Real-time Multi-view Bimanual Gesture Recognition

Geoffrey Poon, Kin Chung Kwan, Wai-Man Pang*
School of Computing and Information Sciences,
Caritas Institute of Higher Education, Hong Kong, China
e-mail: {gpoon,kckwan,wmpang}@cihe.edu.hk

***Abstract*—This paper presents a learning-based solution to tackle the real-time gesture recognition of bimanual (two hands) gestures which is not well studied from the literature. To overcome the critical issue of hand-hand self-occlusion problem common in bimanual gestures, multiple cameras from diversified views are used. A tailored multi-camera system is constructed to acquire multi-views bimanual gesture data, and data from each view is then fed into a separate classifier for learning. Thus, to ensemble results from these classifiers, we proposed a weighted sum fusion scheme of results from different classifiers. The weightings are optimized according to how well the recognition performed of the particular view. Our experiments show multiple-view results outperform single-view results.**

Keywords-Gesture recognition, bimanual gesture, multiview, learning based recognition, occlusion

I. Introduction

Hand gesture is often a common media of communication in human history apart from speech and voice, and is also considered as one of the most convenient and intuitive forms of control and command to the computer. Among the existing hand gesture recognition approaches, vision-based methods are the most popular and suitable for wider range of applications as they require no special equipment or wearing of gloves. The community continuously put most of the efforts in recognizing single-handed gesture, and achieved rather satisfactory results. However, there are fewer works target for gestures formed by two hands (*bimanual gesture*). Although methods for single-hand recognition are applicable to bimanual gesture, a serious problem that need to be solved is the hand-hand self-occlusion.

This paper attempts to tackle the vision-based recognition of static bimanual gesture by learning from color and shape features of the hand gesture sequence. To minimize the problem of self-occlusion, we constructed a multi-camera setup to cover a wider range of view angles. By arranging the cameras in diversified viewing points, more features can be provided to compensate other views in the recognition. However, conflict may occur between the recognized gestures from different views. As a result, a fusion scheme of results from different views, our major contribution, has been designed. This fusion scheme has to consider the performance of a particular view to a certain target gesture, because hand gestures may look similar from a certain view causing ambiguity in recognition. Thus, our fusion scheme tries to introduce weightings so as to balance the contribution of different cameras for different gestures based on their performance. These weightings are obtained by an optimization process.

A total of 6 different bimanual gestures are used in our preliminary study. The experimental results reveal that, with the use of multiple views and an optimized weighting-based fusion scheme, the recognition accuracy is improved for certain gestures comparing to their single view counterparts. The proposed bimanual recognition is deployed on a simple game and a sign language learning system to illustrate its interactive-rate capability.

II. Related Works

Real-time hand gesture recognitions had been widely studied for decades. Based on the technique, we can mainly classify existing methods into vision-based, glove-based, and colored marker approaches.

Glove-based methods employ special designed glove with electromechanical parts to know the relative position of various hand sections of user [1], [2]. Its accuracy in measuring hand gesture is supposed to be the highest among the three approaches. However, the needs of special instrument hinder glove-based methods to be popular.

Colored marker approaches [3] thus require users to paint their hand or wearing glove with colors in different sections. The orientation and pose of hands can be estimated quickly by the colors. However, the criteria of color painted hands look weird and definitely less attractive then vision-based recognition of bare hands.

Comparing to the other two approaches, vision-based methods do not require any special devices other than cameras. Erol et al. [4], Sarkar et al. [5] as well as Rautaray and Agrawal [6] had presented detailed reviews about vision-based hand tracking. However, most of the works only focus on single-hand recognition. Later, vision-based recognition are further enhanced by hand extraction using depth sensing techniques with Kinect [7] or Leap Motion [8]. Wang et al. [9] locate and segment single hand from depth images acquired from Kinect. Their method then performs superpixels to represent the hand surfaces. EMD is applied to measure the distance between the distributions of the centers of superpixels. Tzionas et al. [10] further enhanced 3D hand tracking with discriminative salient points.

Recently, many attempts are targeting for bimanual gesture recognition. Kristensson et al. [11] relies on the more

978-1-5386-6396-7/18/$31.00 ©2018 IEEE

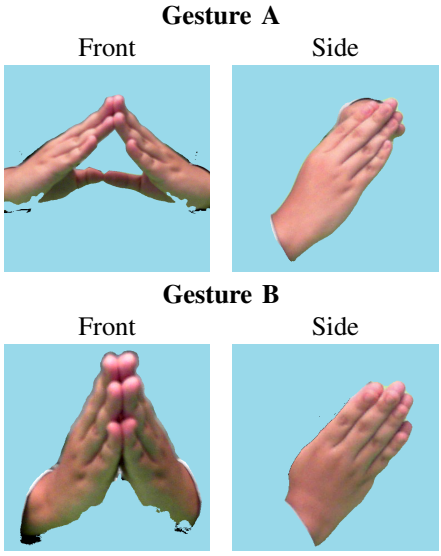

Fig. 1. Gestures may look different in certain view, but similar in another view due to self-occlusion.

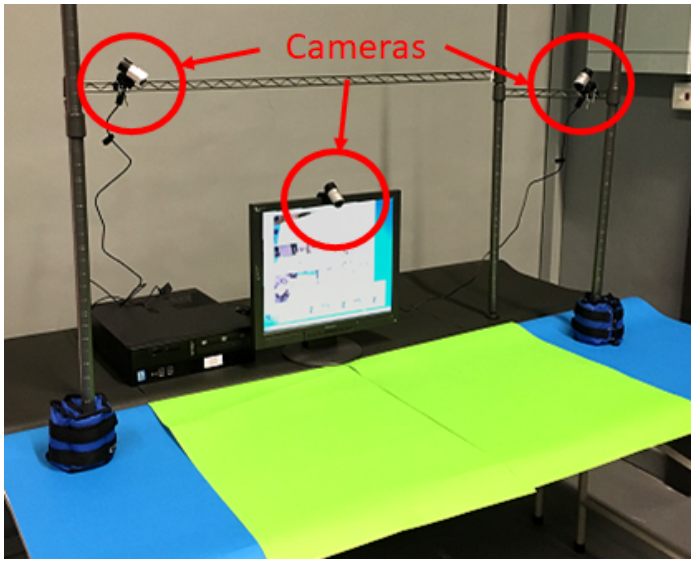

Fig. 2. Our multi-view setup environment with three cameras for capturing data.

robust fully-body skeleton recognition in identifying hand positions, and then enhances the two-hand motional gesture with probabilistic algorithm. However, the work does not put much efforts in recognizing the pose of hand during motion. Oikonomidis et al. [12] proposes recognizing two hands using RGB-D camera with a model-based method that matches all the reprojected hand gestures with the use of Particle Swan Optimization (PSO). However, the tracking fails when there is self-occlusion of two hands. Saeed et al. [13] improves bimanual tracking with temporal coherence and the assumption on limited appearance change during self-occlusion occurs. While this assumption is not always reasonable, normally users will not perform a bimanual gesture separately in each hand at the first place. In contrast, our method tries to solve the occlusion problem with a multi-view approach. By examining from a wider range of viewing angles to the two hands, we improve our confidence in identifying a certain hand posture.

III. Multi-view Setup

Before going into the detail of our method, we first introduce our multi-view setup for acquiring hand gesture data.

One major difficulty in recognizing bimanual gestures is its highly self-occluded properties. Fig. 1 shows two examples of bimanual gestures. These two gestures look rather different from the front view, while we cannot easily differentiate them from the side view. It is because the unique features of these gestures are being occluded. This example illustrates that bimanual gestures are prone to ambiguity if we recognize the gestures with only one view. To overcome this problem, a direct solution is to employ multi-view approach in order to minimize the effect of self-occlusion.

Fig. 2 shows our multi-camera setup. Three cameras are installed in diversified viewing points. We named the views of these three cameras as left, front, and right view respectively. It is worth to note that our setup does not require precise positioning and orientation setting during installation, a rough one is enough. To calibrate the camera, we only need to roughly point the cameras toward the same point. Thanks to the robustness of our learning-based method, our recognition is highly flexible and repeatable for different environments.

IV. Bimanual Gesture Recognition

By using our three-camera setup, we captured a set of bimanual gesture images(examples in Fig. 3) from three different views. These bimanual gestures are chosen from the sign languages and games, thus they are with high practical values. Our training samples are prepared by labeling a square hand region inside each camera frame, followed by rescaling and feature extraction. Finally, we employ Support Vector Machine (SVM), which is a widely used for recognition, to learn a robust classification of the hand features.

A. Image Features Extraction

To improve robustness in learning, it is common to extract features from the images. Similar to most of the existing hand recognition methods, our method begins with a segmentation of hand based on the skin color. It allows the classifier to ignore most of the background, and focus on the targeted hands in the image. We do this by employing the skin color model of [14], which is a set of color criteria for skin in RGB, HSV, and YCrCb spaces. The use of multiple color spaces suppresses the effect of different lighting conditions.

With the segmented hand images, we normalize their size to 200×200 to obtain scale-invariant features. Because of the homogenous and smooth skin surfaces, the variation of features in hands are relatively limited. Thus, we model the image features by considering two basic features: color and shape. Our color feature is constructed by computing the histogram of oriented gradients (HOG) [15] in the color space. For shape feature, edge orientation histograms (EOH) [16]

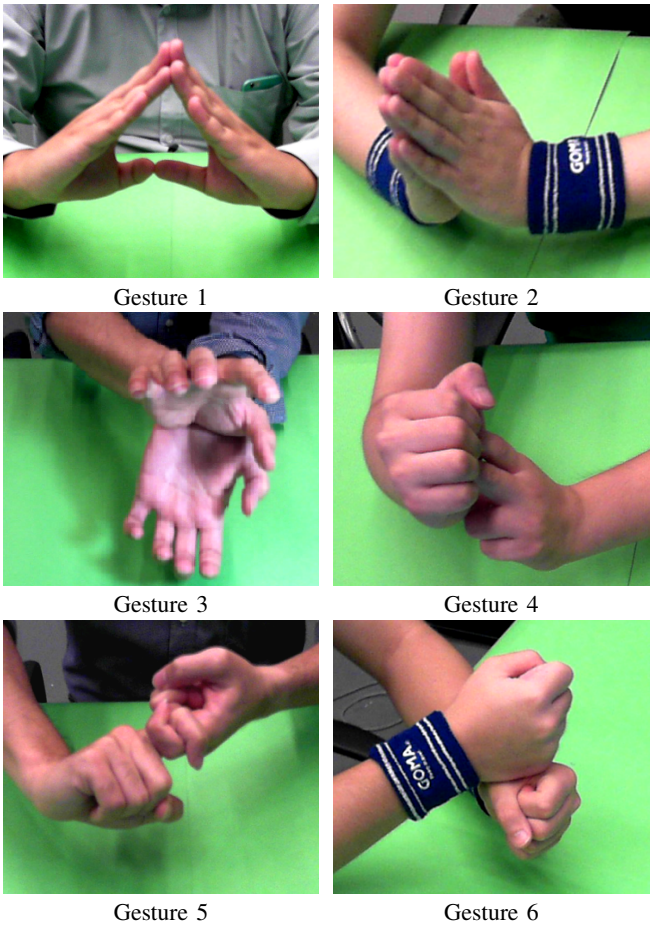

Fig. 3. The six bimanual gestures used in our experiments. They are selected from sign languages and games.

with Canny edge detection is employed. Texture feature like real-time gabor wavelet [17] is not employed as no obvious texture on hand can be found. To suppress the effect of lighting direction, we use unsigned orientation for both HOG and EOH in our model. Both HOG and EOH form feature vectors with 3600 dimensions.

B. Classifying with SVM

With the feature extracted, we train the gesture classifier with SVM. However, naïvely putting all the features into single SVM is inappropriate in our multi-view situation, as the appearances of hands from different views vary significantly. Naïvely learning all features together greatly confuses the SVM classifier and leads to poor results.

Hence, we decide to tackle the different features from different views separately by utilizing independent SVM classifiers. Each SVM is only trained with a single feature of one view. In other words, we used six different SVMs in total to learn the two features for the three views. In all SVMs, we use the Gaussian radial basis function as kernel with gamma $\gamma = 4.8$ to support non-linear learning.

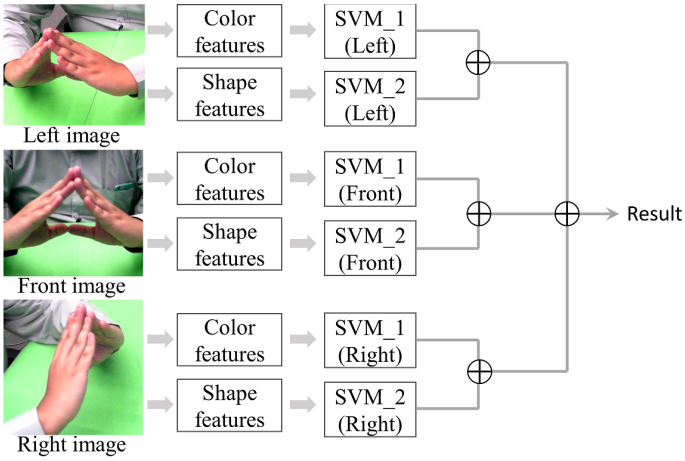

Fig. 4. Our prediction process. A final prediction is decided by combining six SVM of different views and features.

C. Gesture Classification

After training the SVMs, they can be used for gesture recognition individually. However, as we have six SVMs, the prediction may come up six different results in the worse cases. Since bimanual gestures may look similar in certain views and causes ambiguity (e.g. Fig. 1) when the unique features are being occluded, we can expect that the prediction may fail by using the result of this single view in such cases. This poses a problem that we cannot purely selecting one of the six results with highest confidence score. We need to decide the most reliable result by considering all SVMs together.

Our idea is to fuse the results by a weighted sum on all SVM confidence scores from different views. Fig. 4 shows the flow of prediction of our method and the $\oplus$ symbols notate the weighted sum. The final score $S_g(I)$ of having gesture g in image I is defined as

$$S_g(I) = \omega_{g,L} \times S_{g,L}(I) + \omega_{g,F} \times S_{g,F}(I) + \omega_{g,R} \times S_{g,R}(I) \quad (1)$$

where L, F, and R represent the left, front, and right view respectively. ω is the salience of the view for different gesture. It is used to reduce the contribution of the view that may cause ambiguity in bimanual gesture. $S_{g,V}(I)$ is the confidence score from the SVM classifiers of view V, which is defined as the average of SVM scores for color and shape features. The final prediction result is the gesture g' with highest final score $S'_g(I)$.

D. Optimization of Fusion Weightings

To obtain the weighting ω for all gestures and views, we have the following assumption. If a gesture can be easily recognized by the images captured from a view (high accuracy), this gesture should have unique features in this view, so classification result from this view is more likely to be correct. In contrast, if a gesture is often classified wrongly in a view (low accuracy), certain ambiguity should have occurred, possibly because of occlusion. It would be better if we limit

TABLE I
THE WEIGHTING ω OF VIEWS FOR DIFFERENT GESTURES.

	Initial			Final		
	Left	Front	Right	Left	Front	Right
Gesture 1	0.28	0.46	0.26	0.00	0.92	0.08
Gesture 2	0.22	0.46	0.32	0.13	0.30	0.56
Gesture 3	0.35	0.32	0.33	0.04	0.28	0.67
Gesture 4	0.19	0.47	0.34	0.31	0.26	0.41
Gesture 5	0.32	0.38	0.30	0.30	0.61	0.07
Gesture 6	0.31	0.39	0.30	0.00	0.99	0.01
Negative	0.33	0.33	0.33	0.33	0.33	0.33

TABLE II
THE ACCURACY OF RECOGNITION USING DIFFERENT VIEWS.

	Left	Front	Right	Our method
Gesture 1	62.6%	100.0%	68.5%	**100.0%**
Gesture 2	55.6%	96.9%	59.3%	**83.3%**
Gesture 3	92.2%	100.0%	100.0%	**100.0%**
Gesture 4	58.1%	83.3%	70.5%	**97.0%**
Gesture 5	93.2%	98.1%	89.3%	**100.0%**
Gesture 6	91.3%	93.6%	82.1%	**100.0%**
Negative	96.9%	100.0%	95.3%	**98.4%**
Overall	78.5%	96.0%	82.1%	**97.0%**

its contribution to the final result. Base on this assumption, to obtain the weightings, we first conduct a leave-one-out cross-validation (more detail in Sec. V), and compute the accuracy of different gestures from different individual views (Table II). The ratio of these accuracy values are used to calculate an initial ω of one view. Without the loss of generality, the initial weight of left view $\omega_{g,L}$ is defined as:

$$\omega_{g,L} = \frac{A_{g,L}}{A_{g,L} + A_{g,F} + A_{g,R}} \tag{2}$$

where A is the accuracy of recognition using single view. This equation also applies to right and front views.

Note that the initial weights here are not necessary to be optimal. Thus, we further adjust the weightings by optimization with simple gradient descent method and obtained the values of ω (Table I). The objective function is simply the accuracy of using such weighting.For the negative data, we set the weightings of all the views equal.

V. RESULTS AND APPLICATIONS

We carried out a series of experiments to evaluate the performance of our bimanual gesture recognition method. Six subjects are invited to capture the gesture images by our three-camera setup. To increase the variety of the data, these subjects are different in age and with different clothing (e.g., Long or short sleeve). We captured 120 different images from each subjects for each gesture in one view. Thus, there are 2160 images for one gesture in total.

A. Experiments

Our first experiment is a leave-one-out cross-validation. This experiment is to validate the recognition performance based on separated training and testing samples. Each time we use the data set of one subject as a query, and the remaining data of other subjects as the training data set. The experiment is repeated for each subject. Table II shows the mean accuracy of recognition in this experiment. It is obvious that most of our multi-view results outperform other single view results.

As expected, the front view done well for hand recognition. It is because conventional hand gestures are often designed to show in front of the other people for communication. By using our method, we can further improve the accuracy by using combined view. It validates the correctness of our multi-view approach.

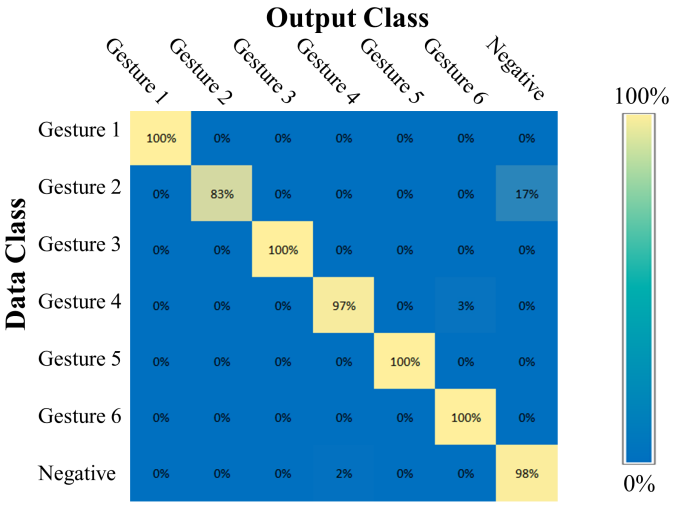

Fig. 5. The confusion matrix of our method. Yellow represents more samples while blue represents less samples.

Fig. 5 shows the confusion matrix of our recognition using combined view. It indicates how many samples of a data class (Y-axis) is classified into an output class (X-axis). The population of each block is color-coded from blue (Less samples) to yellow (More samples). From the confusion matrix, there is an obvious diagonal highlighted in yellow,which means that most of the samples can be classified into the correct classes. It evidences that our method works well for bimanual gesture recognition.

B. Comparisons with Commercial Sensors

The second experiment is to compare our method with existing commercial sensors for bimanual hand gesture recognition. Kinect [7], [18] and Leap Motion [8] are chosen, as they are widely used for gesture recognition.

These two sensors required to reconstruct the skeleton model of our body or hand, which often makes an assumption that every part of our body or hand is visible to the sensor. Thus, when the hands are highly overlapped, their skeleton reconstruction is error-prone and fail to recognize the hand gesture properly. We do the comparison by performing the

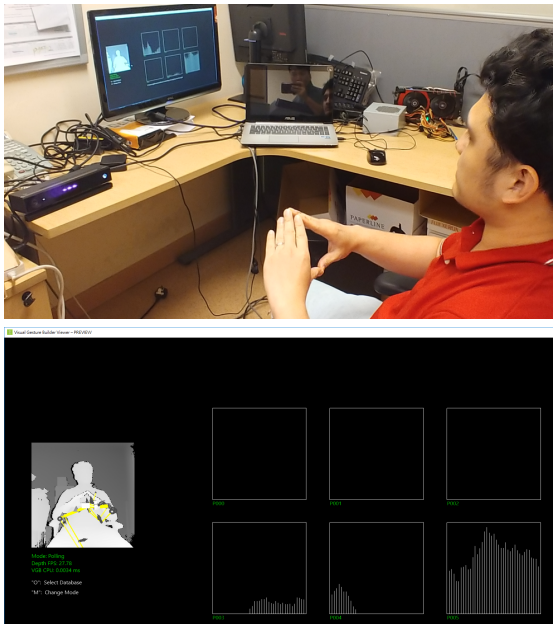

Fig. 6. Our six bimanual gestures are recognized using Kinect Visual Gesture Builder (VGB). It only achieves a 60% accuracy. The screen capture shows Gesture 1 is being confused with Gesture 6.

same gestures as in Fig. 3 and try to recognize with single Kinect and single Leap Motion. According to our experiment, Kinect can only achieve 60% accuracy for bimanual gesture. Leap Motion even failed to detect the existence of hands due to the self-occlusion. Figure 6 shows the experiment settings with the use of Kinect and its Visual Gesture Builder (VGB). We tested with our six bimanual gestures and found poor results. For example, a Gesture 6 is being recognized as Gesture 1 shown in the screen capture of VGB. In contrast, our proposed method can achieve 100% accuracy for Gesture 1 in the experiments as shown in Fig. 5.

VI. Conclusions and Future Works

We presented a learning-based method to perform bimanual gesture recognition using SVM. To tackle the self-occlusion problem, we use multiple cameras from different viewing angles to capture the hand data. Hence, ambiguity occurs in one view can be compensated by another view. To align the results from different views, a weighted sum fusion scheme is proposed. Optimal weightings are obtained in order to adjust the contribution of a particular view based on its ambiguity on a certain gesture.

While our current method has several limitations. First, we assume the input data are static images. Thus, our method cannot handle gestures with motion well in the current implementation. In the future, we will extend our method to motion gestures by using multiple consecutive frames as learning data. Second, our segmentation purely based-on the skin color. Similar to others, using skin color may not be reliable. Lastly, we assume that the hand regions in the data are manually labeled on the image. However, we believe that this problem can be easily solved by any hand tracking or hand detection methods. But more investigations are needed.

Acknowledgment

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong SAR, China (Ref. No. UGC/FDS11/E03/15).

References

[1] F. Parvini, D. Mcleod, C. Shahabi, B. Navai, B. Zali, and S. Ghandeharizadeh, "An approach to glove-based gesture recognition," in *Proceedings of the 13th International Conference on Human-Computer Interaction.*, Berlin, Heidelberg, 2009, pp. 236–245, Springer-Verlag.

[2] A. Karime, H. Al-Osman, W. Gueaieb, and A. El Saddik, "E-glove: An electronic glove with vibro-tactile feedback for wrist rehabilitation of post-stroke patients," in *2011 IEEE International Conference on Multimedia and Expo*, July 2011.

[3] R. Y. Wang and J. Popović, "Real-time hand-tracking with a color glove," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 63:1–63:8, July 2009.

[4] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly, "Vision-based hand pose estimation: A review," *Comput. Vis. Image Underst.*, vol. 108, no. 1-2, pp. 52–73, Oct. 2007.

[5] A. R. Sarkar, G. Sanyal, and S. Majumder, "Hand gesture recognition systems: a survey," *International Journal of Computer Applications*, vol. 71, no. 15, 2013.

[6] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey," *Artificial Intelligence Review*, vol. 43, no. 1, pp. 1–54, 2015.

[7] K. K. Biswas and S. K. Basu, "Gesture recognition using microsoft kinect®," in *5th International Conference on Automation, Robotics and Applications (ICARA)*. IEEE, 2011, pp. 100–103.

[8] L. E. Potter, J. Araullo, and L. Carter, "The leap motion controller: a view on sign language," in *Proceedings of the 25th Australian computer-human interaction conference: augmentation, application, innovation, collaboration*. ACM, 2013, pp. 175–178.

[9] C. Wang, Z. Liu, and S. C. Chan, "Superpixel-based hand gesture recognition with kinect depth camera," *IEEE Transactions on Multimedia*, vol. 17, no. 1, pp. 29–39, Jan 2015.

[10] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall, "Capturing hands in action using discriminative salient points and physics simulation," *International Journal of Computer Vision*, vol. 118, no. 2, pp. 172–193, 2016.

[11] P. O. Kristensson, T. Nicholson, and A. Quigley, "Continuous recognition of one-handed and two-handed gestures using 3d full-body motion tracking sensors," in *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*. ACM, 2012, pp. 89–92.

[12] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Tracking the articulated motion of two strongly interacting hands.," in *CVPR*. 2012, pp. 1862–1869, IEEE Computer Society.

[13] A. Saeed, R. Niese, A. Al-Hamadi, and B. Michaelis, "Coping with hand-hand overlapping in bimanual movements," in *IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*. IEEE, 2011, pp. 238–243.

[14] N. Rahman, K.C. Wei, and J. See, "Rgb-h-cbcr skin colour model for human face detection," in *MMU International Symposium on Information& Communications Technologies (M2USIC 2006)*. MMU, 2006.

[15] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, June 2005, vol. 1, pp. 886–893 vol. 1.

[16] K. Levi and Y. Weiss, "Learning object detection from a small number of examples: the importance of good features," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, June 2004, vol. 2, pp. II–53–II–60 Vol.2.

[17] W.-M. Pang, K.-S. Choi, and J. Qin, "Fast gabor texture feature extraction with separable filters using gpu," *Journal of Real-Time Image Processing*, vol. 12, no. 1, pp. 5–13, 2016.

[18] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, "Robust part-based hand gesture recognition using kinect sensor," *IEEE transactions on multimedia*, vol. 15, no. 5, pp. 1110–1120, 2013.