorithms classify them as the same gesture [3].

Clearly, recognizing such noisy hand contours is challenging, especially if there are many hand gestures to recognize. In order to address this problem, we propose a novel shape distance metric called Finger-Earth Mover's Distance (FEMD). FEMD is specifically designed for hand shapes. As it only matches the fingers while not the whole hand, it can better handle the noisy hand shapes obtained by Kinect sensor.

Fig. 2 shows the framework of our hand gesture recognition system. We use Kinect sensor as the input device which captures both the color image and its corresponding depth map. With the help of depth cue, we can detect the user's hand robustly to the cluttered backgrounds and lighting conditions. Then, we represent the hand shape by its finger parts, which is detected by shape decomposition. Finally, the dissimilarity between the obtained hand shape and each gesture template is measured by the proposed distance metric, FEMD, for gesture recognition.

To evaluate our method, we build a new challenging dataset (containing 1000 cases collected in uncontrolled environments). Tests on this dataset shows that our hand gesture recognition system not only operates accurately and efficiently (a mean accuracy of 93.2% in 0.0750 s per frame), but also is robust to uncontrolled environments and hand gesture variations in orientation, scale, articulation, and shape distortions. We compare our algorithm with shape contexts [13], and skeleton path similarity [3] in Section IV-B4 and show our superiority in hand gesture recognition. Furthermore, on top of our gesture recognition algorithm, we build two real-life HCI demos to illustrate the effectiveness of our method in Section V.

The main contributions of this paper are as follows:

- We propose a part-based hand gesture recognition system, based on a novel distance metric Finger Earth Mover Distance (FEMD). It is robust to orientation, scale, articulation changes as well as local distortions of hand shapes. To our best knowledge, this is the first part-based hand gesture recognition system using Kinect sensor.
- We demonstrate our hand gesture recognition algorithm in two HCI applications. The proposed system operates accurately and efficiently in uncontrolled environments. It is applicable to other HCI applications.

II. Related Work

Many vision-based hand gesture recognition approaches have been proposed in the literature [16]–[18], see [5]–[7] for more complete reviews. Vision-based hand gesture recognition methods can be classified into two categories. The first category is Statistics Learning based approaches: For a dynamic gesture, by treating it as the output of a stochastic process, the hand gesture recognition can be addressed based on statistical modeling, such as PCA, HMMs [16], [17], and more advanced particle filtering [19] and condensation algorithms [20]. The second category is Rule based approaches: Rule based approaches propose a set of pre-encoded rules between input features, which are applicable for both dynamic gestures and static gestures. When testing an hand gesture, a set of features are extracted and compared with the encoded rules, the gesture with the rule that best matches the test input is outputted as the recognized gesture [18].

Unfortunately, all existing hand gesture recognition methods have constraints on the user or the environment, which greatly hinders its widespread use in real-life applications. On one hand, to infer the pose of the palm and angles of the joints, many methods use colored markers to extract high-level features, such as the fingertip, joint locations or some anchor points on the palm [21]–[24]. On the other hand, some methods proposed to represent the hand region by edges or an ellipse [25]–[27] using skin color model. However, a common problem of the methods in these two categories is the inaccurate hand segmentation: none of these methods operates well in cluttered environments due to the sensitivity of colored markers and skin color model to the background. Besides, a few studies try to first fully reconstruct the 3D hand surfaces [8], [28]–[31]. Even though the 3D data provides valuable information that can handle problems like self-occlusion, an accurate, real time and robust 3D reconstruction is still very difficult. Furthermore, the high computational cost forbids its widespread adoption.

Fortunately, recent development of depth sensors (e.g., Kinect sensor) provides a robust solution to hand segmentation. However, due to the low resolution and inaccuracy of the depth map, the obtained hand contour can be quite noisy. Classic shape recognition methods are not robust to severe distortions in hand shapes. For instance, contour-based recognition approaches, such as moments, are not robust when the contour is polluted by local distortions. Skeleton-based recognition methods [15] also suffer from contour distortions, because even little noise or slight variations in the contour often severely perturb the topology of its skeletal representation. Bai *et al.* proposed a skeleton pruning method in [3], which makes skeleton robust to contour noise. However, skeleton-based methods still cannot deal with the ambiguity problem as shown in Fig. 1, as the second and the third shape have more similar skeletons than that of the first and the second shape. As for the correspondence-based shape recognition methods such as shape contexts [13] and inner-distance [14], they are not effective in solving the ambiguity in Fig. 1 either, because the correspondences of the second and the last hands have more similar contexts than the first and the second one do.

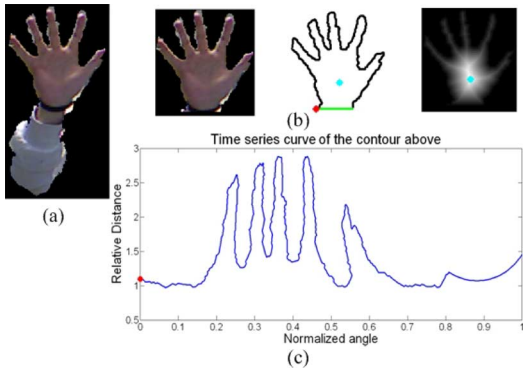


Fig. 3. Hand detection. (a) The rough hand segmented by depth thresholding; (b) A more accurate hand detected with black belt (the green line), the initial point (the red point) and the center point (the cyan point); (c) Its time-series curve representation.

III. Part-Based Hand Gesture Recognition

Now we introduce our part-based hand gesture recognition system. Fig. 2 illustrates the framework, which consists of two major modules: hand detection and hand gesture recognition.

A. Hand Detection

As shown in Fig. 2, we use Kinect sensor as the input device, which captures the color image and the depth map at 640×480 resolution. Generally the depth information derived from Kinect sensor is usable but not very accurate in details.

In order to segment the hand shape, firstly we locate the hand position using the Kinect windows SDK hand tracking function. Then, by thresholding from the hand position with a certain depth interval, a rough hand region can be obtained, as shown in Fig. 3(a). Second, we require the user to wear a black belt on the gesturing hand's wrist, in order to more accurately segment the hand shape. After detecting the black-color pixels, we use RANSAC to fit a line to locate the black belt, as shown in Fig. 3(b). The hand shape is generally of 100×100 pixel resolution, with possibly severe distortions.

After detecting the hand shape, we represent it as a *time-series curve*, as shown in Fig. 3(c). Such a shape representation has been successfully used for the classification and clustering of shapes [32]. The time-series curve records the relative distance between each contour vertex and a center point. We define the center point as the point with the maximal distance after Distance Transform on the shape (the cyan point), as shown in Fig. 3(b); and the initial point (the red point) is defined according to the RANSAC line detected from the black belt (the green line).

In our time-series representation, the horizontal axis denotes the angle between each contour vertex and the initial point relative to the center point, normalized by 360°. The vertical axis denotes the Euclidean distance between the contour vertices and the center point, normalized by the radius of the maximal inscribed circle. As shown in Fig. 3, the time-series curve captures nice topological properties of the hand, such as the finger parts.

B. Hand Gesture Recognition

The hand gesture recognition module in Fig. 2 is the major part of our part-based hand gesture recognition system. With the hand shape and its time-series curve, we now present how to robustly recognize the hand gesture.

1) Template Matching: We use template matching for recognition, i.e., the input hand is recognized as the class with which it has the minimum dissimilarity distance:

$$c = \arg\min_{c} \mathrm{FEMD}(H,\ T_c),$$

where H is the input hand; T_c is the template of class c; $\mathrm{FEMD}(H,\ T_c)$ denotes the proposed Finger-Earth Mover's Distance between the input hand and each template. Now we introduce the Finger-Earth Mover's Distance.

2) Finger-Earth Mover's Distance: In [33], Rubner *et al.* presented a general and flexible metric, called Earth Mover's Distance (EMD), to measure the distance between signatures or histograms. EMD is widely used in many problems such as content-based image retrieval and pattern recognition [34], [35].

EMD is a measure of the distance between two probability distributions. It is named after a physical analogy that is drawn from the process of moving piles of earth spread around one set of locations into another set of holes in the same space. The location of earth pile and hole denotes the mean of each cluster in the signatures, the size of each earth pile or hole is the weight of cluster, and the ground distance between a pile and a hole is the amount of work needed to move a unit of earth. To use this transportation problem as a distance measure, i.e., a measure of dissimilarity, one seeks the least costly transportation—the movement of earth that requires the least amount of work.

References [36] and [37] applied EMD to shape matching and contour retrieval, which represents the contour by a set of local descriptive features and computes the set of correspondences with minimum EMD costs between the local features. However, the existing EMD-based contour matching algorithms have two deficiencies when applied to hand gesture recognition:

- Two hand shapes differ mainly in global features while not local features. As shown in Fig. 4(a) and (b), the fingers (global features) are their major difference. Besides, the large number of local features slows down the speed of contour matching. Therefore, it is better to consider global features in contour matching.
- EMD allows for partial matching, i.e., a signature and its subset are considered to be the same in EMD measure: as in Fig. 4(c) and (d), the EMD distance of these two signatures is zero because the signature in Fig. 4(d) is a subset of Fig. 4(c). However, in many situations partial matching is illogical, such as in the case of Fig. 4(a) and (b), where the finger in Fig. 4(b) is a partial set of the fingers in Fig. 4(a). Clearly, they should be considered different.

Our Finger-Earth Mover's Distance (FEMD) can address these two deficiencies of the contour matching methods using EMD. Different from the EMD-based algorithm which considers each local feature as a cluster [36], we represent the input hand by global features (the finger clusters). And we add penalty on empty holes to alleviate partial matches on global features.

Formally, let $R = \{(\mathbf{r}_1, w_{\mathbf{r}_1}), \ldots, (\mathbf{r}_{\overline{m}}, w_{\mathbf{r}_{\overline{m}}})\}$ be the first hand signature with $\overline{m}$ clusters, where $\mathbf{r}_i$ is the cluster representative and $w_{\mathbf{r}_i}$ is the weight of the cluster; $T = \{(\mathbf{t}_1, w_{\mathbf{t}_1}), \ldots, (\mathbf{t}_{\overline{n}}, w_{\mathbf{t}_{\overline{n}}})\}$ is the second hand signature with $\overline{n}$ clusters. Now we show how to represent a time-series curve as a signature. Fig. 4(e) and (f) show the time-series

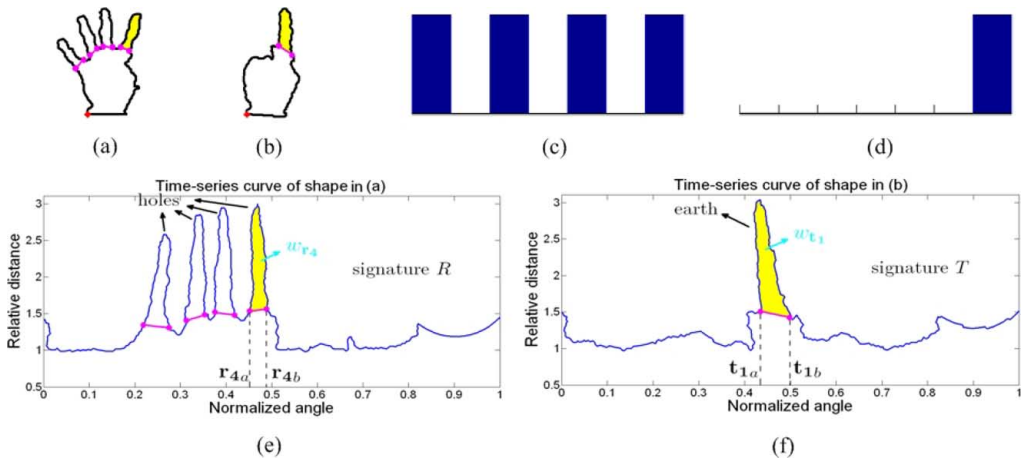


Fig. 4. (a) (b): two hand shapes whose time-series curves are shown in (e) (f). (c) (d): two signatures that partially match, whose EMD cost is 0. (e) (f): illustration of the signature representations of time-series curves.

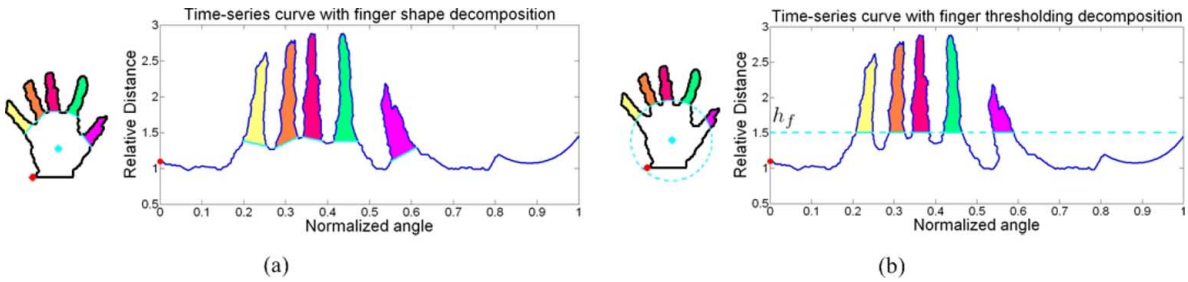


Fig. 5. The parts in color are the fingers detected by the proposed finger detection methods: (a) near-convex decomposition, (b) thresholding decomposition.

curves of the hands in Fig. 4(a) and (b) respectively, where each finger corresponds to a segment of the curve. We define each cluster of a signature as the finger segment of the time-series curve: the representative of each cluster $\mathbf{r}_i$ is defined as the angle interval between the endpoints of each segment, $\mathbf{r}_i = [\mathbf{r}_{ia}, \mathbf{r}_{ib}]$, where $0 \leq \mathbf{r}_{ia} < \mathbf{r}_{ib} \leq 1$; and the weight of a cluster, $w_{\mathbf{r}_i} \in (0, 1)$, is defined as the normalized area within the finger segment.

$\mathbf{D} = [d_{ij}]$ is the ground distance matrix of signature R and T, where d_{ij} is the ground distance from cluster $\mathbf{r}_i$ to $\mathbf{t}_j$. d_{ij} is defined as the minimum moving distance for interval $[\mathbf{r}_{ia},\ \mathbf{r}_{ib}]$ to totally overlap with $[\mathbf{t}_{ja},\ \mathbf{t}_{jb}]$, i.e.,:

$$d_{ij} = \begin{cases} 0, & \mathbf{r}_i \text{ totally overlap with } \mathbf{t}_j, \\ \min(|\mathbf{r}_{ia} - \mathbf{t}_{ja}|,\ |\mathbf{r}_{ib} - \mathbf{t}_{jb}|), & \text{otherwise.} \end{cases}$$

For two signatures, R and T, their FEMD distance is defined as the least work needed to move the earth piles plus the penalty on the empty hole that is not filled with earth:

$$\begin{aligned} \mathrm{FEMD}(R,T) &= \beta E_{move} + (1-\beta)E_{empty}, \\ &= \frac{\beta \sum_{i=1}^{\overline{m}} \sum_{j=1}^{\overline{n}} d_{ij} f_{ij} + (1-\beta)\left|\sum_{i=1}^{\overline{m}} w_{\mathbf{r}_i} - \sum_{j=1}^{\overline{n}} w_{\mathbf{t}_j}\right|}{\sum_{i=1}^{\overline{m}} \sum_{j=1}^{\overline{n}} f_{ij}}, \end{aligned}$$

where $\sum_{i=1}^{\overline{m}} \sum_{j=1}^{\overline{n}} f_{ij}$ is the normalization factor, f_{ij} is the flow from cluster $\mathbf{r}_i$ to cluster $\mathbf{t}_j$, which constitutes the flow matrix F. Parameter β modulates the importance between the first and the second terms. We will investigate the effects of β in Section IV-C. As we can see, E_{empty}, d_{ij} are constants for the two signatures; to compute the FEMD, we need to compute the value of $\mathbf{F}$. $\mathbf{F}$ is defined by minimizing the work needed to move all the earth piles:

$$\mathbf{F} = \arg\min \mathrm{WORK}(R,\ T,\ \mathbf{F}) = \arg\min \sum_{i=1}^{\overline{m}} \sum_{j=1}^{\overline{n}} d_{ij} f_{ij},$$

$$s.t. \begin{cases} f_{ij} \geq 0 & 1 \leq i \leq \overline{m}, 1 \leq j \leq \overline{n}, \\ \sum_{j=1}^{\overline{n}} f_{ij} \leq w_{\mathbf{r}_i} & 1 \leq i \leq \overline{m}, \\ \sum_{i=1}^{\overline{m}} f_{ij} \leq w_{\mathbf{t}_j} & 1 \leq j \leq \overline{n}, \\ \sum_{i=1}^{\overline{m}} \sum_{j=1}^{\overline{n}} f_{ij} = \min(\sum_{i=1}^{\overline{m}} w_{\mathbf{r}_i}, \sum_{j=1}^{\overline{n}} w_{\mathbf{t}_j}). \end{cases}$$

We follow the definition of the flow matrix $\mathbf{F}$ in EMD, as we also intend to find the minimum work needed to move the earth piles. The first constraint restricts the moving flow to one direction: from earth piles to the holes. The last constraint forces the maximum amount of earth possible to be moved. We will demonstrate the superiority of FEMD over EMD for contour matching in Section IV-C.

C. Finger Detection

In order to measure the FEMD distance between hand shapes, we need to represent the hand shape as a signature with each finger as a cluster, namely, to detect the finger parts from the hand shape. In Fig. 5, we propose two finger detection methods to obtain the finger parts from the hand shapes. Now we introduce these two algorithms:

1) Near-Convex Decomposition: We note that the fingers have a common geometric property: they are near-convex parts

of the hand shape. Therefore, we adjust the Minimum Near-Convex Decomposition (MNCD) proposed in [38], [39] to a finger detection method, which is illustrated in Fig. 5(a):

$$\begin{aligned} \mathbf{min} \quad & \alpha\|\mathbf{x}\|_{\mathbf{0}} + (\mathbf{1}-\alpha)\mathbf{w}^{\top}\mathbf{x}, \\ s.t. \quad & \mathbf{Ax} \geq \mathbf{1},\ \mathbf{Bx} \leq 1,\ \mathbf{x} \in \{0,1\}^{n}. \end{aligned}$$

The goal of the first term in the objective function is to reduce the redundant parts that are not fingers, and the second term is to improve the visual naturalness of the decomposition. Parameter α balances the influence between the first and the second term. We will investigate the effects of α in Section IV-C.

2) Thresholding Decomposition: Although near-convex decomposition algorithm can detect the finger parts accurately, it is generally complexly formulated and cannot be solved in real time. Thus we propose an alternative finger detection methods that are more efficient, named thresholding decomposition, as shown in Fig. 5(b).

As mentioned before, the time-series curve reveals a hand's topological information well. As shown in Fig. 5, each finger corresponds to a peak in the curve. Therefore, we can apply the height information in time-series curve to decompose the fingers. Specifically, we define a finger as a segment in the time-series curve, whose height is greater than a threshold h_f. In this way, we can detect the fingers fast. However, choosing a good height threshold h_f is essential. We will investigate the effects of h_f in Section IV-C.

IV. Evaluations

A. Dataset

We collect a new hand gesture dataset using Kinect sensor (http://eeeweba.ntu.edu.sg/computervision/people/home/renzhou/HandGesture.htm). Our dataset is collected from 10 subjects, and it contains 10 gestures for number 1 to 10. Each subject performs 10 different poses for the same gesture. Thus in total our dataset has $10\ \text{people} \times 10\ \text{gestures/people} \times 10\ \text{cases/gesture} = 1000\ \text{cases}$, each of which consists of a color image and the corresponding depth map. Our dataset is a very challenging real-life dataset, which is collected in cluttered backgrounds. Besides, for each gesture, the subject poses with variations in hand orientation, scale, articulation, etc.

B. Performance Evaluation

All experiments were done on a Intel Core™ 2 Quad 2.66 GHz CPU with 3 GB of RAM. Now we evaluate the performance of our system from the following aspects:

1) Robustness to Cluttered Backgrounds: Our hand gesture recognition system is robust to cluttered backgrounds, because the hand shape is detected using the depth information thus the backgrounds can be easily removed. Fig. 6(a) illustrates an example when the hand is cluttered by the background, which is hard for other hand gesture recognition methods that use colored markers to detect the hand. In Fig. 6(b), it shows a difficult case for the skin color-based hand gesture recognition approaches, where the hand is cluttered by the user's face.

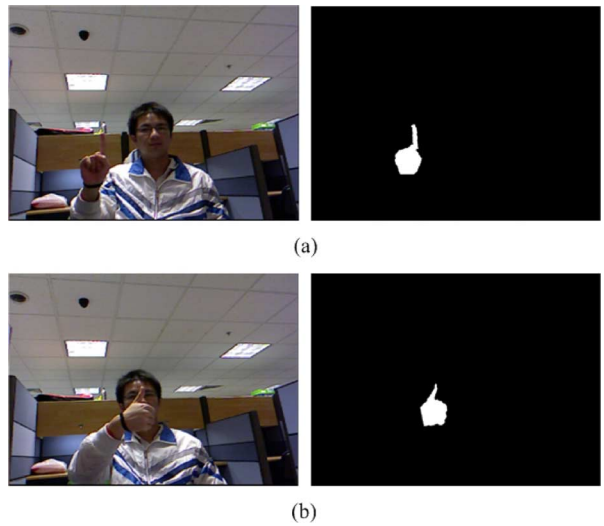


Fig. 6. Our system is robust to cluttered backgrounds. (a) The hand that is cluttered by background can be detected accurately; (b) The hand that is cluttered by face can be detected accurately.

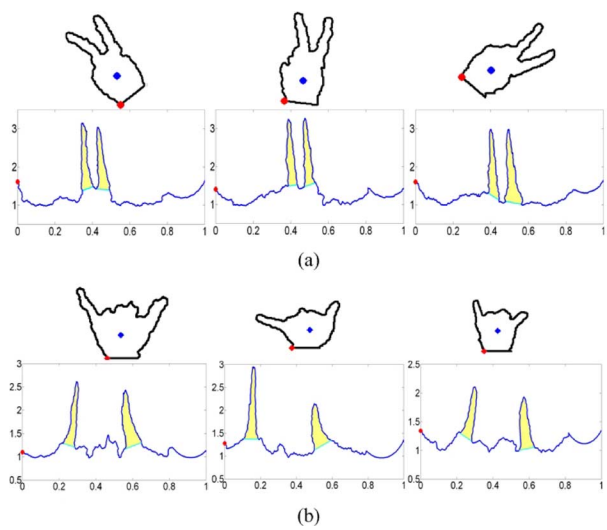


Fig. 7. Our method is robust to orientation and scale changes. (a) The hands with orientation changes, and their time-series curves; (b) The hands with scale changes, and their time-series curves.

However, our hand segmentation is very accurate using Kinect sensor, as shown in the right column of Fig. 6.

2) Robustness to Distortions and Hand Variations in Orientation, Scale, Articulation: In real-life environment, a hand can have variations on orientation, scale and articulation. Besides, because of the limited resolution of the depth map, the hand shapes are always distorted, or ambiguous. However, we can demonstrate that the proposed dissimilarity distance metric, Finger-Earth Mover's Distance (FEMD), is not only robust to the orientation and scale changes of the hand, but also insensitive to distortions and articulations.

Fig. 7(a) shows 3 hands with different orientations. As we can see, the initial point (the red point on the figure) and the center point (the blue point) are relatively fixed in these shapes. Thus the time-series curves of these hands (the second row in Fig. 7(a)) are similar, and their distances are very small. In

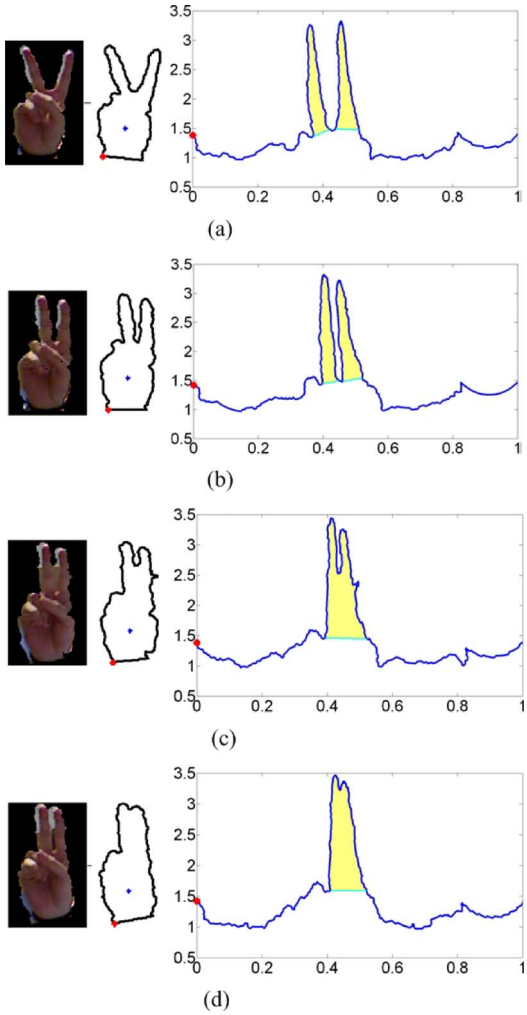


Fig. 8. Our system is insensitive to the distortions and articulation.

Fig. 7(b), there are 3 hands of different size. Because the time-series curve and the FEMD distance are normalized, they are correctly recognized as the same gesture. Hence we can conclude that FEMD is robust to orientation and scale changes.

Furthermore, our hand gesture recognition method is robust to the articulations and distortions brought by imperfect hand segmentation. Since the proposed FEMD distance metric uses global features (fingers) to measure the dissimilarity, local distortions are tolerable. As for articulations, Fig. 8 shows an example: the leftmost column shows 4 hand images of the same gesture; the middle column shows the corresponding hand shapes; and the rightmost column shows their time-series curves. As we can see, the hand shapes in Fig. 8(c) and (d) are heavily distorted. However, as illustrated in the rightmost column of Fig. 8, by detecting the finger parts (the yellow regions), we represent each shape as a signature whose clusters are the finger parts. Particularly, the signatures of Fig. 8(a) and (b) have 2 clusters: $\{(\mathbf{r}_1, w_{\mathbf{r}_1}), (\mathbf{r}_2, w_{\mathbf{r}_2})\}$, and the signatures of Fig. 8(c) and (d) only have 1 cluster: $\{(\mathbf{t}_1, w_{\mathbf{t}_1})\}$. From Section III-B2, we can estimate that $(w_{\mathbf{r}_1} + w_{\mathbf{r}_2}) \approx w_{\mathbf{t}_1}$, and the ground distance $d_{11}, d_{21} \approx 0$. According to the definition, we know that the FEMD distances among the 4 shapes≈ 0. Therefore, our FEMD metric is insensitive to distortions and articulations.

TABLE I
THE MEAN ACCURACY AND THE MEAN RUNNING TIME OF SHAPE CONTEXTS, SKELETON MATCHING, AND OUR METHODS. OUR PART-BASED HAND GESTURE RECOGNITION SYSTEM USING FEMD OUTPERFORMS THE TRADITIONAL SHAPE MATCHING ALGORITHMS

	Mean Accuracy	Mean Running Time
Shape Context without bending cost [13]	83.2%	12.346s
Shape Context with bending cost [13]	79.1%	26.777s
Skeleton Matching [3]	78.6%	2.4449s
Near-convex Decomposition+FEMD	**93.9%**	4.0012s
Thresholding Decomposition+FEMD	93.2%	**0.0750s**

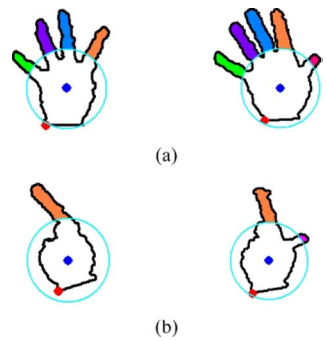


Fig. 9. Two pairs of confusing gestures in Experiment I. (a) Gesture 4 and 5. (b) Gesture 1 and 8.

3) Accuracy and Efficiency: In order to evaluate the accuracy and efficiency of our system, two experiments are conducted on the new dataset. Experiment I uses thresholding decomposition as discussed in Section III-C2 to detect the finger parts for FEMD measurement, and experiment II uses near-convex decomposition as illustrated in Section III-C1 for FEMD finger detection.

Experiment I: Thresholding Decomposition + FEMD: In experiment I, we fix the height threshold $h_f = 1.6$ and the FEMD parameter $\beta = 0.5$.

Fig. 10 is the confusion matrix of experiment I. The mean accuracy is 93.2%. As it shows, the two most confused gesture categories are gesture 5 and 4, gesture 8 and 1. Fig. 9 shows two confused cases of these categories.

Because the thumb is shorter and smaller, if decomposing the hands only by a height thresholding, important finger regions may be lost in some cases. As shown in Fig. 9, the thumbs are not well decomposed. As a result, the FEMD distances of these two cases are very small, which confuse the recognition.

However, thresholding decomposition is fast. Besides, due to the few number of extracted global features, FEMD operates efficiently. Table I gives the mean running time of a hand recognition procedure in experiment I, 0.0750 s. It should be noted that our FEMD algorithm is mainly rewritten and optimized in C++, rather than just using Matlab as in our previous work [1], which leads to the better results than [1]. As we see, thresholding decomposition based FEMD runs in real time.

Experiment II: Near-Convex Decomposition + FEMD: In order to more accurately decompose the fingers from the hands, we conduct another experiment which detects the fingers using near-convex decomposition. Here we fixed the near-convex decomposition parameter $\alpha = 0.5$ and the FEMD parameter $\beta = 0.5$. Fig. 12 shows some finger detection results of our near-convex decomposition algorithm. As we see, the finger parts

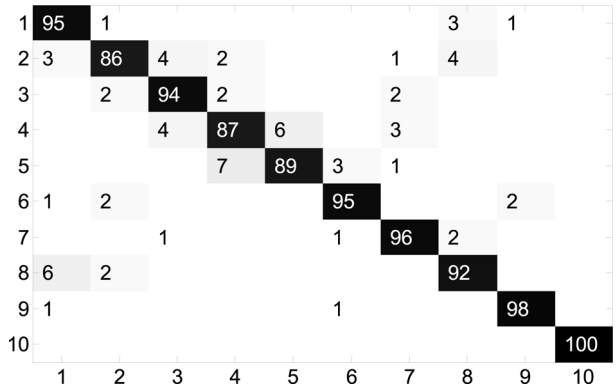


Fig. 10. The confusion matrix of Experiment I.

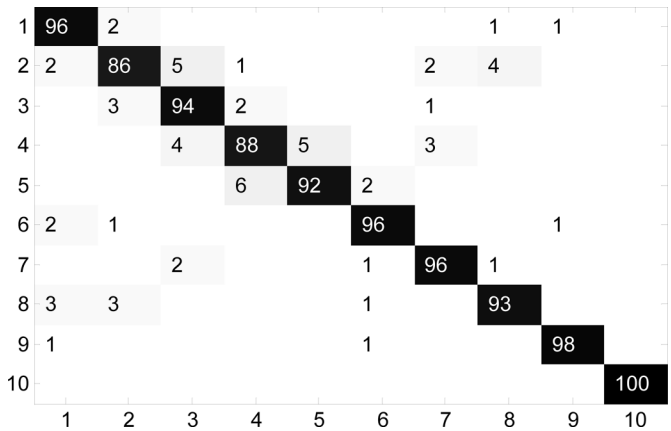


Fig. 11. The confusion matrix of Experiment II.

Fig. 12. Finger Detection results of Experiment II using near-convex decomposition algorithm.

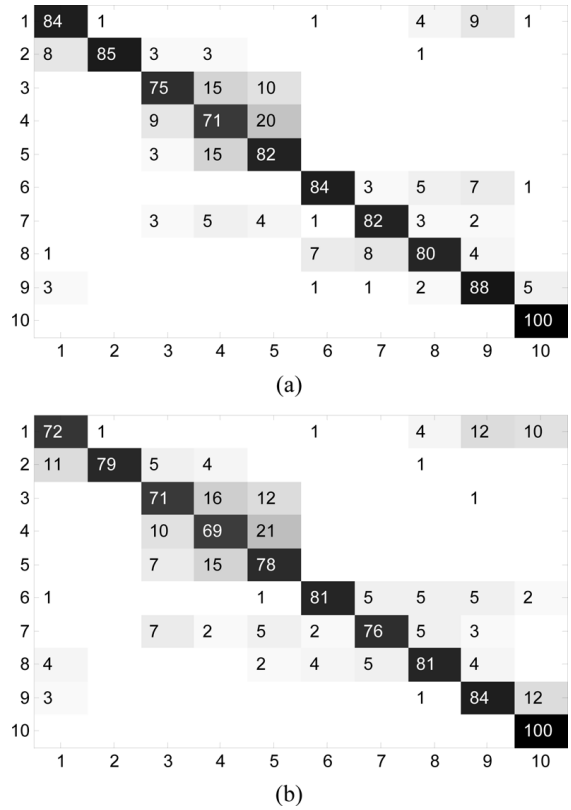


Fig. 13. The confusion matrix of hand gesture recognition using Shape Context [13]. (a) is the result of recognition computed without bending cost, and (b) is the result computed with bending cost.

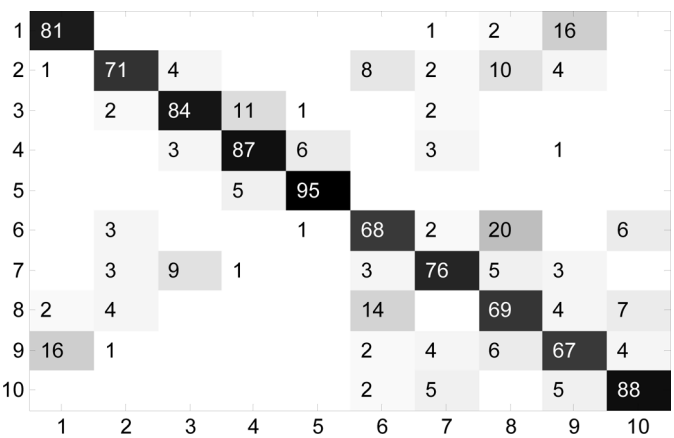


Fig. 14. The confusion matrix of hand gesture recognition using skeleton matching [3].

detection results are more accurate than thresholding decomposition.

Fig. 11 shows the confusion matrix of experiment II. There are no seriously confused categories. In the fourth row of Table I, the mean accuracy and the mean running time of experiment II are given. The mean accuracy of experiment II (93.9%) is higher than that of experiment I (93.2%) owing to the more accurate finger decomposition. On the other hand, the speed of experiment II (4.0012 s) is slower than that of experiment I (0.0750 s), because of the more complex finger detection algorithm.

4) Comparison With Other Methods: FEMD is a part-based hand matching metric. We compare it with the traditional correspondence-based matching algorithm, Shape Context [13] and the skeleton-based matching algorithm, Path Similarity [3]. Their mean accuracies and mean running times are given in Table I. We pre-segment the hand shape using the same method as ours in Section III-A.

Fig. 13 illustrates the confusion matrixes of Shape Context [13]. From both Fig. 13(a) and (b), we find that the most confusing classes are gesture 3, 4, and 5. The reason is that the fingers are more easily distorted in these classes, making them indistinguishable, which we have discussed before in Figs. 1 and 8. Fig. 15 shows some confusing cases for shape context where shapes are locally distorted.

From the first two rows of Table I, we notice that considering the bending cost of TPS transformation worsens the recognition

Fig. 15. Some confusing cases for shape context [13], where shapes are locally distorted.

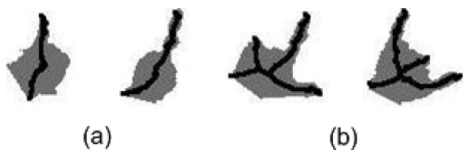


Fig. 16. Some confusing cases for skeleton matching [3], where very different shapes have similar skeletons.

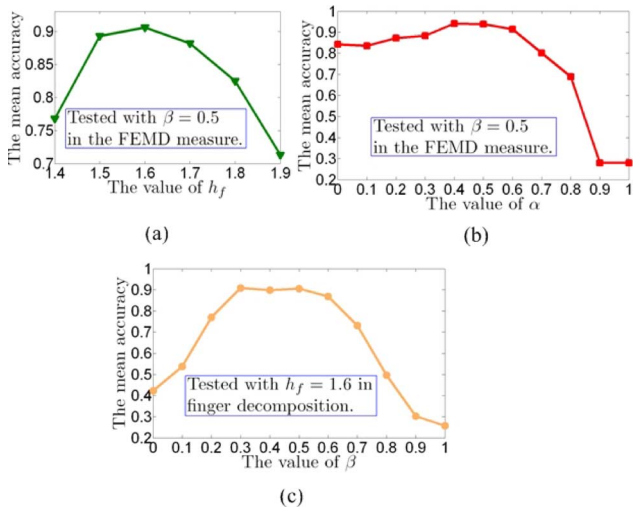


Fig. 17. Parameter sensitivity on h_f, α and β. When $\beta = 1$, FEMD becomes the EMD metric [33].

performance compared to shape contexts computed without the bending cost. The reason is that in order to be rotation invariant, shape context needs to treat the tangent vector at each point as the positive axis for the log-polar histogram frame. However, since our shape is binary, a small variation on the shape could cause severe change of the tangent vectors at points on the shape. Thus adding TPS bending cost worsens the performance.

Fig. 14 shows the confusion matrix of skeleton matching. We first prune the noisy skeleton using the method proposed in [2] and match them using Path Similarity proposed in [3]. From the figure, we notice that many gestures are severely confused, such as between gesture 1 and 9, gesture 6 and 8. The reason is that in those cases, their skeletons have very similar global structure. As shown in Fig. 16, very different hand gestures in (a) (b) have very similar skeletons. Thus skeleton matching algorithms are unable to differentiate these classes.

C. Parameter Sensitivity

In this section, we evaluate 3 important parameters—the height threshold h_f in thresholding decomposition finger detection method (Section III-C2), the parameter α in near-convex decomposition (Section III-C1), and the parameter β in FEMD formulation (Section III-B2).

The results are shown in Fig. 17. In thresholding decomposition, h_f determines the radius of the decomposing circle (see Fig. 5(b)). If h_f is too small (i.e., $h_f \leq 1.5$), the fingers cannot be well decomposed; and if h_f is too large (i.e., $h_f \geq 1.7$), essential finger regions will be lost. Fig. 17(a) shows that we can obtain the best result if setting h_f around 1.6. In finger detection using near-convex decomposition, α balances the impact of the visual naturalness and the number of parts. As shown in Fig. 17(b), if we only minimize the visual naturalness term (i.e., $\alpha = 0$), we will obtain noisy parts that affect the FEMD measure. Besides, the curve drops fast after $\alpha > 0.8$ because if minimizing the parts number too much while ignoring the visual naturalness, we may obtain parts that are not fingers. In the FEMD measure, β modulates importance between the earth-moving work E_{move} and the empty-hole penalty E_{empty}. Fig. 17(c) shows that if either only considering E_{move} (i.e., $\beta = 1$) or only considering E_{emtpy} (i.e., $\beta = 0$), FEMD cannot measure correct dissimilarity between hand shapes. This curve also justifies that FEMD is better than EMD (the special case when $\beta = 1$) for dissimilarity measure between hand shapes.

V. Applications

Lately there has been a great emphasis on Human-Computer Interaction (HCI) research to create easy-to-use interfaces by facilitating natural communication and manipulation skills of humans. Among different human body parts, the hand is the most effective interaction tool because of its dexterity. Adopting hand gesture as an interface in HCI will not only allow the deployment of a wide range of applications in sophisticated computing environments such as virtual reality systems and interactive gaming platforms, but also benefit our daily life such as providing aids for the hearing impaired, and maintaining absolute sterility in health care environments using touchless interfaces via gestures [4].

Now we propose to use the hand gesture as an interface, and introduce two real-life HCI applications on top of our hand gesture recognition system: Arithmetic computation and Rock-paper-scissors game. It should be noticed that, in the demo system, we perform frame-based hand gesture recognition using the FEMD distance metric based on thresholding decomposition finger detection method, taking both accuracy and efficiency into consideration. As shown in Table I, it runs in real time and achieves comparable accuracy as that of FEMD metric based on finger detection using near-convex decomposition.

A. Arithmetic Computation

Arithmetic computation is an interesting HCI application. Instead of interacting with the computer by the keyboard or mouse, we input arithmetic commands to the computer via hand gestures. As shown in Fig. 19, 14 hand gestures are defined to represent 14 commands, namely number 0–9 and operator $+$, $-$, $\times$, $\div$, respectively.

By recognizing each input gesture as a command, the computer can perform arithmetic computations instructed by the

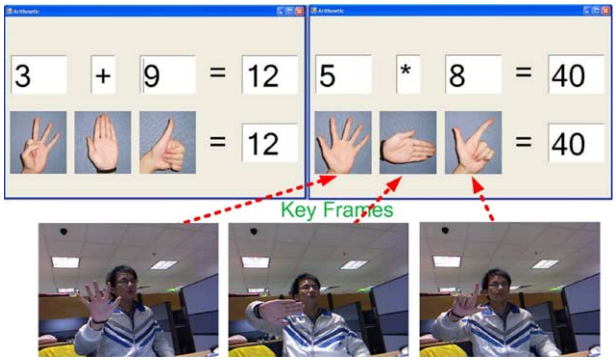


Fig. 18. Arithmetic computation.

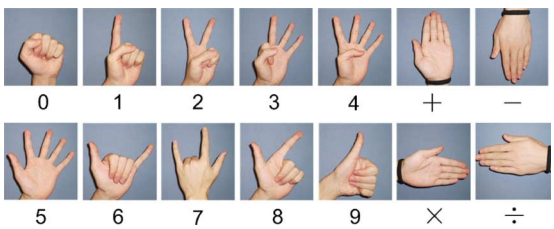


Fig. 19. The 14 gesture commands in our arithmetic computation system.

user. Two examples are shown in Fig. 18. The key frames are shown as well.

B. Rock-Paper-Scissors Game

Rock-paper-scissors is a traditional game. The rule is rock breaks scissors; scissors cut paper; and paper wraps rock. In this demo, we build a Rock-paper-scissors game system played between a human and a computer. The computer randomly chooses a weapon, and the user's gesture is recognized by our system. According to the game rule, our system can decide who is the winner. Fig. 20 shows two examples.

These two demos have been demonstrated in ACM Multimedia 2011, etc. It runs accurately in real time. It is feasible to build more interesting Kinect demos on top of our hand gesture recognition system. The hand gesture dataset we collected with Kinect sensor and the technical demo video [40] showing these two HCI applications are available at http://eeeweba.ntu.edu.sg/computervision/people/home/renzhou.

VI. Conclusion and Future Work

Hand gesture recognition for real-life applications is very challenging because of its requirements on the robustness, accuracy and efficiency. In this paper, we presented a robust part-based hand gesture recognition system using the Kinect sensor. A novel distance metric, Finger-Earth Mover's Distance (FEMD), is used for dissimilarity measure, which represents the hand shape as a signature with each finger part as a cluster and penalizes the empty finger-holes. Extensive experiments on a challenging 10-gesture dataset validate that our part-based hand gesture recognition system is accurate and efficient. More specifically, our FEMD based hand gesture recognition system achieves 93.2% mean accuracy and runs in 0.0750 s per frame when using the thresholding decomposition finger detection method. And it achieves better accuracy of 93.9% when using a more accurate finger detection method, however, at the cost of efficiency. Taking both accuracy and efficiency into consideration, we use thresholding decomposition for finger detection in our real time demo system.

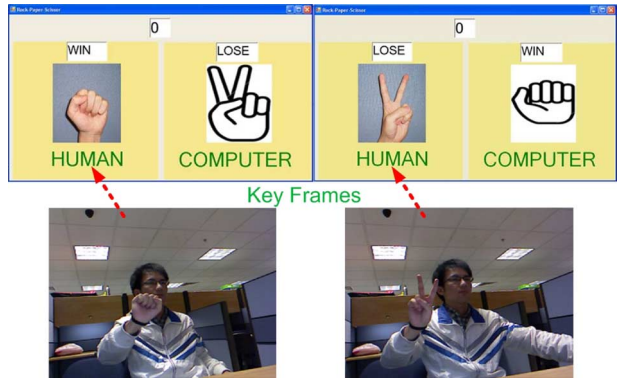


Fig. 20. Rock-paper-scissors game.

One major contribution of our paper is the distance metric based on part-based representation. Traditional distance measures such as shape contexts distance and path similarity is not robust to local distortions and shape variations, since their representations, i.e., shape contexts and skeleton, are not consistent in the case of hand variations or severe local distortions. The proposed FEMD distance metric is based on a part-based representation which represents a hand shape as a signature with each finger part as a cluster. Such a representation enables the computation on the global features, thus it is robust to local distortions. And it is robust to articulation, orientation, scale changes, as discussed in Section IV-B2.

Another contribution of this paper is the real-life HCI applications we built on top of our hand gesture recognition system. It shows that with hand gesture recognition technique we can mimic the communications between human, and involve hand gesture as a natural and intuitive way to interact with machines. Consequently we can benefit our daily life in many aspects such as providing aids for the hearing impaired, and maintaining absolute sterility in health care environments using touchless interfaces via gestures.

Our future research will focus on exploring a more efficient part-based representation, to handle the problem shown in Fig. 9 and the efficiency drawback of near-convex decomposition based finger detection method. And we will further develop interesting HCI applications of our hand gesture recognition system.

References

[1] Z. Ren, J. Yuan, and Z. Zhang, "Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera," in *Proc. ACM Int. Conf. Multimedia*, 2011, pp. 1093–1096.

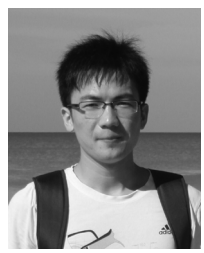
[2] X. Bai, L. J. Latecki, and W.-Y. Liu, "Skeleton pruning by contour partitioning with discrete curve evolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 449–462, 2007.

[3] X. Bai and L. J. Latecki, "Path similarity skeleton graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, pp. 1–11, 2008.

[4] J. P. Wachs, M. Kolsch, H. Stern, and Y. Edan, "Vision-based hand-gesture applications," *Commun. ACM*, vol. 54, pp. 60–71, 2011.

[5] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly, "Vision-based hand pose estimation: A review," *Comput. Vision Image Understand.*, vol. 108, pp. 52–73, 2007.

[6] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, pp. 311–324, 2007.
[7] G. R. S. Murthy and R. S. Jadon, "A review of vision based hand gesture recognition," *Int. J. Inf. Technol. Knowl. Manage.*, vol. 2, pp. 405–410, 2009.
[8] G. Dewaele, F. Devernay, and R. Horaud, "Hand motion from 3D point trajectories and a smooth surface model," in *Proc. Eur. Conf. Computer Vision*, Prague, Czech Republic, 2004, pp. 495–507.
[9] J. Shotton, A. Fitzgibbon, M. Cook, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Colorado Springs, CO, USA, 2011.
[10] Q. Cai, D. Gallup, C. Zhang, and Z. Zhang, "3D deformable face tracking with a commodity depth camera," in *Proc. Eur. Conf. Computer Vision*, Crete, Greece, 2010, pp. 229–242.
[11] KinectHacks, Microsoft Kinect Application Demos. [Online]. Available: http://kinecthacks.net/.
[12] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Providence, RI, USA, 2012.
[13] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, pp. 509–522, 2002.
[14] H. Ling and D. W. Jacobs, "Shape classification using the inner-distance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 286–299, 2007.
[15] K. Siddiqi, S. Bouix, A. R. Tannenbaum, and S. W. Zucker, "Hamilton-Jacobi skeletons," *Int. J. Comput. Vision*, vol. 48, pp. 215–231, 2002.
[16] A. Wilson and A. Bobick, "Parametric hidden markov models for gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, pp. 884–900, 1999.
[17] H. Lee and J. Kim, "An hmm-based threshold model approach for gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, pp. 961–973, 1999.
[18] M.-C. Su, "A fuzzy rule-based approach to spatio-temporal hand gesture recognition," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 30, pp. 276–281, 2000.
[19] C. Kwok, D. Fox, and M. Meila, "Real-time particle filters," *Proc. IEEE*, pp. 469–484, 2004.
[20] A. Doucet, N. de Freitas, and N. Gordon*, Sequential Monte Carlo in Practice*. New York, NY, USA: Springer-Verlag, 2001.
[21] C. Chua, H. Guan, and Y. Ho, "Model-based 3D hand posture estimation from a single 2D image," *Image Vision Comput.*, vol. 20, pp. 191–202, 2002.
[22] N. Shimada, Y. Shirai, Y. Kuno, and J. Miura, "Hand gesture estimation and model refinement using monocular camera-ambiguity limitation by inequality constraints," in *Proc. IEEE Int. Conf. Face and Gesture Recognition*, Nara, Japan, 1998, pp. 268–273.
[23] E. Holden, "Visual recognition of hand motion," Ph.D. dissertation, Dept. Comput. Sci., Univ. Western Australia, Crawley, Australia, 1997.
[24] Y. Fang, K. Wang, J. Cheng, and H. Lu, "A real-time hand gesture recognition method," in *Proc. IEEE Int. Conf. Multimedia and Expo*, 2007, pp. 995–998.
[25] D. Lowe, "Fitting parameterized 3D models to images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, pp. 441–450, 1991.
[26] T. Starner, J. Weaver, and A. Pentland, "Real-time american sign language recognition using desk and wearable computer based video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, pp. 1371–1375, 1998.
[27] M. H. Yang, N. Ahuja, and M. Tabb, "Extraction of 2-d motion trajectories and its application to hand gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 1062–1074, 2002.
[28] M. Bray, E. Koller-Meier, and L. V. Gool, "Smart particle filtering for 3D hand tracking," in *Proc. IEEE Int. Conf. Face and Gesture Recognition*, Los Alamitos, CA, USA, 2004, pp. 675–680.
[29] J. Lin, Y. Wu, and T. Huang, "3D model-based hand tracking using stochastic direct search method," in *Proc. IEEE Int. Conf. Face and Gesture Recognition*, Seoul, Korea, 2004, pp. 693–698.
[30] J. Segen and S. Kumar, "Gesture vr: Vision-based 3d hand interface for spatial interaction," in *Proc. ACM Int. Conf. Multimedia*, 1998, pp. 455–464.
[31] M. Reale, S. Canavan, L. Yin, K. Hu, and T. Hung, "A multi-gesture interaction system using a 3D iris disk model for gaze estimation and an active appearance model for 3D hand pointing," *IEEE Trans. Multimedia*, vol. 13, pp. 474–486, 2011.
[32] E. Keogh, L. Wei, X. Xi, S. Lee, and M. Vlachos, "Lb_keogh supports exact indexing of shapes under rotation invariance with arbitrary representations and distance measures," in *Proc. Int. Conf. Very Large Databases*, 2006, pp. 882–893.
[33] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *Int. J. Comput. Vision*, vol. 40, pp. 99–121, 2000.
[34] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv.*, vol. 40, pp. 1–60, 2008.
[35] J. Zhang, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *Int. J. Comput. Vision*, vol. 73, pp. 213–238, 2007.
[36] K. Grauman and T. Darrell, "Fast contour matching using approximate earth mover's distance," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Washington, DC, USA, 2004, pp. 220–227.
[37] H. Ling and L. Okada, "An efficient earth mover's distance algorithm for robust histogram comparison," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 840–853, 2007.
[38] Z. Ren, J. Yuan, C. Li, and W. Liu, "Minimum near-convex decomposition for robust shape representation," in *Proc. IEEE Int. Conf. Computer Vision*, 2011, pp. 303–310.
[39] Z. Ren, J. Yuan, and W. Liu, "Minimum near-convex shape decomposition," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.
[40] Z. Ren, J. Meng, J. Yuan, and Z. Zhang, "Robust hand gesture recognition with Kinect sensor," in *Proc. ACM Int. Conf. MultiMedia*, Scottsdale, AZ, USA, 2011, pp. 759–760.

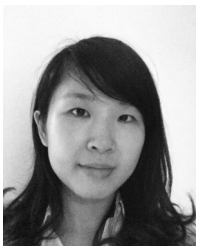


Zhou Ren is currently a Ph.D. student in Department of Computer Science at University of California, Los Angeles. He received the M.Eng. degree with award from the School of Electrical and Electronic Engineering at Nanyang Technological University, Singapore, in 2012. Before that, he received the B.Eng. with highest honor from the Department of Electronics and Information Engineering in Huazhong University of Science and Technology, China, in 2010.

From 2010 to 2012, he was a project officer in the Media Technology Lab at Nanyang Technological University. His research interests include computer vision, human-machine interaction, and statistical machine learning.

Junsong Yuan (M'08) is a Nanyang Assistant Professor at School of EEE, Nanyang Technological University (NTU). He currently serves as the Program Director of Video Analytics at Infocomm Center of Excellence (INFINITUS), School of EEE, NTU. He received his Ph.D. from Northwestern University, Illinois, USA and M.Eng. from National University of Singapore. He was selected to the special Class for the Gifted Young of Huazhong University of Science and Technology and received a B.Eng. of Communication Engineering in 2002.

Dr. Yuan's research interests include computer vision, video analytics, visual search and mining, human computer interaction, etc. He has published over 90 papers in leading journals and conferences of computer vision, pattern recognition, data mining, and multimedia. He serves as editor, co-chair, PC member and reviewer of many international journals/conferences/workshops/special sessions. He received Outstanding EECS Ph.D. Thesis award from Northwestern University, and the Doctoral Spotlight Award from IEEE Conference Computer Vision and Pattern Recognition Conference (CVPR'09). He gives tutorials of human action analysis and video analytics at a few conferences such as ICIP'13, FG'13, ICME'12, SIGGRAPH VRCAI'12, and PCM'12. He has filed three US patents and two provisional US patents.



Jingjing Meng (M'09) received the B.E. degree in electronics and information engineering from Huazhong University of Science and Technology, China, in 2003, and the M.S. degree in computer science from Vanderbilt University, Nashville, TN, USA, in 2006.

From 2007 to 2010, she was a Senior Research Staff Engineer with Motorola Applied Research Center, Schaumburg, IL, USA. Currently she is a Research Associate and part-time Ph.D. student at the School of Electrical and Electronic Engineering. She has filed two US patents and two US provisional patent applications. Her current research interests include computer vision, human computer interaction, and image and video analysis.

Zhengyou Zhang (F'05) 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 the research manager of the Multimedia, Interaction, and Communication (MIC) Group. 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 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.