J Multimodal User Interfaces
DOI 10.1007/s12193-016-0215-x

ORIGINAL PAPER

Gesture recognition based on HMM-FNN model using a Kinect

Xiao-Li Guo[1] · Ting-Ting Yang[1]

Received: 28 July 2015 / Accepted: 16 March 2016

Abstract Addressing the problem of complex dynamic gesture recognition, this paper obtains the body depth image through the body feeling sensor device—Kinect; the threshold segmentation method is used to segment the gestures depth image, on the basis of the common distance between hand and body. Then, the HMM-FNN model, which combines the hidden markov model (HMM) and the fuzzy neural network (FNN), is used for dynamic gesture recognition. This paper mainly focuses on the trainees' common operations of equipment in virtual substation to set the custom gesture interaction sets. Based on the characteristic of the complex dynamic gesture, gesture image was decomposed into three feature sequences—hand shape change, hand position changes in the two-dimensional plane, and movement in the Z-axis direction, for feature extraction. The HMM model is respectively built according to the three sub sequences, and the FNN was connected to judge the semantics of gesture using the fuzzy reasoning. By experimental verification, the HMM-FNN model can quickly and effectively identify complicated dynamic hand gestures. Meanwhile, it has strong robustness. The recognition effect is superior to that of the simple HMM model.

Keywords Kinect · Threshold segmentation method · Complex dynamic gesture · HMM-FNN · Gesture recognition

✉ Ting-Ting Yang
1248192601@qq.com

Xiao-Li Guo
243589657@qq.com

1 Information Engineering College, Northeast Dianli University, Jilin 132012, China

1 Introduction

Due to the natural and intuitive characteristics of gesture, gesture interaction has become one of the least limited interaction patterns in human–computer interaction for users. Currently, vision-based gesture recognition is more widely applied in artificial intelligence, such as the motion sensing game, the virtual laser keyboard, and robot operation [1,2]. Motion sensor Kinect is a Microsoft XBox360 external motion device. It can simultaneously obtain RGB image and depth image, and has gradually become the focus of the study of vision-based gesture recognition. OpenNI—the open source software—defines and forms the standard API, to easily access and use Kinect. However, it only provides four kinds of simple gestures—Click, Wave, RaiseHand, and MovingHand [3], which could not meet the user demands for human–computer interaction using natural gestures. This requires the enrichment and expansion of its gesture set.

In recent years, gesture recognition based on computer vision has become a hot spot. In the paper [4], the k-curvature algorithm is employed to locate the fingertips over the contour, and dynamic time warping is used to select gesture candidates and also to recognize gestures by comparing an observed gesture with a series of prerecorded reference gestures. An average recognition rate of 92.4 % is achieved over 55 static and dynamic gestures. The paper [5] targets real-time recognition of both static hand-poses and dynamic hand-gestures in a unified open-source framework. While hand-pose recognition exploits techniques working on depth information using texture-based descriptors, gesture recognition evaluates hand trajectories in the depth stream using angular features and hidden Markov models (HMM). The accuracy and robustness of the recognition system have been evaluated using a publicly available database and across

Published online: 26 March 2016

many users. The paper [6] introduce the new 4-D color-depth (CoDe4D) LST feature that incorporates both intensity and depth information acquired from RGB-D cameras. The feature detector constructs a saliency map through applying independent filters in xyzt dimension to represent texture, shape and pose variations, and selects its local maxima as interest points. A complete activity recognition system was built by combining features with bag-of-features representations and support vector machines. The experiments using four benchmark color-depth human activity data sets, including UTK Action3-D, Berkeley MHAD, ACT42, and MSR daily activity 3-D data sets. Experimental results demonstrate the promising representative power of our CoDe4D features, which obtain the state-of-the-art performance on activity recognition from RGB-D visual data. The paper [7] presents an image-to-class dynamic time warping (I2C-DTW) approach for the recognition of both 3D static hand gestures and 3D hand trajectory gestures. Firstly, propose a technique to compute the I2C-DTW distance to obtain better generalization capability; then proposes fingerlets model for static gesture representation and strokelets model for trajectory gesture representation to compute the DTW distance between a data sample and a gesture category. The experiment results show that the proposed I2C-DTW approach significantly improves the recognition accuracy on both static gestures and trajectory gestures using UESTC-HTG gesture dataset. In the paper [8], dynamic time warping algorithm and an off-the-shelf software tool is employed for vocal language generation. The proposed method is capable of successfully detecting gestures stored in the dictionary with an accuracy of 91 %. The training process is simple; however, a large amount of calculation is required in its use. Furthermore the identification process is vulnerable to interference from other people's hands in the vicinity. The paper [9] proposed the static hand gestures recognition algorithm based on depth hands localisation and hog feature and the dynamic hand gesture recognition algorithm based on the improved HMMs.

In this paper, the complex gesture recognition model HMM-FNN combines the hidden Markov model (HMM) with the fuzzy neural network (FNN), which is built based on Kinect. The HMM-FNN model decomposes the complex dynamic gesture features into three dimensional feature sequences. HMM models were established according to three feature sequences, combined with FNN reasoning ability to recognize dynamic gestures [10]. The common dynamic operation gesture used by simulation personnel with substation equipment is adopted to train the model in this paper. It was found that this composite model can quickly and accurately identify complex dynamic gesture and can effectively reduce system complexity and reduce computational burden.

2 Depth image acquisition

Kinect's CMOS infrared sensor can sense the outside world and obtain depth information through the sense of infrared laser speckle, which can resist both environmental illumination and complex background [4]. Before Kinect 2.0 was utilized to directly obtain depth image, Kinect 2.0 had to be connected to a computer which conformed to its configuration requirements. The Kinect for Windows SDK 2.0, OpenCV, and OpenNI were respectively downloaded and installed. The hardware and software configuration processes were completed in a Microsoft Visual Studio 2013 environment.

The Kinect object was first initialized to obtain input stream data from the depth camera [11]; then, through ColorFrameReady and DepthFrameReady events trigger, it can obtain the depth of field data [12]; Fig. 1 shows the RGB image and the depth image. After obtaining the depth image, it is necessary to call upon the getMetaData() function to update the gesture data.

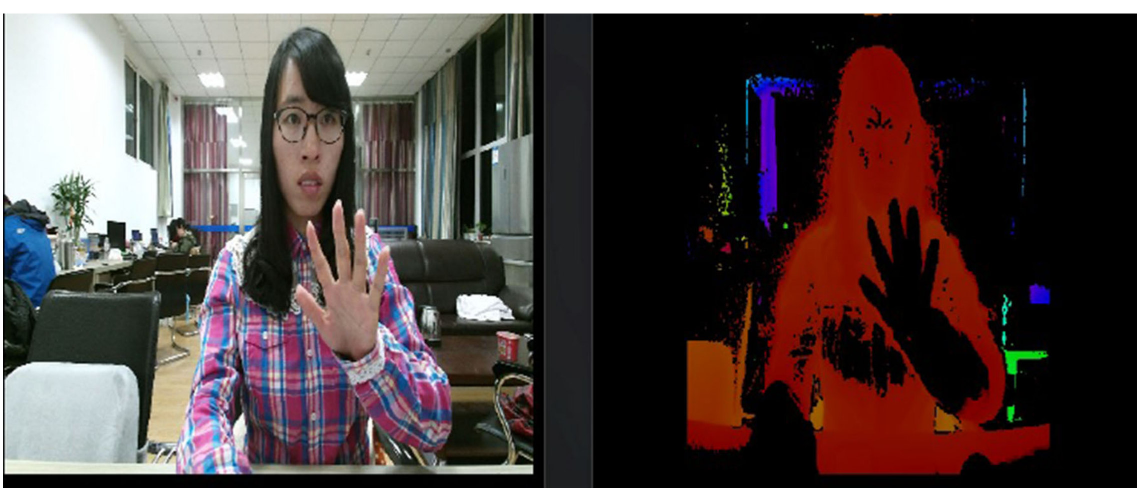

Fig. 1 The RGB image and depth image from Kinect

3 Gesture image acquisition and threshold segmentation

This paper mainly studies hand movements, so after obtaining depth image information, it is necessary to acquire and segment the hand image.

The depth information acquired from Kinect was stored in a two-dimensional array [3]. In the array each data object actually represents the distance from the body to the depth camera. Threshold segmentation is a quick method of extracting gestures from depth image, whose purpose is to make decisions in the image pixels, or directly eliminate some pixels which below or above a certain value. In this paper, the threshold value is the distance from the hand to the depth camera. Generally, when making gestures, hand and body are separated, the common distance between which is 10–20 cm. According to the common distance and experimental data setting the double threshold value, then the hand range of the depth map was separately segmented. First, the Nite module tracks the barycentric coordinates of the hand area, and then this paper uses the cvThreshold() function in OpenCV to implement thresholding [13]. The cvThreshold() function is given by comparing the size of each element in the array with the set threshold value—threshold_type—to perform operations, enabling the gestures depth map section to be segmented, as shown in Table 1.

T is the threshold parameter; M is the max-value; the max-value is the maximum value of CV_ THRESH_BINARY and CV_THRESH_BINARY_INV.

Table 1 Threshold_type values and the corresponding operation in cvThreshold()

Threshold-type values	Operation
CV_ THRESH_BINARY	$\mathrm{dis_i} = (\mathrm{src_i} > T)\ ?\ M : 0$
CV_THRESH_BINARY_INV	$\mathrm{dis_i} = (\mathrm{src_i} > T)\ ?\ 0 : M$
CV_THRESH_TRUNC	$\mathrm{dis_i} = (\mathrm{src_i} > T)\ ?\ M : \mathrm{src_i}$
CV_THRESH_TOZERO_INV	$\mathrm{dis_i} = (\mathrm{src_i} > T)\ ?\ 0 : \mathrm{src_i}$
CV_THRESH_TOZERO	$\mathrm{dis_i} = (\mathrm{src_i} > T)\ ?\ \mathrm{src_i} : 0$

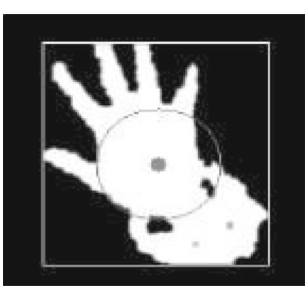

Fig. 2 Hand gesture segmentation image

Finally, the depth map of the gesture area and the background area were made using the binarization process to highlight the effect of the hand and to achieve gesture segmentation [14], as shown in Fig. 2.

4 Tracking and gesture recognition

This paper achieves a deep image of the hand when the threshold segmentation work has been completed. To achieve dynamic gesture recognition, gestures require fast and accurate dynamic tracking. After the OpenNI preliminary gesture was detected [4], we recorded the position of the center of gravity of the hand gestures to track the movement of the hand.

4.1 Dynamic gestures feature extraction

Dynamic gestures studied in this paper have three distinct characteristics—the variable time, variable space and variable integrity of gestures. Based on these, complex dynamic hand gesture features were decomposed into three-dimensional feature sequences—hand shape change, hand position changes in the two-dimensional plane, and movement in the Z-axis direction [7], for feature extraction, respectively.

In the segmented gesture depth image, the hand feature can be represented by the length of each finger and the distance between adjacent fingers. Looking to five fingers, we set the lengths from fingertip to the regional center of hand to be $l_0 \sim l_4$, and the distance between adjacent fingers to be $d_0 \sim d_4$.When sheltered between the fingers, the occluded finger length was set to 0, and the distances between the adjacent finger can be represented by the distances between the visible fingers. Therefore the basic features of the hand shape can be represented through $l_0 \sim l_4, d_0 \sim d_4$. In order to make hand features free from the effects of scale zooming, $l_0 \sim l_4$, $d_0 \sim d_4$ were divided by the current frame hand outsourcing circle diameter to obtain new features values for the standardization process [15]. In addition, different hands correspond to different value clusters. Accordingly, this paper set different base values corresponding to varied hand shapes, so that different hand eigenvalues were in different numerical range segments. In addition, the eccentricity ratio was introduced to be a part of the formula of the hand shape eigenvalues V. To a certain extent, the eccentricity ratio described the compactness of area. In this way, V not only can reflect the shape feature of the finger position in hand shape, but also can reflect the statistical characteristics of the whole hand area. Hand shape eigenvalues V may be determined according to formula (1):

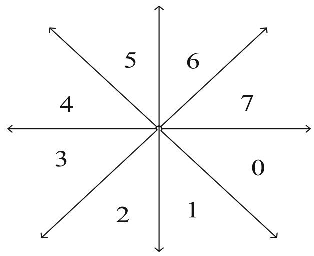

Fig. 3 The division of direction angle area

$$V = a \times [V_{\text{base}} + 100 \times \sum_{i=0}^{4} (l_i/d_h) + 100 \times \sum_{j=0}^{n} (d_j/d_h) + (100 - a) \times E. \quad (1)$$

In the Formula (1), a ($0 \le a \le 100$) is a proportionality constant; and n ($0 \le n \le 5$) is the number of visible fingers. The eccentricity ratio E can be calculated by Formula (2):

$$E = \sqrt{\frac{(A+B) - \sqrt{(A-B)^2 + 4H^2}}{(A+B) + \sqrt{(A-B)^2 + 4H^2}}}, A = \sum m_i (y_i^2 + z_i^2),$$
$$B = \sum m_i (z_i^2 + x_i^2), H = \sum m_i x_i y_i, \quad (2)$$

where A is rigid about the X-axis moment of inertia, B is rigid about the Y-axis moment of inertia, and H is product of inertia.

Two-dimensional movement of the whole hand takes place in two-dimensional space. The hand-shaped track feature values were represented by a set of discrete direction values of movement. The direction of movement region is divided as shown in Fig. 3. To describe the trajectory of gesture in a certain period of time, it was necessary to select any two adjacent points p_1, p_2 ($\| p_1 p_2 \| >$a set threshold) coordinates in the hand image [16]. Then the angle of $\overline{p_1 p_2}$ direction $\beta (0 \le \beta \le 2\pi)$ was calculated. According to the direction in which region β belongs, the discrete values of movement direction can be obtained. Thus we can get direction sequences of gesture trajectory for this period $R : r_1, r_2, \ldots, r_i, \ldots, r_7 (0 \le i \le 7)$.

For the Z-axis direction movement, the Z-axis direction was obtained by recording the distance from the hand regional center to the depth camera. Compared to the hand regional center position at the initial time, we obtained a set of consecutive normalized distance values.

4.2 HMM-FNN model

The hidden Markov model is a form of statistical analysis using a dual stochastic process combining the hidden Markov chain, which has a certain number of states, with a display random function set. HMM has a strong ability for sequential model building. It is a kind of dynamic information processing method based on sequential accumulative probability [17]. The FNN is a combination of fuzzy theory, and neural networks. Together, they have the advantage of achieving learning, association, identification and information processing [18]. However it is only applicable to discrete feature vectors, not to the timing sequence. According to the motion characteristics of complex dynamic gestures, the HMM-FNN model builds the appropriate timing model, and simultaneously combines the FNN model based on fuzzy rules modeling and reasoning ability, to become a complex structure model which can meet the demands of dynamic gesture recognition.

This paper adopts the HMM-FNN model, according to the type of feature extraction, with the HMM model used mainly for gesture modeling of three feature sequences. Both the hand shape change feature sequences and Z axis movement feature sequences use a one dimensional continuous HMM model to build models. One dimensional discrete HMM was adopted to build the whole hand motion models in a 2D plane. They were recorded as: posture HMM (PH), trajectory HMM (TH), and Z-axis HMM (ZH), respectively, corresponding to the HMM model of hand shape change, two-dimensional plane motion trajectory, and Z axis direction motion.

The Gaussian mixture model (GMM) uses 3–5 Gaussian models to represent the characteristics of each pixel in the image. After obtaining a new frame image, GMM updates, with each pixel in the image matching the current GMM, then it determines the background point and foreground point [16]. The observed state O likelihood under the GMM model is:

$$p\,(O/\lambda) = \sum_{i=1}^{M} \omega_i g_i\,(x)\,, \quad (3)$$

where ω_i is the weight of components i, $g_i(x)$ is a one-dimensional Gaussian density function, and $g_i(x) \sim N(\mu, \sigma^2)$. $\lambda = (A, B, n)$ were the model parameters of the HMM model. The HMM model can be described by five parameters, including the implicit state S, the observed state O, the initial state probability matrix n, the implicit state transition probability matrix A and the confusion matrix B. The continuous HMM model adopts GMM to determine the output probability density of observable characteristics.

4.3 Dynamic gesture recognition

Based on the previous realization of gesture tracking, this paper achieves recognition of dynamic gestures. The tracked dynamic gesture sequence is divided into three series, namely the plane trajectory point sequence, the hand shape change sequence, and the distance sequence from hand regional center to the depth camera. After quantifying the plane trajectory

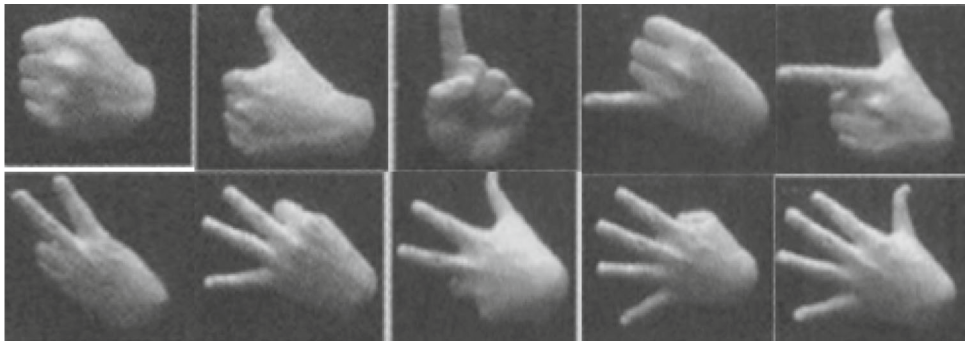

Fig. 4 Ten kinds of changes of hand shape from pos1 to pos10

point sequence in the direction of movement, the discrete orientation feature sequences were obtained and then passed on for the discrete HMM models to assess. The latter two sequences were input to continuous HMM models to assess. Upon the completion of inputting the feature sequence to the HMM model, the forward recursive method was used to calculate the likelihood $p(Q/\lambda)$of the HMM model. The $p(Q/\lambda)$ was set as the fuzzy membership to complete fuzzy pretreatment. Last, the sum–product was utilized to fuzzy reason and to identify the gesture semantics.

4.4 Model training

The training of the HMM-FNN model includes HMM parameter training and FNN connection weight training. Training HMM uses the forward-backward algorithm, which is a kind of expectation maximization algorithm based on the maximum-likelihood criterion, in which, through iterations, model parameters are re-estimated to get the optimum model parameter values in the sense of maximum likelihood.

When training the FNN model, the likelihood probability $p(Q/\lambda)$ of the HMM model was set to be the membership of input sample for this category. Studying the FNN model parameters mainly involved the weight of the antecedent of each fuzzy rule and the contribution of each fuzzy rule. An error back-propagation algorithm was utilized in training the FNN model parameters. It is a kind of typical learning rule for multi-layer feed forward neural networks without feedback. The idea is to adjust the network weights and thresholds, using back-propagation to keep the error of function value to a minimum [19]. The studying error function of the network model J_e was set to be $J_e = \frac{1}{2}(t-z)^T(t-z)$, in which vector t indicated the expected output of the network and vector z indicated the actual output of the network. The weight-correction value of the antecedent of each fuzzy rule and the contribution of each fuzzy rule can be calculated by the BP algorithm, then the weight updated were iterated according to the method of $w_{T+1} = w_T + \Delta w$ (w_T is the weight in time T). Until the parameters of the network study achieved convergence or reached the maximum number of iterations, the training process ended.

5 Experiment and result analysis

This study adopts Kinect for Windows 2.0. The hardware requirements of Kinect2.0 is as follows: a 64-bit processor, a 2.5 GHz i7 processor or faster processor, a 4 GB RAM, DX11 graphics adapter, and USB 3; the OS is: win8, win8.1. This experiment used a PC, with the following configuration: an Intel Core i7 2.6 GHz, 64-bit processor, 4 GB RAM, a graphics card, an NAIDIA GeForce GT 720M, and a built-in USB3.0 bus. The OS is win8.1. In the Microsoft Visual Studio 2013 compiler environment, we used VC++ programming to achieve recognition.

First, this paper set ten static gestures for the tracking process, including one kind of hand clenched gesture, three kinds of one finger stretched out gestures, two kinds of two fingers stretched out gestures, two kinds of three fingers stretched out gestures, one kind of four fingers stretched out gesture and one kind of five fingers stretched out gesture, marked as pos1 $\sim$ pos10, respectively, as shown as Fig. 4.

The changes of dynamic hand gestures were pos1 to pos2, pos2 to pos10 and so on, setting the number of state nodes to be five in the PH models. Eight kinds of gestures in a two-dimensional plane were designed, which were up, down, left, right, clockwise, counterclockwise rotation, N-shaped movement and Z-shaped motion, as shown as Fig. 5, with the states of the node in the TH models setting to be 4. In GMM, the number of Gauss nodes was also set to 4. Push, pull, pull then push, and push then pull four kinds of movement were set in the direction of Z-axis, with the states of the node in the ZH models setting to be 3.

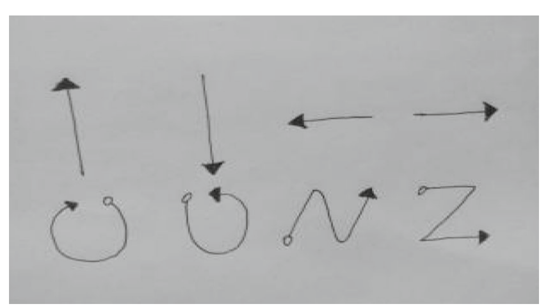

Fig. 5 Eight kinds of gestures in a two-dimensional plane

Table 2 The custom set of interaction gestures

Label	Semantics	Gestures
1	Select	Stretch out one finger to draw a circle
2	Move	Stretch out thumb and index finger, and move the whole hand
3	Rotate	The distance between the two fingers is about 50–100 cm
4	Scale	Stretch out middle finger, ring finger and little finger, the fingertip
5	Slide	Of which compose the palm plane, then rotate the palm
6	Click	Stretch out four finger, and $l_0 \sim l_4$ all decrease or magnify
7	Free	Stretch out five finger, and wave
		Stretch out five finger and push hand
		Make a fist

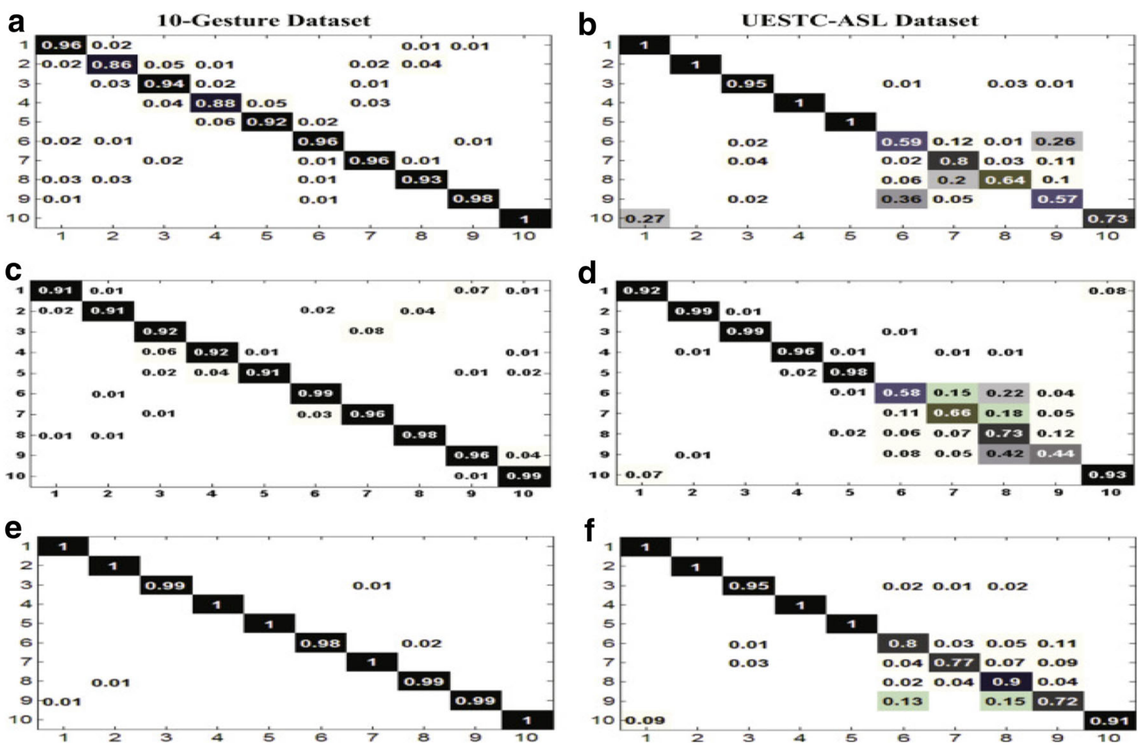

Fig. 6 The confusion matrix on ten-gesture dataset and UESTC-ASL dataset: **a**, **b** DTW; **c**, **d** HMM; **e**, **f** HMM-FNN

Based on the common operation of virtual substation equipment—select, move, zoom, rotate, this paper designed the corresponding interact gestures, as shown in Table 2.

Each kind of gestures was done 15 times by 10 experimenter, then there are 150 groups of data for each kind of gesture. We randomly selected 100 groups to be the training data, the remaining 50 groups of data comprising of test data. First, the model of PH, TH, and ZH were trained, in which the number of state nodes was set based the complexity of the gesture. In the GMM, the number of Gauss state was same to that of the HMM model state node. Then the error back propagation algorithm was utilized to train the connection weights to FNN.

We can adjust the connection weights to improve recognition effect, when training the model of FNN. For example, the motion of hand gestures 7 is only related to hand shape change, therefore, we can eliminate the connection to the TH node when building the HMM model, and can set the weight of hand shape change to be slightly larger than that of the Z-axis direction when setting the weights of the rule conditions.

In this experiment, we compared the proposed HMM-FNN model with both the DTW approach and the HMM approach on the ten-gesture dataset and UESTC-ASL dataset, shown in Fig. 6a, c, e shows the three confusion matrices of DTW, HMM and HMM-FNN, respectively, and we can see the recognizing rate of the HMM-FNN approach is 99.5 %.

Analyzing the experimental results, it indicates that the recognition rate of HMM-FNN is much higher than that of HMM and that of DTW. The results shows that although the training process of the HMM-FNN model is more complicated, it can effectively and rapidly recognize the complex dynamic gesture, by connecting to the FNN which has the fuzzy reasoning capacity. Meanwhile, Kinect2.0 can distinguish one more gesture recognition, and can set the nearest gesture to be the main gesture. The identification process is not affected by other people's gesture, and the ability of anti-interference is strong.

6 Conclusions

This paper mainly studied on the dynamic gesture recognition based on Kinect, and built the HMM-FNN model. Through the experiment identification, this study can accuracy and rapidly recognize a set of dynamic gesture, and enrich the gesture set of Kinect. Compared to the HMM model, the HMM-FNN model adequately take the space and fuzzy feature into consideration. Towards to the space feature, the HMM-FNN model decomposed the complex dynamic hand gesture features into three-dimensional feature sequences—hand shape change, hand position changes in the two-dimensional plane, and movement in the Z-axis direction, for respective recognition, which reduce the complication of the dynamic gesture recognition. Towards to the fuzzy feature, the HMM-FNN model combined the FNN model to fuzzy reasoning, which strengthens the robustness of the system.

The next step, we plan to apply this method of gesture recognition to the substation visualization training platform [20]. Combined virtual reality engine unity, the trainees dynamic operating gesture is as the input information of the substation visualization training platform, in order to use the natural habit of action to complete the operation of dismantling, installation, maintenance, and so on, to achieve human–computer interactive learning, and to strengthen training skills for security operation.

Compliance with ethical standards

Funding This study was funded by the key transformation project of provincial science and technology plan (No. 20140307008GX) and the "Double ten" cultivation project of Jilin provincial education department (吉教科合字 [2014] No. 109).

Conflict of interest The authors declare that they have no conflict of interest.

Research involving human participants and/or animals In this research, ten people were chose to participate in the gesture recognition experiment. They are Tingting Yang, Yanli Wen, Liqing Sun, Chunlei Shi, Xudong Ma, Jun Qi, Qing Li, Yang Yu, Jiajia Zhang, Ning Zhou.

Informed consent All participants voluntarily agreed to participate in this study and all gave written informed consent.

References

1. Wang Y, Zhang Q-Z (2013) Gesture recognition based on Kinect depth information. J Beijing Inform Sci Technol Univ 28(1):22–26
2. Tomoya M, Jungpil S, Pankoo K (2014) Hand gesture and character recognition based on Kinect sensor. Int J Distrib Sens Netw 2014:1–6. doi:10.1155/2014/278460
3. Chen Y, Chen Z, Zhou X (2013) Gesture recognition based on Kinect and application in the virtual assembly technology. Electron Des Eng 21(10):3–7
4. Plouffe G, Cretu AM (2016) Static and dynamic hand gesture recognition in depth data using dynamic time warping. IEEE Trans Instrum Meas 65(2):305–316
5. Pedersoli F, Benini S, Adami N, Leonardi R (2014) An open source framework for hand pose and gesture recognition using Kinect. Vis Comput 30(10):1107–1122
6. Zhang H, Parker Lynne E (2016) CoDe4D: color-depth local spatio-temporal features for human activity recognition from RGB-D videos. IEEE Trans Circuits Syst Video Technol 26(3):541–555
7. Cheng H, Dai Z, Liu Z (2016) An image-to-class dynamic time warping approach for both 3D static and trajectory hand gesture recognition. doi:10.1016/j.patcog.2016.01.011
8. Halim Z, Abbas G (2015) A Kinect-based sign language hand gesture recognition system for hearing- and speech-impaired: a pilot study of Pakistani sign language. Assist Technol 27(1):34–43
9. Wu X, Yang C, Feng Q (2015) Research on Kinect-based hand gesture recognition algorithm and its applications. Comput Appl Softw 32(7):173–177
10. Wang X, Dai G, Zhang X et al (2008) Recognition of complex dynamic gesture based on HMM-FNN model. J Softw 18(9):2302–2312
11. Yangecnu (2012) Kinect for Windows SDK 开发入门（一）：开发环境配置. http://www.cnblogs.com/yangecnu/archive/2012/03/30/KinectSDK_Geting_Started.html. Accessed 7 Apr 2015
12. Yangecnu (2012) Kinect for Windows SDK 开发入门（四）：景深数据数据处理（上）. http://www.cnblogs.com/yangecnu/archive/2012/04/04/KinectSDK_Depth_Image_Processing_Part1.html. Accessed 7 Apr 2015
13. Lei J, Wang W (2013) Research and implementation of image threshold segmentation based on OpenCV. Mod Electron Tech 36(24):72–76
14. Cao C, Li R, Zhao L (2012) Hand posture recognition method based on depth image technology. Comput Eng 38(8):16–18
15. Chong W, Zhong L, Shing-Chow C (2015) Superpixel-based hand gesture recognition with Kinect depth camera. IEEE Trans Multimed 1(17):29–39
16. Wang D, Shi C, Zhang M (2012) Multi-touch gesture recognition based on propagation neural networks. Pattern Recogn Artif Intell 23(3):408–413
17. Lin C, Hu J, Jie X (2012) Survey on models and evaluation of quality of experience. Chin J Comput 35(1):1–15
18. Tan T, Ren K, Chen X et al (2011) Fuzzy neural network technology. J Chongqing Univ Arts Sci (Nat Sci Ed) 30(1):71–74
19. Zhu Q, Li K, Zhang Z et al (2010) An improved Gaussian mixture model for an adaptive background model. J Harb Eng Univ 31(10):1348–1353, 1392
20. Qu Z, Hou S, Zhang Y et al (2014) Realization of substation visualization training platform. J Northeast Dianli Univ 34(3):75–79