A Modified EM Algorithm for Hand Gesture Segmentation in RGB-D Data

Zhaojie Ju, Yuehui Wang, Wei Zeng, Haibin Cai and Honghai Liu

Abstract— **This paper proposes a novel method with a modified Expectation-Maximisation (EM) Algorithm to segment hand gestures in the RGB-D data captured by Kinect. With the depth map and RGB image aligned by the genetic algorithm to estimate the key points from both depth and RGB images, a novel approach is proposed to refine the edge of the tracked hand gesture, which is used to segment the RGB image of the hand gestures, by applying a modified EM algorithm based on Bayesian networks. The experimental results demonstrated the modified EM algorithm effectively adjusts the RGB edges of the segmented hand gestures. The proposed methods have potential to improve the performance of hand gesture recognition in Human-Computer Interaction (HCI).**

I. Introduction

Recently, the problem of acquisition and recognition of human hand gesture from RGB-Depth (RGB-D) sensors, such as Microsofts Kinect, is an important subject in the area of the computer vision and pattern analysis. In order to extract and recognise hand gestures from RGB-D data, many researchers conducted significant contribution, including the hand gesture extracting, tracking, recognising and so on[1], [2], [3]. These achievements are of much importance for the research in areas of human-computer interaction (HCI). The Kinect developed by Microsoft Corporation is largely welcomed by the researchers, as it can simultaneously acquire data of RGB image and depth map of the scene by its IR emitter and camera sensors. Its broad applications covers 3D reconstruction [4], [5], image processing [6], human-machine interface [7], [8], [9], [2], robotics [10], object recognition [11], [12], just to name a few[13], [14]. However, there are many problems such as distortion and disaccord of depth and RGB images in corresponding pixels, especially the limitations in extracting of correct human hand gestures. Due to the noises and holes in the RGB-D data, precisely segmenting the hand gestures is still a challenging task.

In computer vision, camera calibration is a necessary step in scene reconstruction in order to extract metric information from images [15]. This includes internal calibration of each camera as well as external parameters of relative pose calibration between the cameras. Colour camera calibration has been studied extensively and different calibration techniques have been developed for depth sensors depending on the circumstances [3]. In a similar manner, the calibration of RGB image and depth map is much essential for their consistency and synchronisation. For recovering and tracking the 3D position, orientation and full articulation of a human hand from markerless visual observations, an algorithm of minimising the discrepancy between the appearance and 3D structure of hypothesized instances of a hand model and actual hand observations was developed in [16]. Li implemented a novel algorithm for contactless hand gesture recognition, and it is a real-time system which detects the presence of gestures, to identify fingers, and to recognise the meanings of nine gestures in a predefined popular gesture scenario [17]. For handling the noisy hand shapes obtained from the Kinect sensor, Zhang designed a approach of distance metric for hand dissimilarity measure, called Finger-Earth Movers Distance [15]. As it only matches fingers while not the whole hand shape, it can better distinguish hand gestures of slight differences. In [18], Van *et al.* designed a robust and real-time system of 3D hand gesture interaction with a robot for understanding directions from humans. The system was implemented to detect hand gestures in any orientation and more in particular pointing gestures while extracting the 3D pointing direction.

Because of the complexity and dexterity of the human hand, recognising the unconstrained human hand motions is a fundamental challenge in existing algorithms [19]. Kinect provides a promising way to realise stable, effective and natural human-computer interaction [20], [1]. The rest of this paper is organised as follows. The problem of hand gesture segmentation via Kinect is given in Section II; Depth and RGB image alignment is introduced in Section III. Hand gesture segmentation using an EM algorithm is proposed in section IV. Experimental results are discussed in Section V. Conclusions are followed in Section VI.

II. Problem of Hand Gesture Segmentation via Kinect

Depth and colour/RGB images are simultaneously captured by Kinect at a frame rate of up to 30 fps. More than 300,000 depth-coloured points are captured in each frame. One "perfect" frame will consist of these points with absolutely correct alignment of the depth and colour data. However, due to the limitations of the systematic design and the random errors, the alignment of the depth and RGB images highly relies on the identification of the mathematical model of the measurement and the calibration parameters involved. The characterisation of random errors is important and useful in further processing of the depth data, for example in weighting the point pairs or planes in the registration algorithm [21], [22].

Z. Ju, W. Zeng, and H. Liu are with Intelligent Systems & Biomedical Robotics group, University of Portsmouth, Portsmouth, UK. Y. Wang and H.Cai are with Zhejiang University of Technology, Hangzhou, China. This paper is supported by State Key Lab of Digital Manufacturing Equipment Technology, China, DMETKF2013001.

(a) RGB (b) Uncalibrated (c) Calibrated

Fig. 1: Hand gesture segmentation (a) RGB image; (b)uncalibrated; (c) Calibrated using official calibration

A proprietary algorithm is used to calibrate Kinect devices when manufacturing, and these calibrated parameters stored in the devices' internal memory are used to perform the image construction. The official calibration is adequate for human body motion analysis or casual use, but it lacks accuracy in hand gesture segmentation and recognition. Fig. 1b shows the result of hand segmentation based on depth threshold without official calibration from the RGB image in Fig. 1a, and it shows the colour finger can not be seen and the mismatch between the depth and colour images is huge. Fig. 1c shows the result using the official calibration. It clearly indicates that only half of the finger can be seen in the segmented RGB image, and this will severely affect the further hand gesture recognition. Other calibration algorithms have been proposed to solve the problem of the disparity/depth distortion, *e.g.* Smisek *et al.* [23] introduced a depth distortion correction component as the average of the residuals in metric coordinates, while Daniel *et al.* [3] propose a disparity distortion correction that depends on the observed disparity which further improves accuracy. These algorithms require a lot calibrating images and the optimisations are based on the whole scene, which means they are not practical and may sacrifice the precision of local space to achieve an overall minimisation. Since depth range of the Kinect devices is around 50cm to 5m and the resolution is about 1.5mm at 50cm and 5cm at 5m, the hand, as a small part of the body, needs to be closer to the camera to get a clearer image and it asks for higher precision in depth and RGB image alignment for hand segmentation and then for hand gesture recognition. In addition, due to the noise and holes of the depth data, the image segmentation based on the depth information has lots of mismatched pixels including the background pixels in the segmented objects and object pixels left in the background [14]. This problem with mismatched pixels has not been addressed in the current literature. In this paper, a two-step novel method is proposed to precisely segment the hand gestures using RGBD image. Genetic algorithm is used to match the depth mag with RGB image, and then an Expectation-Maximisation (EM) algorithm is proposed to further adjust the segmentation edge based on the depth map, RGB image and locations of the pixels.

III. Hand Gesture Segmentation using a modified EM algorithm

The depth and RGB images have been roughly adjusted and aligned using the alignment method in [24]. However, due to the noises and holes in the RGB-D data, the colour map of the human hand can not be effectively segmented using only the depth information [14].

A. The proposed EM Algorithm

Each pixel in the Kinect image has RGB values, a depth value and its 2D location, based on which the estimation of the probability of this pixel belonging to the hand gesture can be expressed by $p(H=1|RGB,D,L)$ or $p(H|RGB,D,L)$ where D is Depth and L is Location. H is a binary variable indicating whether a pixel belongs to a hand or not, when $H=1$ or H means this pixel belongs to a hand and $H=0$ or $\tilde{H}$ means this pixel does not. The events of *RGB*, *Depth* and *Location* can be reasonably assumed to be independent, and according to the Bayesian Network, we can have

$$\begin{aligned} p(H|RGB,D,L) &= \frac{p(H)p(RGB|H)p(L|H)p(D|H)}{\sum_H p(H)p(RGB|H)p(L|H)p(D|H))} \\ &= \frac{p(H)p(RGB|H)p(L|H)p(D|H)}{p(H)p(RGB|H)p(L|H)p(D|H)+p(\tilde{H})p(RGB|\tilde{H})p(L|\tilde{H})p(D|\tilde{H})} \end{aligned} \tag{1}$$

where $p(H)$ is prior probability of the hand gesture; $p(RGB|H)$ is the probability of the RGB value given that this pixel is part of a hand and it assumes to be a Gaussian distribution with a mean of μ_{RGBH} and a covariance of Σ_{RGBH}; $p(D|H)$ is the probability of the depth value given this pixel is part of a hand and it assumes to be Gaussian distributed with a mean of μ_D and a covariance of Σ_D; $p(L|H)$ is the probability of the pixel location given this pixel belongs to a hand and its distribution is given below:

$$p(L|H) = \frac{1}{2}\left(erf(\frac{\text{dist(L)}}{\sqrt{2}\delta_L})+1\right) \tag{2}$$

where function erf is a Gauss error function:

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x \mathrm{e}^{-t^2} \mathrm{d}t \tag{3}$$

and the function dist(L) is to get the minimum distance between the pixel and the hand edge. dist(L) is negative when the pixel is inside of the edge and positive when outside of the edge. $p(\tilde{H})$ is the probability of this pixel not belonging to a hand, and $p(\tilde{H}) = 1 - p(H)$; $p(RGB|\tilde{H})$ is the probability of the RGB value given that this pixel is part of the background and it assumes to be a Gaussian distribution with a mean of μ_{RGBB} and a covariance of Σ_{RGBB}; $p(D|\tilde{H})$ is the probability of the depth value given this pixel is part of the background and it assumes to be of uniform distribution $unif(depth_{min}, depth_{max})$, where the $depth_{min}$ is the minimum of the depth value in the scene and $depth_{max}$ is the maximum; $p(L|\tilde{H})$ is the probability of the pixel location given this pixel belongs to the background and $p(L|\tilde{H}) = 1 - p(L|H)$.

The parameters are $\Theta = (\mu_{RGBH}, \Sigma_{RGBH}, \mu_{DH}, \Sigma_{D\tilde{H}}, \Sigma_{RGB\tilde{H}}, \mu_{D\tilde{H}})$. The resulting density for the samples is

$$p(\mathscr{U}|\Theta) = \prod_{t=1}^{n} p(x_t|\Theta) = \mathscr{L}(\Theta|\mathscr{U}) \tag{4}$$

where $\mathscr{U}$ means all the captured pixel information including the *RGB*, *Depth* and *Location* and $\mathscr{U} = \{u1, \ldots, u_n\}$, $u_i = \{RGB_i, D_i, L_i\}$ and n is the number of the pixels.The function $\mathscr{L}(\Theta|\mathscr{U})$ is called the likelihood of the parameters given the data, or the likelihood function. The likelihood is considered as a function of the parameters Θ where the data $\mathscr{U}$ is fixed. In the maximum likelihood problem, the objective is to estimate the parameters set Θ that maximizes $\mathscr{L}$. That is to find Θ^* where

$$\Theta^* = \arg\max_{\Theta} \mathscr{L}(\Theta|\mathscr{U}) \tag{5}$$

Usually, the EM algorithm (e.g., [25], [26]) is proposed to maximise the $\mathscr{L}$. The iteration of an EM algorithm estimating the new parameters in terms of the old parameters is proposed and given as follows:

- E-step: compute "expected" classes of all pixels for hand gesture and background, $p(H|RGB_t, D_t, L_t)$ and $p(\tilde{H}|RGB_t, D_t, L_t)$ using Eq. 1.
- M-step: compute maximum likelihood given the pixel class membership distributions according to equations 6-11.

$$p(H)^{new} = \frac{1}{n} \sum_{t=1}^{n} p(H|RGB_t, D_t, L_t); \tag{6}$$

$$p(\tilde{H})^{new} = 1 - p(H) \tag{7}$$

$$[\mu_{RGBJ}^{new}, \mu_{DJ}^{new}] = \frac{\sum_{t=1}^{n} p(J|RGB_t, D_t, L_t)[RGB_i, D_i]}{\sum_{t=1}^{n} p(J|RGB_t, D_t, L_t)}; J = H/\tilde{H} \tag{8}$$

$$\Sigma_{RGBJ}^{new} = \frac{\sum_{t=1}^{n} p(J|RGB_t, D_t, L_t)(RGB_t - \mu_{RGBJ}^{new})(RGB_t - \mu_{RGBJ}^{new})^T}{\sum_{t=1}^{n} p(J|RGB_t, D_t, L_t)}; \quad J = H/\tilde{H} \tag{9}$$

$$\Sigma_{DH}^{new} = \frac{\sum_{t=1}^{n} p(H|RGB_t, D_t, L_t)(D_t - \mu_{DH}^{new})(D_t - \mu_{DH}^{new})^T}{\sum_{t=1}^{n} p(H|RGB_t, D_t, L_t)} \tag{10}$$

$$edge^{new} = f(p(H|RGB, D, L)) \tag{11}$$

where $f()$ is the function to estimate the new edge of the hand according to the probabilities of all pixels belonging to the hand gesture, and its details are given in Sec. III-B .

B. Edge Estimation

The probability of $p(H|RGB_i, D_i, L_i)$ can be normalised as:

$$p'(H|RGB_i, D_i, L_i) = \begin{cases} 0, p(H|RGB_i, D_i, L_i) < 0.01 \\ 1, p(H|RGB_i, D_i, L_i) > 0.99 \\ p(H|RGB_i, D_i, L_i), else \end{cases} \tag{12}$$

according to $p'(H|RGB_i, D_i, L_i)$, we can easily have two edges: external edges $\{x_i^E\}, i = 1, \ldots, n^E$ for all pixels whose probabilities are less than 0.01 and internal edges $\{x_i^I\}, i = 1, \ldots, n^I$ whose probabilities are more than 0.99. n^E and n^I are the number of the edge points on external edge and internal edge respectively. For each point x_i^E on the external edge, there is a point x_j^I who has a minimum distance between x_i^E and the internal edge points; similarly, for each point x_i^I on the internal edge, there is a point x_j^E who has a minimum distance between x_i^I and the internal edge points. Assume the points pairs $(x_i^I, x_i^E), i = 1, \ldots, n^P$ are the unique pairs with such minimum distances, n^P is the number of those unique pairs. Assume a line l_i is determined by the pair points (x_i^I, x_i^E), we can find the pixels x_k^i who are near to this line and whose probabilities are less than 0.99 and more than 0.01, as shown in green circle in Fig. 2. Based on the near points and their projection point, we can estimate the edge model on this line by

$$[\delta_i, a_i] = \arg\min_{\delta_i, a_i} \sum_{k=1}^{n^P} \left(\frac{1}{2} \left(\mathrm{erf}\left(\frac{D(pr(x_k^i), x_i^E) - a_i}{\sqrt{2}\delta_i} \right) + 1 \right) - p(H|x_k^i) \right) \tag{13}$$

where $pr(x_k^i)$ is the projection point of x_k^i and $D(x_i, x_j)$ is the distance between the location x_i and x_j; The minimum problem can be solved by Least-Square Fitting method. One example of the fitting results is given in Fig. 3. Then the edge point on the line l_i can be found by

$$edge_i^{new} = \frac{a_i}{D(x_i^I, x_i^E)} (x_i^I - x_i^E) + x_i^E \tag{14}$$

as shown in red circle in Fig. 2.

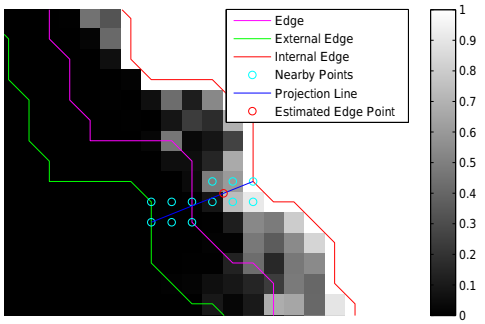


Fig. 2: An example shows three edge lines (external edge in green, original edge in magenta and internal edge in red) based on the probabilities of the pixels belonging to a hand (the probability is shown in a grayscale), and the estimated edge point in red circle has been identified based on the pixels in green circles near to the projection line

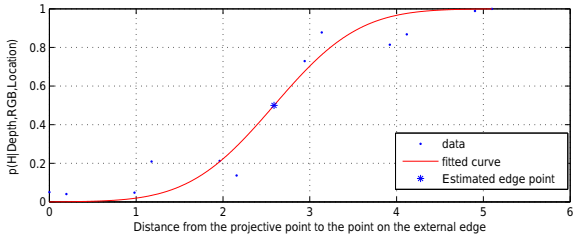


Fig. 3: An example of the fitting result. The projection points on the line are in blue dots; the fitting curve is in red line; the estimated edge point is in blue star.

C. Implementation

To initialise the parameter set Θ, the hand gesture will be segmented based only on the depth information. Firstly, Spatial-temporal filtering (STF) [27], [28] is employed to track the hand position and based on the tracking result the hand initial depth can be automatically chosen, as shown in Fig. 4. The initial edge of the hand gesture can be achieved using Sobel method [29]. The pixels in the hand edge belong to hand with a full probability to the hand gesture, $p(H|RGB_i, D_i, L_i) = 1$,and others have a full probability to the background, $p(\tilde{H}|RGB_i, D_i, L_i) = 1$. The parameter set can be achieved by equations 6 to 10. The EM algorithm for segmenting the hand gesture is shown in Algorithm 1 in the appendix.

IV. Experiment Results

The above algorithm has been implemented in Matlab. Various data have been collected and vaulted to show its performance. Genetic algorithm can always find the best solution due to the pre-set searching bound for each variable and the close precise initialisation [24].

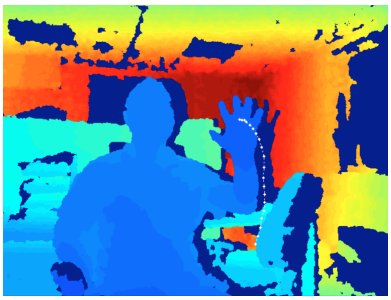


Fig. 4: Hand tracking using Spatical-Temporal filtering [27], [28], and the hand trajectory is shown in white dots.

Generation	f-count	Best f(x)	Max Constraints	Stall Generations
1	1060	1.06589	0	0
2	2100	0.793731	0	0
3	3140	0.784879	0	0
4	4180	0.767132	0	0
5	5220	0.76685	0	0

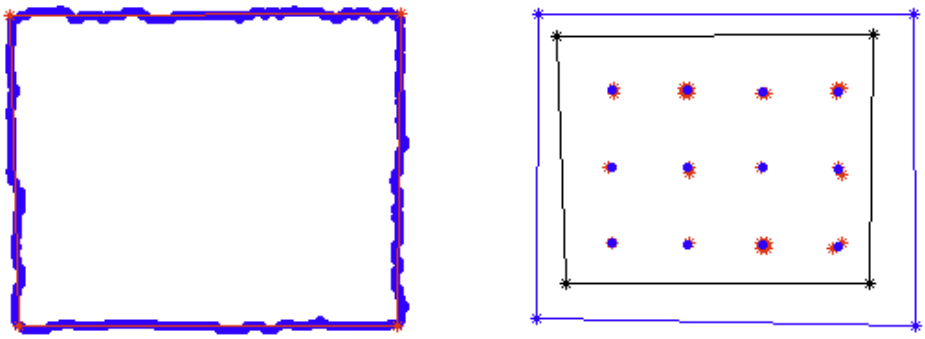
TABLE I: Depth apexes optimisation using genetic algorithm

One example of the genetic algorithm results for the above depth images is shown in Tab. I. The best average distance, 0.92 pixels, is found after five generations. The four apexes for both depth and RGB images are shown in Fig. 5a and 5b respectively. It demonstrated that the proposed algorithm is able to find the best key points based on the images captured. In addition, the numbers of edge points are not constant and the corners identified in the RGB image are more than 12 crossing points on the checkerboard, which may cause problems for the algorithms using the edge points/corners as the key points. The method in this paper uses four estimated apexes instead of the edge points/corners as the key points, and can find the optimised solution independent of the numbers of edge points or corners. Fig. 6a compares these estimated apexes in the RGB image. The difference between them will be used to determine the transformation matrix from the depth coordinates to RGB coordinates.

Then we transform the depth edge into RGB image coordinate system shown in red in Fig. 6b and the original depth edge is in blue. It is clear to see that using the red edge to segment the checkerboard is much better than the blue.

Hand gesture segmentation has been evaluated based on the proposed alignment method. Improvements have been achieved and segmentation results of hand gesture “five” are shown with the comparison between the official calibration and the proposed alignment algorithm in Fig. 7a and 7b, where the proposed algorithm corrects the alignment of the depth image with RGB image, and almost all the coloured fingers have been extracted.

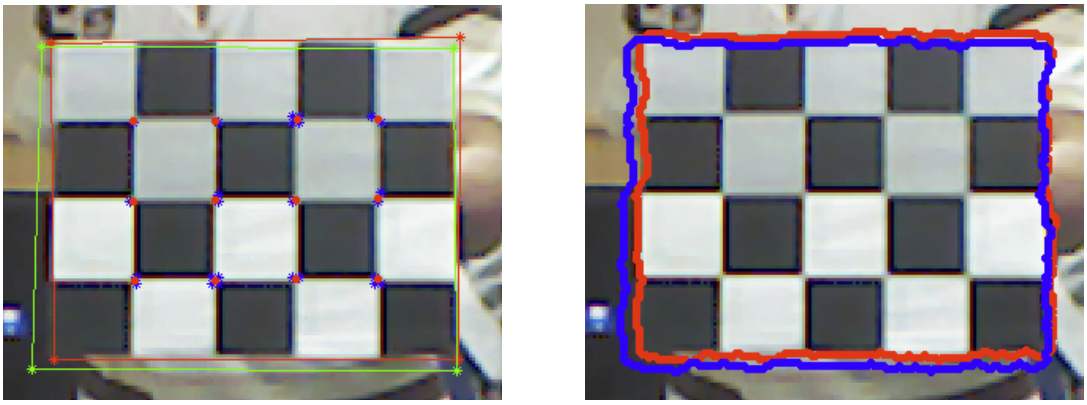
It can also be seen that though the segmented hand gesture shown in Fig. 7b is much better aligned than the one in Fig. 7a, there are still some mismatched pixels, some of which belonging to the background are selected as hand pixels and some of which being part of the hand are misplaced into the



(a) Four apexes (red stars) found in the depth image

(b) Four apexes (blue stars) found in the RGB image

Fig. 5: Solutions of the genetic algorithm



(a) Comparison between Apexes estimated in depth image (green) and RGB image (red)

(b) Comparison between transformed depth edge points (red) and original edge points (blue)

Fig. 6: Data comparison

background. To correct these mismatched pixels, the edge of the segmented hand gesture is further refined by the proposed EM algorithm and the results for the gesture "five" are given in Fig. 7. Fig. 7c shows the result of the EM algorithm with 2 iterations and Fig. 7d with 4 iterations. The refined hand gestures contains less mismatched points and are much cleaner than those in figures 7a and 7b. Results on five other hand gestures, (*i.e.* "one", "two", "three", "I love you" and "good luck") are shown in Fig. 8, where gestures in the first raw are the segmented hand gestures with official calibration, the ones in the second raw are results with only the proposed alignment method, and the third raw gives the final refined results by further applying the proposed EM algorithm on the aligned gestures in the second raw.

V. Concluding Remarks

In this paper, novel methods have been proposed to segment hand gestures in RGB-D data using the Kinect device. The proposed alignment method employs genetic algorithms to estimate the key points from both depth and RGB images, and it is robust to the uncertainties of the point numbers identified by using the common image processing tools such as corner and edge detectors. It is capable of correctly positioning the depth image with the RGB image. However, due to the noise and holes in the depth map, the segmented result using the depth information has lots of mismatched pixels, which need further adjustment. To solve this problem, a novel approach using EM algorithm has been proposed to further refine the edge of the segmented hand gesture. The experimental results show that the results by the proposed methods precisely segment the hand gestures and are much better than the official calibrated images and the results with only the proposed alignment method. The proposed methods in this paper will further provide a significant improvement to the performance of the hand gesture recognition using Kinect. Our future research will be on the efficiency improvement of the proposed methods to adapt to the real-time applications, and on hand gesture feature extraction and hand gesture recognition based on the newly aligned and segmented images [19], [26].

Appendix

(a) With the official calibration method (b) with the proposed alignment method (c) with the EM algorithm with 2 iterations (d) with the EM algorithm with 4 iterations

Fig. 7: Segmentation results of the gesture "five" using different methods

Algorithm 1 EM algorithm to segment hand gesture

Require: Fix R {R is the depth range used to segment hand gesture via only depth information.}.

$D^0 \leftarrow STF[27],[28]$ {Use Spatial-temporal filtering to track the hand and get the hand depth value}

$p(\tilde{H}|RGB_i, D_i, L_i) = 1/0$ {Use Sobel method to get the edge of the hand according to the R and D^0, and set the initial probability for each pixel}

repeat

  $\{p(H)^{new}, p(\tilde{H})^{new}\} \leftarrow Eq.6$ and 7 {Compute the new prior probabilities of the hand gesture and background using Eq. 6 and 7}

  $\{\mu^{new}_{RGBJ}, \mu^{new}_{RGBJ}\} \leftarrow Eq.$ 8{Compute the new RGB and Depth centres of the hand gesture and background using Eq. 8}

  $\{\Sigma^{new}_{RGBJ}, Sigma^{new}_{DH}\} \leftarrow Eq.$ 9 and 10 {Compute the RGB variance of the hand gesture and background using Eq. 9 and the Depth variance of the hand gesture using Eq. 10}

  $edge^{new} \leftarrow Eq.$ 11{Get the new edge using Eq. 11}

  $p^{new}(H|RGB, D, L) \leftarrow Eq.$ 1{Upgrade the probabilities using Eq. 1 }

  $\log(\mathscr{L}(\Theta|\mathscr{U})^{new}) \leftarrow Eq.$ 4{Compute the log-likelihood using Eq. 4}

until $\frac{\log(\mathscr{L}(\Theta|\mathscr{U})^{new}}{\log(\mathscr{L}(\Theta|\mathscr{U})^{old}} - 1 \leq threshold$ {Stop if the relative difference of the log-likelihood between two adjacent iterations is blow the preset threshold}

References

[1] K. Khoshelham, S. O. Elberink, Accuracy and resolution of kinect depth data for indoor mapping applications, Sensors 12 (2) (2012) 1437–1454.

[2] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, R. Moore, Real-time human pose recognition in parts from single depth images, Communications of the ACM 56 (1) (2013) 116–124.

[3] C. Herrera, J. Kannala, et al., Joint depth and color camera calibration with distortion correction, Pattern Analysis and Machine Intelligence, IEEE Transactions on 34 (10) (2012) 2058–2064.

[4] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, A. Fitzgibbon, Kinectfusion: Real-time dense surface mapping and tracking, in: Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on, IEEE, 2011, pp. 127–136.

[5] S. Izadi, R. A. Newcombe, D. Kim, O. Hilliges, D. Molyneaux, S. Hodges, P. Kohli, J. Shotton, A. J. Davison, A. Fitzgibbon, Kinectfusion: real-time dynamic 3d surface reconstruction and interaction, in: ACM SIGGRAPH 2011 Talks, ACM, 2011, p. 23.

[6] N. Silberman, R. Fergus, Indoor scene segmentation using a structured light sensor, in: Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, IEEE, 2011, pp. 601–608.

[7] J. Preis, M. Kessel, M. Werner, C. Linnhoff-Popien, Gait recognition with kinect, in: 1st International Workshop on Kinect in Pervasive Computing, 2012.

[8] J. L. Raheja, A. Chaudhary, K. Singal, Tracking of fingertips and centers of palm using kinect, in: Computational Intelligence, Modelling and Simulation (CIMSiM), 2011 Third International Conference on, IEEE, 2011, pp. 248–252.

[9] W. Xu, E. J. Lee, Gesture recognition based on 2d and 3d feature by using kinect device, in: International Conference on Infromation and Security Assurance, Vol. 6, 2012.

[10] J. Sturm, S. Magnenat, N. Engelhard, F. Pomerleau, F. Colas, W. Burgard, D. Cremers, R. Siegwart, Towards a benchmark for rgb-d slam evaluation, in: Proc. of the RGB-D Workshop on Advanced Reasoning with Depth Cameras at Robotics: Science and Systems Conf.(RSS), Los Angeles, USA, Vol. 2, 2011, p. 3.

[11] L. Bo, X. Ren, D. Fox, Unsupervised feature learning for rgb-d based object recognition, ISER, June.

[12] L. Spinello, K. O. Arras, People detection in rgb-d data, in: Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, IEEE, 2011, pp. 3838–3843.

[13] Z. Zhang, Microsoft kinect sensor and its effect, Multimedia, IEEE 19 (2) (2012) 4–10.

[14] L. Cruz, D. Lucio, L. Velho, Kinect and rgbd images: Challenges and applications, in: Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2012 25th SIBGRAPI Conference on, IEEE, 2012, pp. 36–49.

[15] Z. Zhang, Flexible camera calibration by viewing a plane from unknown orientations, in: Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, Vol. 1, Ieee, 1999, pp. 666–673.

[16] I. Oikonomidis, N. Kyriazis, A. Argyros, Efficient model-based 3d tracking of hand articulations using kinect, in: British Machine Vision Conference, 2011, pp. 101–1.

[17] Y. Li, Hand gesture recognition using kinect, in: Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on, IEEE, 2012, pp. 196–199.

[18] M. Van den Bergh, D. Carton, R. De Nijs, N. Mitsou, C. Landsiedel,

(a) “One” (b) “Two” (c) “Three” (d) “I love you” (e) “Good luck”

Fig. 8: Segmentation results of hand gestures “one”, “two”, “three”, “I live you” and “good luck” using different methods. Row one: segmented gestures with official calibration; row two: segmented gestures with the proposed alignment method; row three: segmented gestures with the proposed alignment and EM algorithm.

K. Kuehnlenz, D. Wollherr, L. Van Gool, M. Buss, Real-time 3d hand gesture interaction with a robot for understanding directions from humans, in: RO-MAN, 2011 IEEE, IEEE, 2011, pp. 357–362.

[19] Z. Ju, H. Liu, A Unified Fuzzy Framework for Human Hand Motion Recognition, IEEE Transactions on Fuzzy Systems 19 (5) (2011) 901–913.

[20] M. Tang, Recognizing hand gestures with microsofts kinect, Palo Alto: Department of Electrical Engineering of Stanford University:[sn].

[21] K. Khoshelham, Automated localization of a laser scanner in indoor environments using planar objects, in: Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on, IEEE, 2010, pp. 1–7.

[22] K. Khoshelham, Accuracy analysis of kinect depth data, in: ISPRS workshop laser scanning, Vol. 38, 2011, p. 1.

[23] J. Smisek, M. Jancosek, T. Pajdla, 3d with kinect, in: Consumer Depth Cameras for Computer Vision, Springer, 2013, pp. 3–25.

[24] Z. Ju, Y. Wang, C. S.Y., H. Liu, Image alignment for hand gesture segmentation using kinect, in: Proc. International Conference on Machine Learning and Cybernetics, Tianjing, China, 2013, pp. 1–8.

[25] J. A. Bilmes, et al., A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models, International Computer Science Institute 4 (510) (1998) 126.

[26] Z. Ju, H. Liu, Fuzzy gaussian mixture models, Pattern Recognition 45 (3) (2012) 1146–1158.

[27] P. Dollár, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, in: Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on, IEEE, 2005, pp. 65–72.

[28] H.-M. Zhu, C.-M. Pun, Hand gesture recognition with motion tracking on spatial-temporal filtering, in: Proceedings of the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry, ACM, 2011, pp. 273–278.

[29] H. Farid, E. P. Simoncelli, Optimally rotation-equivariant directional derivative kernels, in: Computer Analysis of Images and Patterns, Springer, 1997, pp. 207–214.