A Comprehensive Evaluation of Moving Static Gesture Recognition with Convolutional Networks

Xiyuan Wang, Zhang Chen
Graduate School at Shenzhen
Tsinghua University
Shenzhen, China
e-mail: wxyzzycry@sina.com, z.then.thu@gmail.com

Xueqian Wang*, Qianchuan Zhao, Bin Liang
Graduate School at Shenzhen
Tsinghua University
Shenzhen, China
e-mail: Wang.xq@sz.tinghua.edu.cn,
bliang@tsinghua.edu.cn

***Abstract*—Static gestures is of great value for intelligent interaction. Most current researches on this field do not consider the complexity and randomicity of its practical applications. In this paper, we make a comprehensive study on moving static gesture recognition which is closer to the reality with Convolutional Neural Networks (CNN). First of all, Leap Motion (LM) is used as an acquisition equipment. Then we investigate the performance of networks in different depth and architectures. We design entirely new experimental scenes namely static gesture in motion. Finally, our proposed method is tested on a self-built database containing 12 different gestures and compared to built-in algorithm of LM in different recognition areas. Experimental results show CNN is robust and effective for static gesture recognition in the complex environment.**

Keyword-static gesture recognition; Convolutional Neural Networks (CNN); leap motion; complex environment

I. INTRODUCTION

Human–Computer Interaction (HCI) mode has been widely used such as robot manipulation, medical surgery and so on [1-2]. In recent years, due to the advantages of convenience and intuition, gesture recognition and gesture-based interaction, as an area of HCI, have attracted an increasing interest in many studies [3].

Gesture-based interaction is aided through two major methods namely contact-based [4] and vision-based; the latter is widely studied for its convenience. As the popularity of low-cost but high-precision consumer depth cameras, like Microsoft's Kinect and Leap Motion [5], gesture information is frequently captured by these devices for gesture-based interaction. LM has been applied to robot control [6-8] for its high tracking precision. Unlike mechanical motion, hand motion influenced by many factors (distraction, fatigue, etc.) is always unstable and gestures could be auxiliary instructions to offset this defect; for instance, making a fist denotes stopping sending signals even though the hand is moving. Previous researches studied the static or dynamic gesture recognition, but up to now, there are no literatures that explored the recognition of static gesture in motion which can be seen as a subset of dynamic gestures. Therefore, this paper presents the first attempt to investigate moving-gesture recognition based on CNN and make a comparison of performance between CNN and built-in algorithm of LM.

The paper is organized in the following way: Previous related work in the field of gesture recognition is introduced in Section 2. Details about our proposed method is described in Section 3. The experiments and the results are shown in Section 4. Finally, Section 5 draws the conclusions.

II. RELATED WORK

Hidden Markov Models (HMMs) and Support Vector Machines (SVMs) are commonly used for vision-based gesture recognition. Elmezain et al. [9] proposed a system to recognize the alphabets and numbers in real time from color image sequences based on HMMs. Keskin et al. [10] used artificial neural network (ANN) and SVM to recognize American Sign Language digit achieved by Kinect. Similarly, extracting data from LM, Mendes et al. [11] used ANN to recognize static gestures, HMM to dynamic gestures and SVM to both types of gestures. Marin et al. [12-13] extracted a set of features about the depth images from Kinect and LM, then fed them into a multi-class SVM classifier to recognize the gestures. The above methods need to take lots of time to manually design new features.

In the field of image recognition, researchers have obtained significant progress with CNNs. Instead of feeding hand-designed features, the network uses almost raw images as its input and precisely makes prediction even in the case that the images have some degree of shift, scale, and distortion [14]. As raw images can be directly achieved from LM, it has been used as an acquisition device in many studies. In [15], Hu et al. used CNN to recognize trajectory images created from raw images of LM. McCartney et al [16] converted 3D gesture paths from LM to 2D image projections and then used CNN to classified gestures. In [17], Nasr-Esfahani et al. got hand picture from LM and applied CNN to hand gesture recognition in Operating Rooms. Molchanov et al. [18] developed a robust system combining LM with a short-range radar and a color camera for dynamic car-driver hand-gesture recognition. As the hand is fixed in a suitable position of depth cameras, they all adopted simple networks to recognize images which were resized to a small size without worrying about loss of effective pixels. However, in practical application of HCI, gesture-control

978-1-5386-7226-6/18/$31.00 ©2018 IEEE

plays an important part and a CNN structure must be carefully chosen to have a satisfactory performance in a more unstable application environment.

III. Proposed Method

A. The Proposed System

The Leap Motion shown in Fig. 1(a) is commonly used as an acquisition equipment in gesture-based interaction. It has a field of view of about 150 degrees and the effective range extends from approximately 25 to 600 millimeters. It gene-rates hand data set at 100-120 frames per second (fps).

As the Fig. 1(b) shows, the gesture image is captured by LM. Then the image is processed and passed to a trained CNN model. Finally, the model outputs the classification result.

B. Image Preprocessing

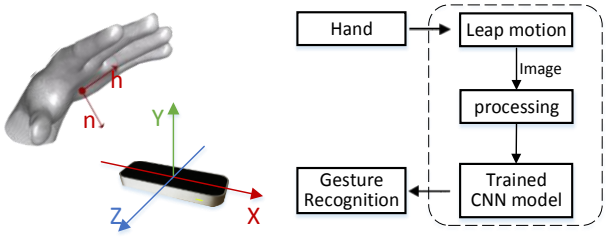
The raw image is first converted to binary image. Then, a 240*240 patch is extracted through a sliding window from the raw image with size of 640*240 to make the recognition target located in the center of the image. We did not resize the patch to smaller pixels, considering that a smaller size may cause the loss of valid pixels when the hand is close to the edge of the effective range.

C. Model

TABLE I. CNN Configuration

CNN Configuration			
A (7 layers)	B (8 layers)	C (9 layers)	D (12 layers)
Input (240×240 binary image)			
Conv3_1(32)	Conv3_1(32) Conv3_1(32)	Conv3_1(32) Conv3_1(32)	Conv5_2(64) cccp1,cccp2
Maxpool			
Conv3_1(64)	Conv3_1(64)	Conv3_1(64) Conv3_1(64)	Conv5_2(96) cccp3,cccp4
Maxpool			
Conv3_1(96)	Conv3_1(96)	Conv3_1(96)	Conv3_1(64) cccp5,cccp6
Maxpool			
Conv3_1(64)	Conv3_1(64)	Conv3_1(64)	Conv3_1(32) cccp7,cccp8
Maxpool			Global Average Pooling
Conv3_1(32)	Conv3_1(32)	Conv3_1(32)	
Maxpool			
Fc64			
Fc32			
Softmax12			

Table I shows 4 networks. Their forward propagation time are less than 100ms (intel i7-6700). The first three differ in the network depth, and the fourth adopts NIN structure [19]. All convolutional layers are convolved with 3×3 filters using a stride of 1 pixel. A stack of small-size filters (two 3*3 filters are identical to a single 5*5) can improve the network performance and significantly reduce the computational cost [20]. The convolutional layer parameters are denoted as "conv< receptive field size>_<stride>(number of channels)" and cccp refers to "cascaded cross channel parameteric pooling" structure.



(a) (b)

Figure 1. (a) LM system. (b) Recognition system.

Except for the softmax layer, all neurons use rectified linear unit (ReLU) activation function:

$$f(x) = \max(0, x) \tag{1}$$

The output of the softmax layers is computed as:

$$P_i = \frac{e^{V_i}}{\sum_j e^{V_j}} \tag{2}$$

where V_i is the output of the neuron i.

D. Training

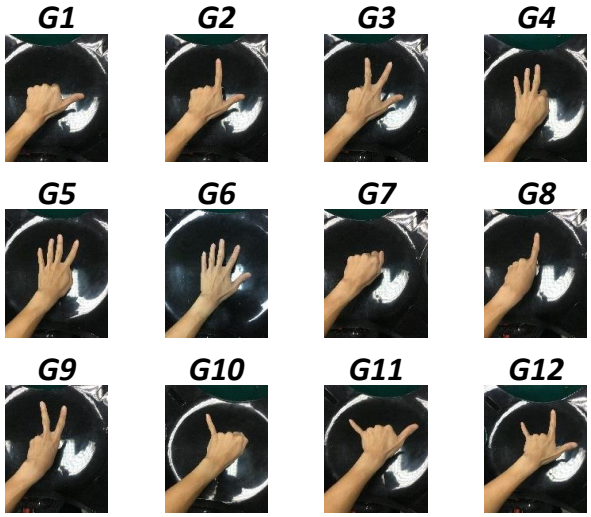


Figure 2. Gestures from G1 to G12

The database (shown in Fig. 2) collected from LM contains 12 different gestures created by left hand. The set totally has 144000 examples and each type has 12000 examples performed by 10 different people. We randomly take 120000 samples for training and the rest for validation.

The training is conducted by optimizing cross-entropy cost function using mini-batch gradient descent with Adam [21]. The batch size is set to 40 and the learning rate is set to 10^{-3}. Dropout technique [22] is applied in the first two fully-connected layers (dropout ratio is set to 0.5). As ReLU

is the activation function, the weights of all layers are initialized with He initialization [23]. Training stops when the cost function does not improve. Table II shows the final result.

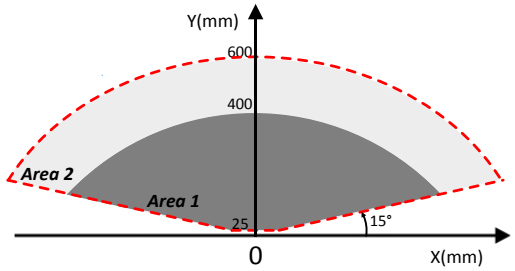
TABLE II. VALIDATION ERROR

Model	A	B	C	D
Validation error	13.48%	4.03%	0.57%	3.99%

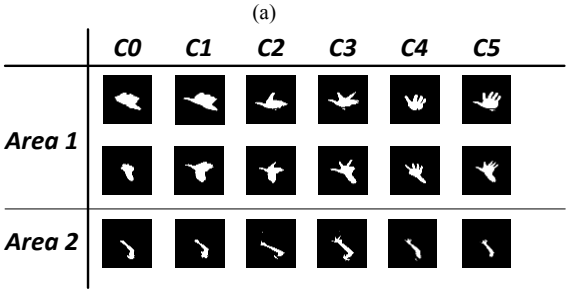
Model C has the lowest error rate and will be applied in the following experiments.

IV. EXPERIMENTS

When using LM or other deep cameras, the gesture information reduces as the hand gets farther, so it is necessary to consider performance of CNN in different regions shown in Fig. 3.



(a)



(b)

Figure 3. (a) Effective range of LM. (b) Images in different areas.

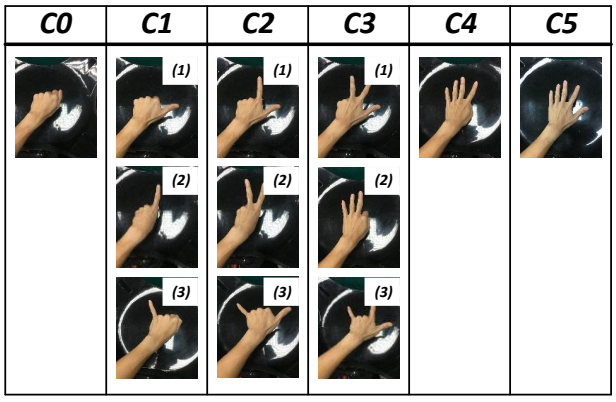


Figure 4. Gesture categories.

So far, there are no experiments on static gesture that are conducted under our scenes. Therefore, we only make a comparison between our method and algorithm of LM which is used as a benchmark. LM has no actual function of static gesture recognition but its data of fingers number (0-5) can be used to match our gestures (Fig. 4). C0-C5 represent the number of fingers from 0 to 5 and C1-C3 all have 3 types.

A. Static Gesture in Motion

We respectively collected 2000 examples of each type in two areas in the way mentioned-above during hand moving.

TABLE III. TESTING ERROR OF EXPERIMENT A

Gesture	**Area 1** (error/2000)		**Area 2** (error/2000)	
	CNN error	Built-in error	CNN error	Built-in error
C0	0	***0***	81	***0***
C1(1)	***1***	26	***8***	49
C1(2)	***15***	42	***6***	23
C1(3)	42	***11***	***508***	574
C2(1)	17	***9***	20	***0***
C2(2)	***26***	148	***2***	531
C2(3)	***1***	24	325	***220***
C3(1)	***20***	166	***130***	152
C3(2)	***28***	74	***17***	1261
C3(3)	3	***0***	***66***	98
C4	***9***	31	***171***	406
C5	***14***	49	***0***	40
Avg.	***15***/2000	48/2000	***111***/2000	280/2000
Acc.	***99.3%***	97.6%	***94.5%***	86.0%

×The *italic* represents better.

Table III shows the error of recognition of each type in two areas. Average error rate of our method for every 2000 examples is lower than the algorithm of LM which is used as a benchmark. In area 1, our method in 8 types of gestures is better than the benchmark, and the error gaps between the two methods is from 22 to 146. In the contrary, the error gaps in poor performance of 4 gestures of our method is from 3 to 31. The above situation is more obvious in area 2. In area 2, the algorithm of LM in recognizing gesture C2(2) and C3(2) is highly unstable, but our method conquers this shortage. As C0 and C2(1) have distinct features, our method does not show an advantage when the hand is far from LM.

B. Changing during Moving

The gesture regularly changed 11 times (Fig. 5) during moving in the effective range. Each gesture has 3000 and totally 36000 examples were collected.

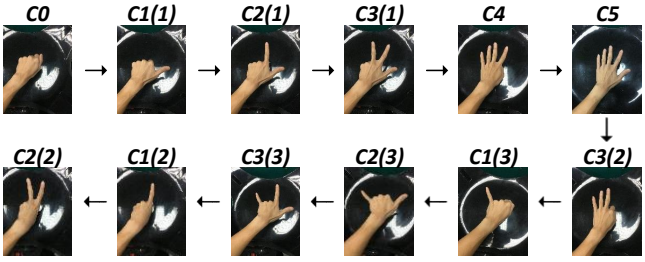


Figure 5. Gesture regularly changes.

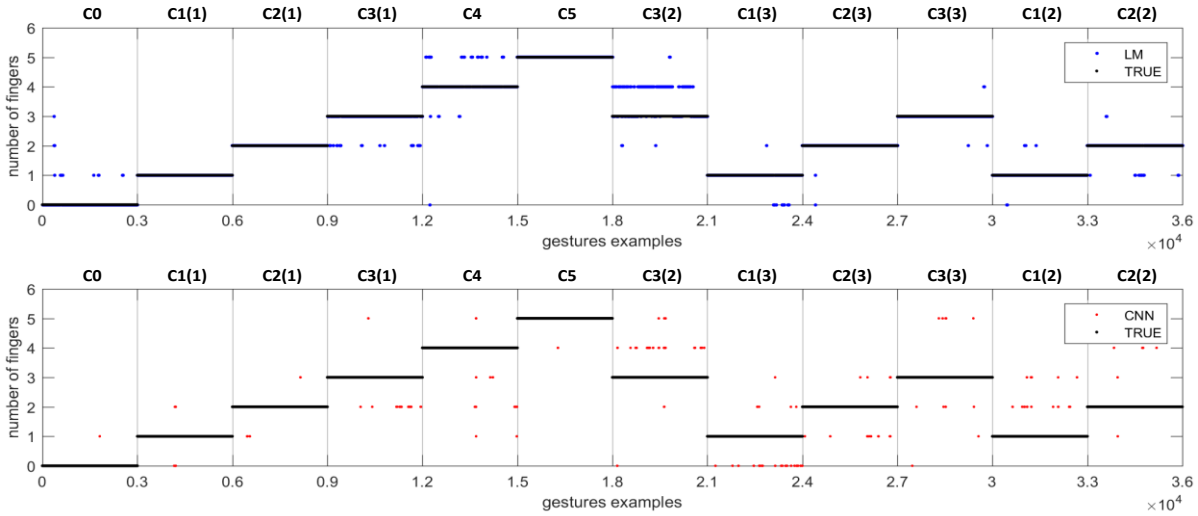


Figure 6. Error distribution. The black dots represent the true value of gesture and blue dots and red dots respectively represent recognition error of two methods. For the convenience of comparison, the value of sample (misclassified by CNN) whose finger value is same with the true type is set to 5 (for example, gesture C2(1) is misclassified to C2(2) so its value is set to 5).

TABLE IV. TESTING ERROR OF EXPERIMENT B

Gesture	ALL areas (error/3000)	
	CNN	Built-in
C0	***1***	70
C1(1)	25	***0***
C1(2)	53	***48***
C1(3)	101	***98***
C2(1)	25	***1***
C2(2)	***30***	101
C2(3)	53	***3***
C3(1)	***67***	261
C3(2)	***151***	1647
C3(3)	***22***	23
C4	***47***	246
C5	10	***0***
Avg.	***49/3000***	208/3000
Acc.	***98.4%***	93.1%

*The *italic* represents better.

As shown in Table IV, the overall performance is similar to scenario 1. Average errors of our method are significantly lower than the algorithm of LM. Because the range of motion is broadened to the whole recognition area, there are only 6 gestures not better than the benchmark, but error gaps between two methods still show advantages of our method. Fig. 6 visualizes the error distribution. It shows that our method effectively relieves the value jumping of the reference algorithm in C4 and C3(2) and a lag vanishes when gesture changes (e.g. C2(1) to C3(1)). The error of CNN is more evenly distributed and the whole recognition process is more stable. We also find that the position of errors of two methods is close, probably because they are both based on images.

V. CONCLUSION

In this paper, we make a comprehensive research on moving static gesture recognition with CNN. 4 different networks are evaluated during the design, with the consideration of speed and performance. According to the characteristics of the deep camera, we designed a novel and rigorous experimental scenario and made a comparative experiment. The experimental results show that CNN is robust and effective for static gesture recognition in an unstable environment. In the future, we will explore a method that is characterized by a higher accuracy and real-time performance.

ACKNOWLEDGMENT

The work presented in this paper was supported by the National Natural Science Foundation of China (Grant No. 61673239, 61703228), Science and Technology Research Foundation of Shenzhen (with number JCYJ20160428-182227081).

REFERENCES

[1] Dix, Alan. "Human-computer interaction, " Encyclopedia of database systems. Springer US, 2009. 1327-1331.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] Taylor, Russell H, "Medical Robotics and Computer-Integrated Surgery." IEEE International Computer Software and Applications Conference IEEE, 2008:1-1.

[3] Rautaray, Siddharth S., and Anupam Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey." Artificial Intelligence Review 43.1 (2015): 1-54.

[4] Kim, Ji-Hwan, Nguyen Duc Thang, and Tae-Seong Kim, "3-d hand motion tracking and gesture recognition using a data glove." Industrial Electronics, 2009. ISIE 2009. IEEE International Symposium on. IEEE, 2009.

[5] Leap Motion Controller. Available online: https://www.leapmotion.com (accessed on 10 February 2018).

[6] D Bassily, C Georgoulas, J Guettler, T Linner, T Bock, "Intuitive and adaptive robotic arm manipulation using the leap motion controller." ISR/robotik 2014; 41st international symposium on robotics; proceedings of. VDE, 2014.

[7] Kruusamäe, Karl, and Mitch Pryor, "High-precision telerobot with human-centered variable perspective and scalable gestural interface." Human System Interactions (HSI), 2016 9th International Conference on. IEEE, 2016.

[8] Du, Guanglong, and Ping Zhang, "A markerless human–robot interface using particle filter and Kalman filter for dual robots." IEEE Transactions on Industrial Electronics 62.4 (2015): 2257-2264.

[9] Elmezain, Mahmoud, Ayoub Al-Hamadi, and Bernd Michaelis, "Real-time capable system for hand gesture recognition using hidden markov models in stereo color image sequences." (2008).

[10] C Keskin, F Kıraç, YE Kara, L Akarun, "Real time hand pose estimation using depth sensors." Consumer depth cameras for computer vision. Springer, London, 2013. 119-137.

[11] N Mendes, P Neto, M Safeea, AP Moreira, "Online Robot Teleoperation Using Human Hand Gestures: A Case Study for Assembly Operation." Robot 2015: Second Iberian Robotics Conference. Springer, Cham, 2016.

[12] Marin, Giulio, Fabio Dominio, and Pietro Zanuttigh, "Hand gesture recognition with leap motion and kinect devices." Image Processing (ICIP), 2014 IEEE International Conference on. IEEE, 2014.

[13] Marin, Giulio, Fabio Dominio, and Pietro Zanuttigh, "Hand gesture recognition with jointly calibrated leap motion and depth sensor." Multimedia Tools and Applications 75.22 (2016): 14991-15015.

[14] LeCun, Yann, and Yoshua Bengio, "Convolutional networks for images, speech, and time series." The handbook of brain theory and neural networks 3361.10 (1995): 1995.

[15] Hu, Ji-Ting, Chun-Xiao Fan, and Yue Ming, "Trajectory image based dynamic gesture recognition with convolutional neural networks." Control, Automation and Systems (ICCAS), 2015 15th International Conference on. IEEE, 2015.

[16] McCartney, Robert, Jie Yuan, and Hans-Peter Bischof, "Gesture recognition with the leap motion controller." Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2015.

[17] Nasr-Esfahani, Ebrahim, et al. "Hand Gesture Recognition for Contactless Device Control in Operating Rooms." arXiv preprint arXiv:1611.04138 (2016).

[18] P Molchanov, S Gupta, K Kim, K Pulli, "Multi-sensor system for driver's hand-gesture recognition." Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on. Vol. 1. IEEE, 2015.

[19] Lin, Min, Q. Chen, and S. Yan, "Network In Network." Computer Science (2013).

[20] Simonyan, Karen, and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition." Computer Science (2014).

[21] Kingma, Diederik P, and J. Ba, "Adam: A Method for Stochastic Optimization." Computer Science (2014).

[22] GE Hinton, N Srivastava, A Krizhevsky, I Sutskever, RR Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors." Computer Science 3.4(2012):págs. 212-223.

[23] K He, X Zhang, S Ren, J Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification." Proceedings of the IEEE international conference on computer vision. 2015.