

A Pointing Gesture Based Egocentric Interaction System: Dataset, Approach and Application

Yichao Huang, Xiaorui Liu, Xin Zhang* and Lianwen Jin*
School of Electronic and Information Engineering
South China University of Technology
Guangzhou, P. R. China

eexinzhang@scut.edu.cn; eelwjin@scut.edu.cn

Abstract

With the heated trend of augmented reality (AR) and popularity of smart head-mounted devices, the development of natural human device interaction is important, especially the hand gesture based interaction. This paper presents a solution for the point gesture based interaction in the egocentric vision and its application. Firstly, a dataset named EgoFinger is established focusing on the pointing gesture for the egocentric vision. We discuss the dataset collection detail and as well the comprehensive analysis of this dataset, including background and foreground color distribution, hand occurrence likelihood, scale and pointing angle distribution of hand and finger, and the manual labeling error analysis. The analysis shows that the dataset covers substantial data samples in various environments and dynamic hand shapes. Furthermore, we propose a two-stage Faster R-CNN based hand detection and dual-target fingertip detection framework. Comparing with state-of-art tracking and detection algorithm, it performs the best in both hand and fingertip detection. With the large-scale dataset, we achieve fingertip detection error at about 12.22 pixels in $640px \times 480px$ video frame. Finally, using the fingertip detection result, we design and implement an input system for the egocentric vision, i.e., Ego-Air-Writing. By considering the fingertip as a pen, the user with wearable glass can write character in the air and interact with system using simple hand gestures.

1. Introduction

The egocentric vision, also known as the first-person vision, usually refers to capture and process images and videos from cameras worn on the person's head. With the development of smart wearable cameras and augmented reality headset such as Facebook Oculus, Microsoft HoloLens, and Google Glass, the egocentric vision and its

potential applications have drawn lots of attention. Natural and simple human device interaction is an essential factor that encourages people to use it in their daily life. As shown in several conceptual and demonstration videos [1], we believe the pointing gesture and its fingertip trajectory is one of important interaction patterns. Various instructions like pointing, selecting and writing can be easily given by the user. Hence, we focus on the pointing gesture based interaction in the egocentric vision.

Considering both indoor and outdoor situation for a wearable device, the depth camera is not applicable but only RGB color sequences. Hence, the key challenge is to detect and track the fingertip location accurately in real-time under various situations. This is a very difficult task due to many factors like background complexity, illumination variation, hand shape deformation, fingertip motion blur etc. With the depth sensor, several advanced developments in the hand tracking are proposed [23, 22] but we have different camera. The object tracking approaches [10, 8] can be employed for hand tracking but still face difficulties on the fingertip tracking because it's too small. Deep learning framework has provided promising results in the object detection field, including the hand detection [2] but this framework is too slow due to redundant proposals. In [9], a two CNN-based stages hand and fingertip detection framework is proposed but it is not good enough for real-world applications. Currently, in this field, we need a large benchmark dataset to train and evaluate the performance on hand detection and fingertip detection.

In this paper, we have the following three contributions. First, we establish a large dataset, called EgoFinger, containing egocentric videos of various pointing gestures in different scenarios. We in-depth analyze the dataset attributes, like background and foreground color distribution, hand occurrence likelihood, scale and pointing angle distribution of hand and finger, and the manual labeling error analysis. Second, a two-stage Faster R-CNN based hand detec-

tion and dual-target fingertip detection framework is presented. Comparing with state-of-art tracking and detection algorithm, it performs the best in both hand and fingertip detection. Thirdly, we develop and demonstrate a fingertip-based application, the input system for the egocentric vision, Ego-Air-Writing, which allows user write in the air with their fingertip.

2. Related Work

Given the fingertip-based human computer interaction system, the key challenge is to accurately locate such a small fingertip from a large dynamic image or video in real-time. Previous research can be summarized as two categories: tracking methods and detection methods. We review few most related egocentric hand datasets to explore the uniqueness and necessarily of our Ego-finger dataset.

2.1. Tracking methods

How to track a small object like fingertip from a high dimension image (i.e $640px \times 480px$) accurately and robustly remains a great challenge. Template matching[17] and mean-shift [11] methods have been applied for in the constraint environment, and interesting related applications have been demonstrated. In the latest work [25], the tracker is composed by several sections with HOG feature and linear regression classifier, which presents good performance in tracking benchmark [26]. Also, a state-of-art real-time tracking method, Kernelized Correlation Filters (KCF) [8], uses a discriminative classifier to distinguish between the target and surrounding environment. However, these methods are mainly designed and evaluated on short videos (less 30 seconds) and cannot deal with long time continuous tracking problems like drifting and error accumulation. For the long time tracking, the Tracking-Learn-Detection (TLD) [10] is proposed by combining temporal clues and detection-based feature update. It is not fast but works well on the large object appearance variation problem. Still, the long time small object tracking is a challenge issue.

2.2. Detection methods

By using the depth sensor, the hand detection [23, 22] and segmentation [3] have been improved a lot. Unfortunately, considering both indoor and outdoor situations and its wearable application features, we can only employ the RGB camera. In [12, 13], the RGB-based hand detection and segmentation have produced nice results but face challenges with the saturation and illumination variation. In [18], the skin detector, DPM-based context and shape detector are combined to detect hands, but their method is time-consuming due to the sliding window strategy. Deep learning related methods have nice results on the detection problem. Region-based CNN is applied in [2] to detect hands but this framework is too slow due to repeated computation of

redundant overlapping proposals. In [4], the detector only reports the existence of the hand without its location. Faster R-CNN [19] is the most recent general object detection algorithm with good performance. A two stages CNN-based hand and fingertip detection framework is proposed in [9] but the detection accuracy can still be improved for the real world application.

2.3. Related datasets

Currently, in the domain of egocentric hand-related research, it is not easy to obtain a benchmark dataset to evaluate the performance of their methods on hand detection and tracking. The latest data set, called EgoHands [2], contains images captured by Google Glass with manually labeled hand regions (pixel-wise). The dataset aims at recognizing human behavior by egocentric hand gesture. In [3], authors present a small dataset of egocentric gesture with a specific goal of enhancing museum experience. Several data sets collect RGB-D images with the depth camera. In [14], SKIG dataset has Kinect capturing moving gestures under various background conditions. GUN-71 [20] provides a grasp gesture dataset. In [21], a real-time fingertip detection method is proposed with the fingertip labeled RGB-D dataset. This dataset is indoor and not designed for the egocentric vision. To our best knowledge, there is no data set designed and collected for the pointing gesture fingertip-based interaction in the egocentric vision. We believe our data set will provide an effective and efficient benchmark for the related research field.

3. Dataset: EgoFinger

To locate the fingertip in the egocentric vision using deep learning methods, we firstly establish a large-scale dataset containing egocentric images of labeled hand and fingertip, called **EgoFinger**¹. The dataset covers various situations and challenges including the complex background, illumination change, deformable hand shape and skin-like object disturbance, etc. Moreover, we present in-depth analysis on the dataset attributes, like the background and foreground color distribution, hand occurrence likelihood, hand and finger scale and pointing angle distribution, and manual labeling error analysis. We believe the EgoFinger dataset is diversity and credible as a benchmark dataset for the finger and hand related research in the egocentric vision.

The dataset contains 93,729 RGB frames of egocentric videos captured, collected and labeled by 24 different individuals. Half of data is in the indoor environment and half outdoor. Fig 3 demonstrates few representative samples from EgoFinger dataset.

¹The dataset and demo can be downloaded from <http://www.hcii-lab.net/data/SCUTEgoFinger/index.htm>

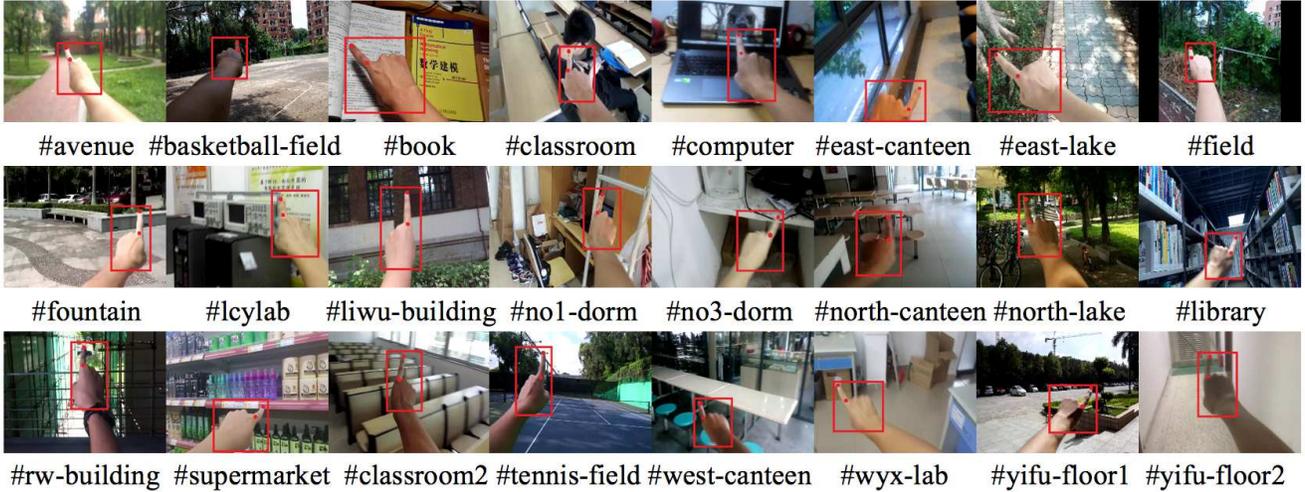


Figure 1. Examples of the dataset frames captured in 24 scenarios.

3.1. Dataset acquisition

By carefully analyzing challenges of the hand and fingertip detection problem in the egocentric vision, we design the data acquisition process to fully represent various situations. In Table 3.1 we summarize the related challenges and corresponding design for the dataset collection. For example, we have half samples taken in indoor and half samples from outdoor, which describes the real world complex background.

Challenges	Causes	Designs
Background complexity	Complicated real world environment	24 different scenes including half indoor and half outdoor
Hand shape deformability	Different user hand and different pointing direction	Unconstrained user hand scale and unconstrained pointing direction
Illumination variety	Light exposure, shading, etc.	Unconstrained walking of experimenter
Skin-like object interference	Faces, arms, wooden object, etc	Collect samples without sleeves and pointing at human face
Motion blur	Frame rate slower than movement	Manually drop out samples that human cannot recognize

Table 1. Challenges, their causes and corresponding dataset designs

The dataset is collected by 24 experimenters separately in 24

different scenes. These scenes are designed to be half indoor and half outdoor. For more detailed information, we give the frame number of all 24 packages in Table 2. During data recording process, every person is required to collect video sequences with the mobile camera. They need to keep hand and fingertip visible inside camera frame with a pointing gesture. Except for this requirement, experimenters are allowed to go anywhere around the appointed scene and to capture video under any illumination with any camera.

After video sequences collection, collectors are required to manually label the important point coordinates of hand and finger. We define a bounding box to locate hand and manually label the top-left point and bottom-right point of the bounding box. Additionally, we define two important points for labeling, that is, the fingertip and index finger joint. We believe the finger physical constraint could improve the detection accuracy.

Two points need to be clarified here. First, we deliberately collect images by right hand because we could simply gain left hand samples by mirroring all samples. Secondly, we collect samples with only one hand using pointing gesture because the dataset aims to evaluate methods of single hand detection and fingertip detection, while single hand situation is more common in egocentric vision. Dataset of more gestures will be established in the future.

3.2. Dataset analysis

To clarify the complexity of background and foreground, we analyze the dataset from the aspect of color distribution. To begin with, we select an indoor package and an outdoor package and then calculate the RGB histogram to describe the general environment. After that we draw the correlation index of two histogram so as to confirm the fact of background complexity. Package Basketball field and package Classroom are selected as representatives.

Fig. 3 shows the color histogram of an indoor scene package and an outdoor scene package to visually reveal the distinction of background environment. Furthermore, the correlation indexes of each channel are -0.0711, 0.239939 and 0.218826 in order blue, green and red. The result proves the background complexity

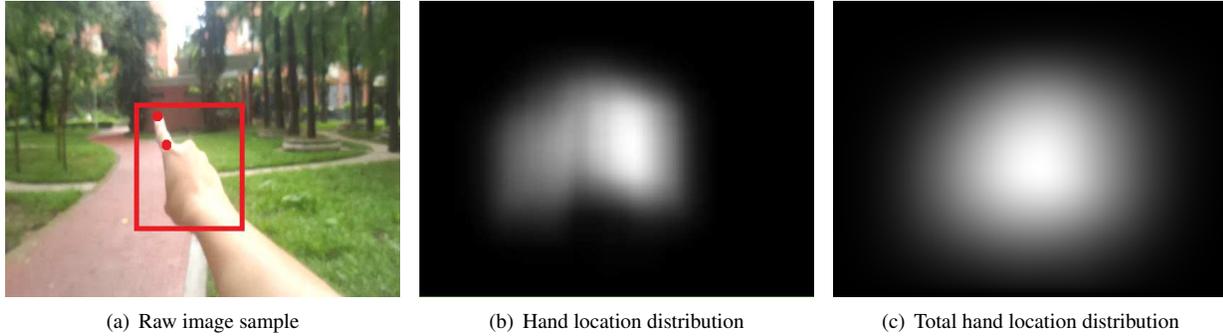


Figure 2. Hand position frequency calculation and result.

claimed in previous sections.

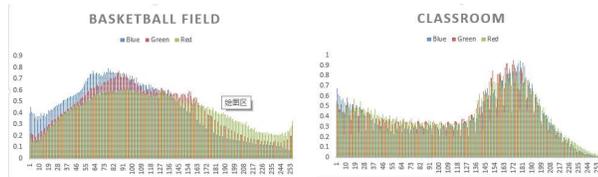


Figure 3. Color distribution of package Basketball field and package Classroom

To evaluate the location frequency of hand, we calculate a 2D distribution matrix of hand position by the following algorithm shown in 2. Loop over a sample package and find out hand location in each frame. Each pixel of distribution matrix add one in the area corresponding to hand location district. After looping, a normalization to 0-255 is taken to the distribution matrix as for visualization.

Fig 2(b) and Fig 2(c) shows the result of location frequency. It fits the Gaussian distribution. The discovered distribution reveals that the hand locates in the vision center more often, which is essentially consistent with human eye visual mechanism. Ac-

Outdoor Scene	Frame No.	Indoor Scene	Frame No.
Avenue	4058	Chinese Book	3001
BasketballField	2894	N. Canteen	4314
Tennis Field	5124	W. Canteen	4185
Football Field	4145	RW Building	3611
N. Lake	3419	N. Library	2495
Yifu Building F1	3806	Yifu Building F2	2870
E. Canteen	3738	Lcy Lab	4084
E. Lake	4151	Classroom	4868
Fountain	4158	Wyx Lab	3222
No. 31 Building	4368	Computer Screen	5088
Liwu Building	5488	Supermarket	1682
No.1 N. Dorm	4281	No.3 North Dorm	4679

Table 2. Detail information of scenes and frame numbers

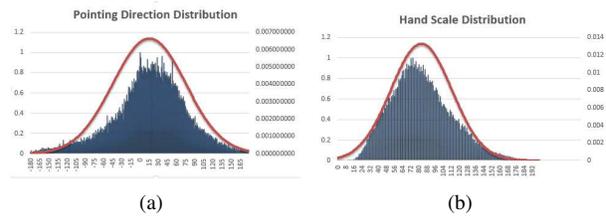


Figure 4. (a) Hand scale distribution (mean: 82.42607841, var: 30.06176942); (b) Pointing direction distribution (mean: 15.22603769, var: 60.13557164).

ording to human vision research, people unconsciously focuses on the central part of vision, which academically named “center bias” [24, 5] or “center prior” [7, 6]. While gazing at the target object, human eyes will relocate target by transferring it to the vision center.

Due to difference of hand appearance, hand-camera distance and pointing direction of each individual at each moment, hands are deformable in even a short period of time. So as to reveal the distribution of hand scale and pointing direction, we draw histograms as in Fig 4.

By applying normality test, we found that both distributions are subjected to Gaussian distribution, which confirms that the dataset is distributed in balance covering numerous different hand instance and is fitted with human daily using behavior.

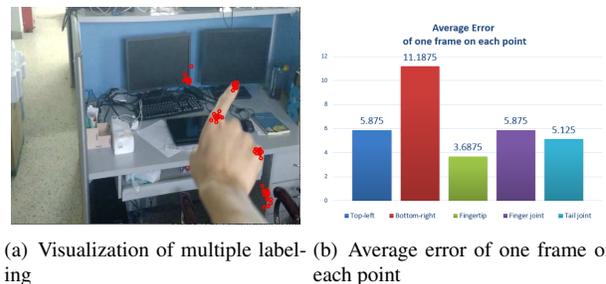


Figure 5. Error analysis

Errors are inevitable while manual labeling. In the case of the dataset, manual labeling error mainly because of individual difference of understanding on points. Experimenters considered differently on how is the bounding box like or on where fingertip is. In

order to evaluate the manual labeling error, experimenters are required to label the same video of about 1000 frames. Visualization of multiple experimenter labeling and the histogram of average error of one frame on points are shown in Fig 5.

The result shows that experimenters are more divergent in bounding box top-left and bottom-right points and are relatively coincident while labeling fingertip. Therefore, while evaluating hand detection accuracy or fingertip detection preciseness on the dataset, the manual error should be brought into consideration.

4. Hand and Fingertip Detection

Although recent CNN-based approaches can generate good results on the object detection, directly locating the tiny fingertip in RGB videos is still very challenge. Following the two-stage CNN-based pipeline in [9, 15], we first detect the hand by extracting it in the bounding box. In this work, we employ the faster R-CNN framework for hand detection based on EgoFinger dataset. Secondly, we find the fingertip position within the hand region.

4.1. Faster R-CNN based hand detection

As discussed before, the Faster R-CNN (FRCNN) [19] has good performance on the object detection. In this paper, we modify faster R-CNN method for hand detection for RGB egocentric vision by using EgoFinger dataset. The region proposal network takes an image as input and outputs a set of proposals with their scores of objectiveness with the novel strategy of sliding anchors. According to the proposals, features in corresponding locations of the feature maps are extracted by spatial pooling on the last constitutional layer. Taken these features as input, the following fully convolution layer output the final scores for all categories and the corresponding parameters of the object bounding box. After non-maximum suppression, the final detection result can be obtained. The detected hand region will be the input of following fingertip detection.

4.2. CNN-based fingertip detector

Given the extracted hand region, the complex background disturbance is reduced. We propose a dual-target CNN-based fingertip detection framework. We believe the physical constraint between the index fingertip and index finger joint can facilitate the fingertip detection. Consider this prior, we label the data accordingly and train a CNN-based network with two key points location. The framework is shown in the following Fig. 6.

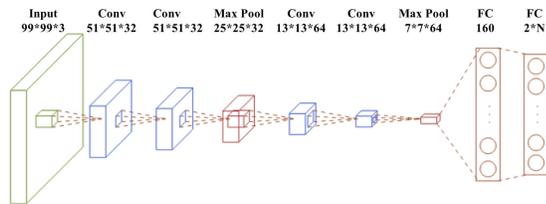


Figure 6. The CNN-based network for the fingertip detection (N is the number of target points, i.e., fingertip and joint).

5. Experimental Results and Discussions

5.1. Data preparation and augmentation

Given totally 24 videos in the EgoFinger, we randomly select 21 as training set, which contains 80,755 frames, and the rest as testing set containing 12,974 frames. Data augmentation techniques are applied in our experiment to reduce the risk of overfitting. For the training data set, we firstly flip the image horizontally to generate left-hand samples. Secondly, we randomly enlarge or shrink the size of image scale. Moreover, we randomly rotate the image with certain angle.

5.2. Hand tracking and detection

Since the hand detection and tracking is the first stage of our proposed algorithm, we evaluate its performance here. We compare four algorithms, i.e., KCF [8], HOG+LR [25], CNN-based hand detection (BHD) [15], CNN-based attention hand detection (AHD) [15] and FRCNN, and their performance is shown in Fig. 7. It is worth to mention that KCF and HOG+LR are tracking methods and fail on long videos. To have a fair comparison, we picked three short video clips (less 1000 frames) from our testing set. Generally speaking, short video clips are less challenge due to smaller environment variation and less object deformation. Also, following the tracking framework, we manually initialize the first frame hand position. Among all three videos, CNN-based detection methods are better than tracking methods because they can avoid the drifting problem. In the whole testing data set, FRCNN-based hand detection reaches the best performance and we will use its result for the fingertip detection.

5.3. Fingertip detection

We have implemented the related tracking algorithms for fingertip problem, like KCF [8], mean-shift [11] and template matching [17]. These methods all failed and loosed track in less than hundred frames and cannot generate quantitative comparison. In the proposed CNN framework, based the hand detection result, we can find the position of fingertip. Fingertip detection comparison results are showed in Fig 5.3. We compare the fingertip detection error using three different hand detection results, i.e., manually labeled ground truth (GT), AHD and FRCNN. The GT-F detects fingertip with 9.37 pixels error in average, FRCNN-F is 12.22 pixels and AHD-F is 15.72 pixels. The hand detection accuracy has direct and important impact on the fingertip detection.

6. Interaction application: Ego-Air-Writing system

We design an input system for the egocentric equipment by considering the fingertip as a pen and allowing it write in the air. Hence, the fingertip trajectory is recognized as the input character for the system. We called the system **Ego-Air-Writing**. The input system is constructed by three main modules, which are the gesture recognition, fingertip localization and character recognition. The following Fig. 9 shows the whole process of writing in the air for the egocentric vision equipment.

The Ego-Air-Writing system has three states: preparation, writing and selection. The writing state is mainly based on the

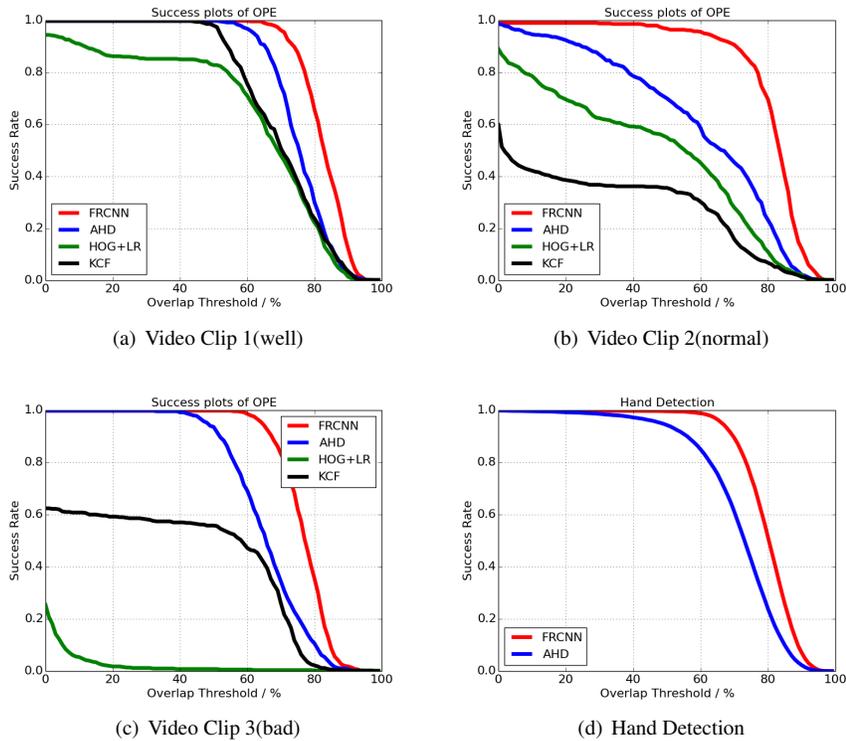


Figure 7. Performance comparison of hand detection/tracking performance on three short egocentric videos (a-c) and whole testing set (d). (*OPE* represents one pass evaluation.)

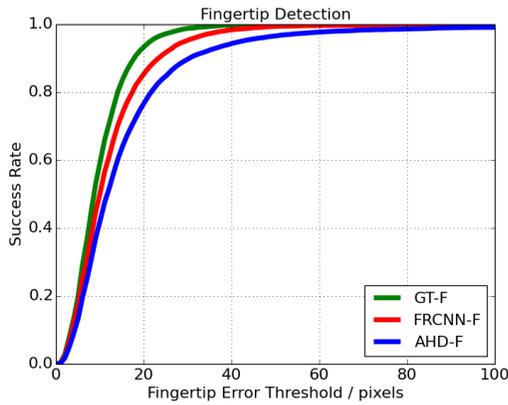


Figure 8. Fingertip detection error comparison on the whole testing data set.

frame-wise detected fingertip location to construct a character trajectory. As for the preparation and selection state, we define two simple hand posture as the controlling signals, as shown in Fig. 9. When the preparation posture is detected, the system clears previous writing record, then emits start signal to get fingertip detection algorithm ready. During air writing, the system locates the fingertip position in real-time and shows it on the writing area. When the fingertip stops moving for a few frames, the system considers the writing finished. Then the smoothen writing trajectory is used for the character recognition using [16]. The first five recognized

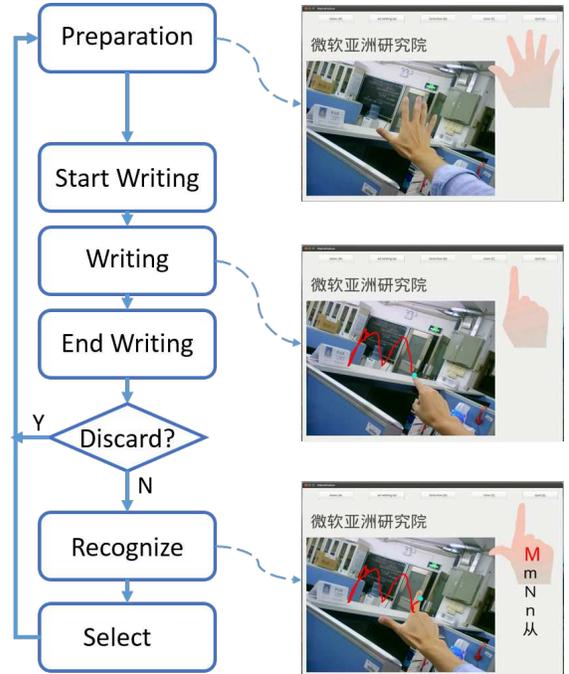


Figure 9. Illustration of gesture transition and flowchart of Ego-Air-writing system.



Figure 10. Character writing examples using Ego-Air-Writing system.

characters are returned and shown on the right side of the interface. User can select one of them with designed hand posture. We have collected a small data set and apply the CNN-extracted feature for these hand postures recognition. In the following Fig. 10 we present two character writing process in the Ego-Air-Writing System.

7. Conclusion and Future Work

This paper discusses the pointing gesture based interaction solution for the smart wearable glasses application. We have proposed the two-stage CNN-based method to detect the fingertip from egocentric videos in all possible application scenarios. To train such model, we have collected a large-scale pointing gesture dataset of multiple subjects and various situations. We designed the dataset acquisition process carefully to make it general, complete and representative. By applying Faster R-CNN based hand detection and multi-point fingertip method, the overlap rate of hand detection is 80% and fingertip detection error is 12.22 pixels in the 640*480 image. Last but not the least, we implement and demonstrate the input system for the egocentric vision using the pointing and other few simple gestures. By considering the fingertip as a pen, it is effective and efficient to input and interact with the wearable glasses. In future work, we plan to further improve the fingertip detection accuracy, enlarge the dataset with multiple gestures and develop corresponding gesture recognition algorithms. With accurate gesture recognition, precise fingertip detection and some other relevant techniques, people could design and build more interesting interaction applications and systems for the egocentric vision based equipment.

8. Acknowledgement

This research is supported in part by MSRA Sponsored Project (No.: FY16-RES-THEME-075), NSFC (Grant No.: 61472144), GDSTP (Grant No.: 2015B010101004, 2015B010130003, 2015B010131004), Fundamental Research Funds for Central Universities (Grant No.: 2015ZZ027).

References

- [1] <https://www.microsoft.com/microsoft-hololens/en-us>. 1
- [2] S. Bambach, S. Lee, D. J. Crandall, and C. Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1949–1957, 2015. 1, 2
- [3] L. Baraldi, F. Paci, G. Serra, L. Benini, and R. Cucchiara. Gesture recognition in ego-centric videos using dense trajectories and hand segmentation. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*, pages 702–707. IEEE, 2014. 2
- [4] A. Betancourt, P. Morerio, L. Marcenaro, M. Rauterberg, and C. Regazzoni. Filtering svm frame-by-frame binary classification in a detection framework. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 2552–2556. IEEE, 2015. 2
- [5] M. Bindemann. Scene and screen center bias early eye movements in scene viewing. *Vision research*, 50(23):2577–2587, 2010. 4
- [6] M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S. Hu. Global contrast based salient region detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 37(3):569–582, 2015. 4
- [7] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(10):1915–1926, 2012. 4
- [8] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 37(3):583–596, 2015. 1, 2, 5
- [9] Y. Huang, X. Liu, X. Zhang, and L. Jin. Deepfinger: A cascade convolutional neuron network approach to finger key point detection in egocentric vision with mobile camera. In *The IEEE Conference on System, Man and Cybernetics (SMC)*, pages 2944–2949. IEEE, 2015. 1, 2, 5
- [10] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(7):1409–1422, 2012. 1, 2
- [11] T. Kurata, T. Okuma, M. Kurogi, and K. Sakaue. The hand mouse:GMM hand-color classification and mean shift tracking. In *IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pages 119–124, 2001. 2, 5
- [12] C. Li and K. M. Kitani. Model recommendation with virtual probes for egocentric hand detection. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2624–2631. IEEE, 2013. 2

- [13] C. Li and K. M. Kitani. Pixel-level hand detection in egocentric videos. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3570–3577. IEEE, 2013. 2
- [14] L. Liu and L. Shao. Learning discriminative representations from rgb-d video data. In *IJCAI*, 2013. 2
- [15] X. Liu, Y. Huang, X. Zhang, and L. Jin. Fingertip in the eye: A cascaded cnn pipeline for the real-time fingertip detection in egocentric videos. *CoRR*, abs/1511.02282, 2015. 5
- [16] T. Long and L. Jin. Building compact mqdf classifier for large character set recognition by subspace distribution sharing. *Pattern Recognition*, 41:2916–2926. 6
- [17] W. Mayol, A. Davison, B. Tordoff, N. Molton, and D. Murray. Interaction between hand and wearable camera in 2D and 3D environments. In *Proc. British Machine Vision Conference*, 2004. 2, 5
- [18] A. Mittal, A. Zisserman, and P. H. Torr. Hand detection using multiple proposals. In *BMVC*, pages 1–11. Citeseer, 2011. 2
- [19] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Neural Information Processing Systems (NIPS)*, 2015. 2, 5
- [20] G. Rogez, J. S. Supancic, and D. Ramanan. Understanding everyday hands in action from rgb-d images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3889–3897, 2015. 2
- [21] X. Suau, M. Alcoverro, A. López-Méndez, J. Ruiz-Hidalgo, and J. R. Casas. Real-time fingertip localization conditioned on hand gesture classification. *Image and Vision Computing*, 32(8):522–532, 2014. 2
- [22] J. S. Supancic, III, G. Rogez, Y. Yang, J. Shotton, and D. Ramanan. Depth-based hand pose estimation: Data, methods, and challenges. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 1, 2
- [23] J. Tompson, M. Stein, Y. Lecun, and K. Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics (TOG)*, 33(5):169, 2014. 1, 2
- [24] P.-H. Tseng, R. Carmi, I. G. Cameron, D. P. Munoz, and L. Itti. Quantifying center bias of observers in free viewing of dynamic natural scenes. *Journal of vision*, 9(7):4, 2009. 4
- [25] N. Wang, J. Shi, D.-Y. Yeung, and J. Jia. Understanding and diagnosing visual tracking systems. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 2, 5
- [26] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2