# A RGB-D Face Recognition Approach without Confronting the Camera

Wei Zhou, Jian-xin Chen, and Lei Wang
Key Lab of Broadband Wireless Communication and Sensor Network Technology
(Nanjing University of Posts and Telecommunications), Ministry of Education
Nanjing, China, 210003
e-mail: weizhou_geek@163.com; chenjx@njupt.edu.cn; wanglei@njupt.edu.cn;

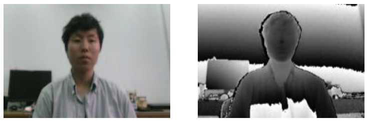
***Abstract*—Face recognition research mainly focuses on traditional 2D color images, which is extremely susceptible to be affected by external factors such as various viewpoints and has limited recognition accuracy. In order to achieve improved recognition performance, as well as the 3D face holds more abundant information than 2D, we present a 3D human face recognition algorithm using the Microsoft's Kinect. The proposed approach integrates the depth data with the RGB data to generate 3D face raw data and then extracts feature points, identifies the target via a two-level cascade classifier. Also, we build a 3D-face database including 16 individuals captured exclusively using Kinect. The experimental results indicate that the introduced algorithm can not only achieve better recognition accuracy in comparison to existing 2D and 3D face recognition algorithms when the probe face is exactly in front of Kinect sensor, but also can increase 9.3% of recognition accuracy compared to the PCA-3D algorithm when it is not confronting the camera.**

***Keywords-3D face recognition; Kinect; RGB-D images; XML file; classifier***

## I. INTRODUCTION

Nowadays, face recognition technology presents tremendous value of research and board application prospects. However, the face recognition is a challenging problem which suffers not only from viewpoint various and illumination like the other object recognition, but also from the high inter-class similarity [1] and facial expression which are distorted specific to the face recognition. Due to 2D color images used for face recognition encompass limited information about a human face and the 2D face recognition is easily affected by various external factors, researchers have considered utilizing 3D data captured by specialized sensors in recent years. Although using 3D information has led to the improvement of recognition performance compared to the 2D face recognition, the high cost of acquisition device limits the feasibility in practical applications.

With the advent of new motion-sensing technique, the 2D color image called RGB image along with the Depth image can be obtained by the camera Kinect at the same time. Kinect [2] is equipped with three asymmetric cameras: color camera, depth camera and infrared projector. RGB camera can be used for shooting color video images in the range of perspectives. 3D Depth camera combined with the infrared laser projector is used for returning per pixel depth value by transmitting and receiving infrared, which represents the relative distance between that pixel and the Kinect sensor during the image capturing.



(a) RGB image (b) Depth image

Figure 1. RGB-D image captured by Kinect sensor.

A RGB-D [3] image provides more object information than the 2D color image because the raw 3D data collected by Kinect are in the form of depth map instead of true 3D mesh. Apart from using Kinect for 3D scene reconstruction [4], robotic applications [5] and general object recognition [6], recent papers have used Kinect sensor with the depth information for face detection, gender recognition [7, 8] and so on. For example, Li et al. [9] proposed an algorithm that used the RGB-D images captured by Kinect for recognizing faces under the effect of different covariates. Moreover, Gaurav et al. [10] described an approach which combined entropy, visual saliency and depth information with HOG for feature extraction and random decision forest for classification based on the Kinect face databases containing 106 faces. Kinect also has its unique face recognition algorithm explained in [11, 12] by Microsoft.

Figure 1 shows an example RGB-D image captured by the Kinect sensor, which includes 2D RGB image and 3D depth image. Due to the limitation of 2D image-form principle, we assert that the RGB-D image obtained from the Kinect sensor can potentially be used to mitigate the effect of external factors such as pose, expression and illumination. Also, 2D face recognition approaches can only be used to recognize a human face when it is directly in front of the Kinect. Since the feature of an RGB-D image captured by Kinect is quite different from a regular 3D map, existing 3D face recognition approach may not be directly applied to the RGB-D image. Therefore, in this paper, we propose a novel RGB-D face recognition algorithm which can be used on the data obtained by Kinect sensor to deal with the scenario that the probe face is not confronting the camera.

The key contributions in this paper are three-fold:

- Propose a novel algorithm integrated the RGB images with the Depth map captured by Kinect for 3D face recognition to improve the recognition performance.

**978-1-4673-8126-0 /15/$31.00 ©2015 IEEE**

- Compare the proposed algorithm with other existing 2D and 3D face recognition approaches under different scenarios and the proposed algorithm improves the recognition accuracy when a probe face is directly in front of the Kinect sensor.
- The accuracy of recognition with various viewpoints is discussed using the PCA-3D algorithm and the proposed algorithm outperforms the PCA-3D algorithm when a probe face is not confronting the Kinect camera.

The other parts of this paper are organized as following: Section II introduces the framework of the 3D face recognition system while section III describes the specific algorithms of classification. Then, section IV reports the experimental results of 3D face recognition using Kinect. It suggests that the face recognition performance has been improved compared to other algorithms by using both the RGB information and the depth data. The last section draws the conclusions.

## II. Framework of System

The overall structure of the 3D face recognition system using Kinect is shown in Figure 2. Firstly, 3D face raw data can be generated by integrating the RGB image with the depth map captured by Kinect sensor. The TinyXml is then used to extract and operate the feature points. At the end of the system, Feature information is used as input to the trained two-level cascade classifier included Decision Tree Classifier and Improved Euclidean Distance Classifier for establishing the identity label.

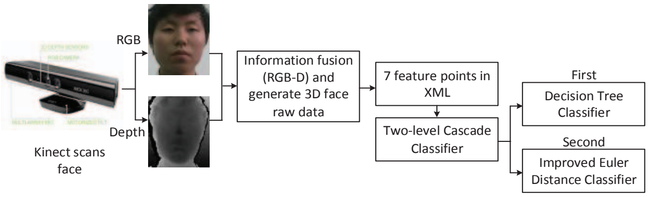

RGB
Depth
Kinect scans face
Information fusion (RGB-D) and generate 3D face raw data
7 feature points in XML
Two-level Cascade Classifier
First
Decision Tree Classifier
Second
Improved Euler Distance Classifier

Figure 2. The 3D face recognition system using Kinect.

### A. *Generating 3D Fundamental Data using RGB-D*

DirectX [13] is a multimedia application programming interface created by Microsoft and implemented by C++. In this paper, Direct2D is used to draw the background while Direct3D is used to draw the 3D human face.

Kinect sensor provides 1347 independent vertices for each tracked face model. These vertices are corresponding to 2630 triangles, as well as 7890 indexes. From the aspect of initialization, we obtain the color frame to display background, the skeleton data frame for tracking ID and HD facial frame to get the vertices.

By using the existing interface of Kinect and integrating the RGB image with the depth map, the 3D face fundamental data can be obtained in txt files.

### B. *Extracting Feature Points*

Before saving the 3D face data, we extract 7 face feature points and store them in XML files: the top, bottom, left side, right side of a human face, as well as the left eye, the right eye and the nose in the current capture image.

TinyXML [14] is a C++ tool library used to parse xml files and store data. This parsing library generates the model of DOM tree in memory. Therefore, traversing this XML tree is convenient, which means we can read and write xml data whenever we want. In this paper, we use TinyXML tool to access the XML files, which includes facefeature.xml and cascade1.xml.

According to the feature data, we compute the length as well as the width of the face, the distance between two eyes and nose-eyes middle distance. The length and width ratio of the face can be used for the first level of the two-level cascade classifier. All these four types of face information can be used for the second level of the cascade classifier.

### C. *Two-level Cascade Classifier*

Classification is an important method of data mining to classify the samples and predict data. In real-time system, 3D face feature vector of a probe individual should be compared with the 3D face samples in the database when using a single classifier to classify and discriminate the probe face. In order to ensure the real-time performance of the system and as far as possible to reduce the computational complexity, the classification discriminant of cascade classifier can be used. A strong cascade classifier is generated by concatenating the weak classifier and the strong classifier. In other words, the strong classifier can be constituted by continue to choosing other feature to train from the features after the weak classifier selection and then conducting cascade.

## III. Algorithm of Classification

Among several approaches of classification, two-level cascade classifier is explored for classification in this paper. The two-level cascade classifier composes of Decision Tree Classifier and Improved Euclidean Distance Classifier.

### A. *Decision Tree Classifier*

In the first step of two-level cascade classifier, the Decision Tree Classifier can be used. The Decision Tree Classifier uses a decision tree as a predictive model.

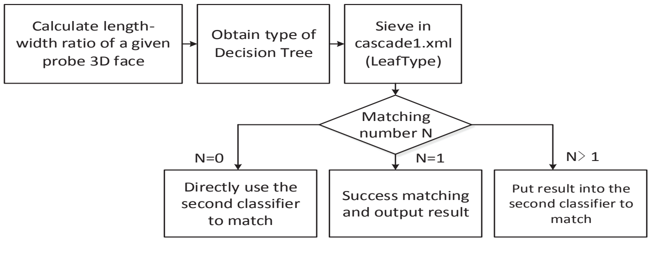

Calculate length-width ratio of a given probe 3D face
Obtain type of Decision Tree
Sieve in cascade1.xml (LeafType)
Matching number N
N=0
N=1
N> 1
Directly use the second classifier to match
Success matching and output result
Put result into the second classifier to match

Figure 3. Illustrating the steps of Decision Tree Clssifier.

After the first step discriminant, a large number of non-target samples can be removed and the number of candidate 3D face data sharply declines.

Figure 3. shows the process of Decision Tree Classifier. First, length-width ratio of a given probe 3D face should be computed to obtain a type of Decision Tree. Next, according to the type of Decision Tree, we sieve in cascade1.xml (LeafType) to get the matching number. If the matching number is zero, we should directly use the second classifier to match; if the matching number is only one, we can output the matching result. Otherwise, we ought to put result into the second following classifier to match.

### *B. Improved Euclidean Distance Classifier*

Then, choosing the improvement of variance reciprocal Weighted Euclidean Distance (WED) Classifier which can greatly improve the efficiency of discriminant. The Euclidean Distance Classifier includes the Standard Euclidean Distance Classifier and the Improved Euclidean Distance Classifier.

#### *1) Standard Euclidean Distance Classifier*

The standard variance reciprocal weighted Euclidean Distance Classifier is widely used in the comparison between two templates. Comparing the unknown 3D face features with already trained 3D face features. If and only if the variance reciprocal weighted Euclidean Distance WED between the probe and the k-th feature achieve the minimum, input data are classified as k-th 3D face data. To extract 7 dimensional feature vector of the 3D face data in the database and calculate the average $u_{ki}$ and the standard deviation $\delta_{ki}$ of i-th Dimensional feature about each type of 3D face data. Variance reciprocal weighted Euclidean Distance is calculated using (1),

$$WED(\mathrm{k}) = \sum_{i=1}^{7} \frac{(x_i - u_{ki})^2}{(\delta_{ki})^2} . \qquad (1)$$

In the formula of standard variance reciprocal weighted Euclidean Distance, the coefficient of $(x_i - u_{ki})^2$ is $1/(\delta_{ki})^2$ . Therefore, the contribution of each 3D face feature is inconsistent when computing $WED(\mathrm{k})(k = 1,2,...,\mathrm{m})$ .

#### *2) Improved Euclidean Distance Classifier*

Considering the efficient of universal attributes, the improvement of variance reciprocal weighted Euclidean Distance Classifier is obtained using (2),

$$WED(\mathrm{k}) = \sum_{i=1}^{7} (\frac{1}{\delta_i} \times w_{ki} \times (x_i - u_{ki})^2), k = 1,2,...,m , \qquad (2)$$

Where the data tuple is $X,(x_1, x_2,..., x_7, c)$ . $x_i (i = 1,2,...,7)$ represent attribute values of each feature. $c$ represents category and belongs to $\{c_1, c_2,..., c_m\}$ , where $m$ is the number of classes. $u_{ki},(u_{k1}, u_{k2},..., u_{k7})$ represent the mean vector of each type of feature.

The training steps of the algorithm using the Improved Euclidean Distance Classifier are as follows:

**Algorithm 1** The Training Steps of Improved Euclidean Distance Classifier

```
begin
Require: vectors of m classes
1: Compute the mean vectors $u_k (k = 1,2,...,m)$ of m classes.
      For the feature i of each class,
         Statistic of maximum $\max_{ki}$ and minimum $\min_{ki}$ .
2: Compute the variances $\delta_{ki} (k = 1,2,...,m; i = 1,2,...,7)$ of each attributes for all classes.
3: Compute $\delta_i = \sum_{k=1}^{m} \delta_{ki}, (i = 1,2,...,7)$.
end begin
```

Here $m$ is the number of all the classes. The training steps of Improved Euclidean Distance Classifier use the vectors of $m$ classes to compute the mean vectors, the variance of each attribute for all classes, as well as the maximum and minimum of the feature $i$ .

In addition, the classification steps are as follows:

When we calculate $w_{ki}$ , if the value of $x_i$ is between $\min_{ki}$ and $\max_{ki}$ , then $w_{ki} = 1$ . Otherwise, $w_{ki} = 100$ . In other words, when $x_i$ overflows the range of minimum and maximum, $x_i$ has much more contributions to the discriminant and the weight should be increased.

**Algorithm 2** The Classification Steps of Improved Euclidean Distance Classifier

```
begin
Require: the 3D face data to be classified
1: Compute the distances $WED(k)(k = 1,2,...,m)$ between the face data
            and the samples in database.
      Statistic of maximum $\max_{ki}$ and minimum $\min_{ki}$ .
2: If the distance $WED(k)(k = 1,2,...,m)$ between the face data
            and the $k-th$ sample is the minimum，
   then the input 3D face data can be classified as the $k-th$ face data.
end begin
```

Algorithm 2 depicts the classification steps of Improved Euclidean Distance Classifier, which uses the 3D face data to be classified as the input and computes the distances between the face data and the samples in database. After the training steps and classification steps of Improved Euclidean Distance Classifier, we can implement the classification of the probe 3D face.

## IV. EXPERIMENTAL RESULTS

The recognition performance of the proposed algorithm is analyzed on the 3D-face database, which are generated by combining the RGB image with the depth map.

Also, several existing algorithms based on 2D and 3D images are compared in this paper. We also do the experiments over different distributions of the collected data.

In order to further analyze the recognition performance of different kinds of face recognition approaches, we also establish a 3D face database including 16 male and female subjects with multiple 3D face information captured by Kinect.

Figure 4. 2D RGB images of different individuals captured by Kinect.

Figure 5. 3D depth maps of different individuals captured by Kinect.

Figure 4. and Figure 5. show that the RGB images from different individuals have high inter-class differentiability while the Depth maps from different individuals own low intra-class variation. Hence, we can integrate the characteristics of the two kinds of images to generate RGB-D images, which can provide more information about the captured faces and recognize the probe face from various viewpoints.

Table I. demonstrates the recognition accuracy of various 2D and 3D face recognition algorithms on the established 3D face database, which obey Uniform Distribution $U(a,b), \mathrm{a} < \mathrm{b}$. While Table II. demonstrates the recognition accuracy of various 2D and 3D face recognition algorithms on the established 3D face database over Normal Distribution $N(\mu,\sigma^2)$.

TABLE I. THE RECOGNITION ACCURACY OVER UNIFORM DISTRIBUTION

| Modality of face data | Algorithm | Recognition Accuracy(%) |
|---|---|---|
| 2D color image | HOG | 75.2 |
| | PHOG | 80.6 |
| | FPLBP | 85.7 |
| 2D color image & 3D depth map | PCA-3D | 83.6 |
| | Proposed | 90.5 |

TABLE II. THE RECOGNITION ACCURACY OVER NORMAL DISTRIBUTION

| Modality of face data | Algorithm | Recognition Accuracy(%) |
|---|---|---|
| 2D color image | HOG | 86.6 |
| | PHOG | 88.2 |
| | FPLBP | 91.1 |
| 2D color image & 3D depth map | PCA-3D | 93.7 |
| | Proposed | 95.9 |

From the above results, we can find that 2D color image along with 3D depth map can generally achieve better recognition performance than those algorithms only using 2D information such as HOG, PHOG and FPLBP. One hand, the proposed algorithm extract 3D face data from the RGB-D images captured by Kinect can yield higher recognition performance compared to the existing 3D approaches. On the other hand, the face recognition accuracy is bounding to the distribution of the face database, and the Normal Distribution outperforms the Uniform Distribution. Furthermore, when the 3D face data captured by Kinect follows uniform distribution, the random error influences the recognition accuracy between FPLBP and PCA-3D.

Although the 2D face recognition approaches can only identify the probe face, which is directly in front of the Kinect sensor. In this paper, we propose a novel 3D face recognition algorithm based on Kinect sensor, which can improve the recognition accuracy as well as implementing the recognition of a probe face from different kinds of viewpoints.

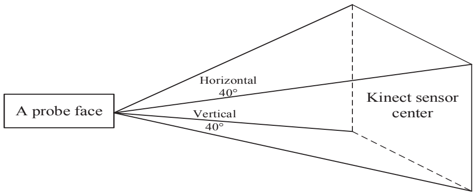


Figure 6. Experimental scenario of two 3D face recognition approaches from several viewpoints.

Figure 6. shows the following experimental scenario of two 3D face recognition approaches from several viewpoints. The horizontal and vertical angles are both 40° , then we divide each of the 40° into +20° , +10° , 0° , -10° and -20° and respectively detect the recognition accuracy of horizontal and vertical angles.

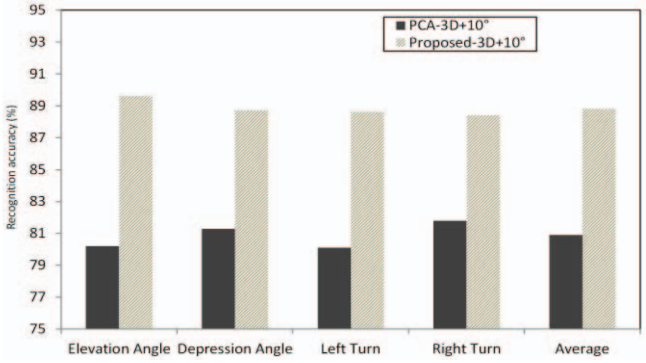


Figure 7. Recognition Accuracy of four different 10° deflection viewpoints using two 3D face recognition algorithms under the Normal Distribution.

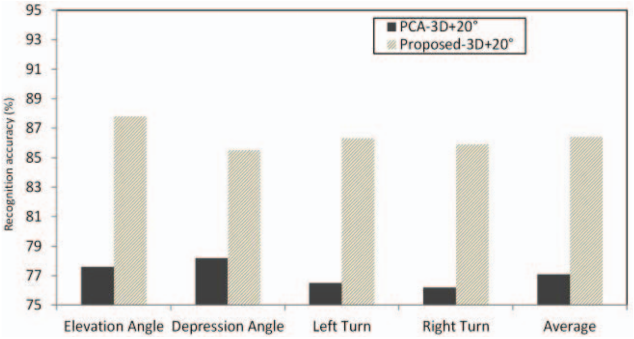


Figure 8. Recognition Accuracy of four different 20 ° deflection viewpoints using two 3D face recognition algorithms under the Normal Distribution.

3D face recognition approaches such as PCA-3D and proposed algorithm in this paper can both recognize human faces from various viewpoints. Figure 7. and Figure 8. show that the proposed 3D face recognition algorithm can perform better than the PCA-3D approach when recognizing the probe face from different viewpoints both under Normal Distribution and the 10° deflection viewpoints have higher recognition accuracy than the 20° deflection viewpoints using both the PCA-3D and the proposed-3D face recognition algorithms. In the experiments, we first access the Kinect sensor as the center. Then from four kinds of vertical and horizontal directions contained elevation angle, depression angle, left turn and right turn, we respectively detect the accuracy of 3D face recognition. Also, each of the directions includes 10° deflection and 20° deflection.

Figure 7. depicts the recognition accuracy of four different 10° deflection viewpoints contained Elevation Angle, Depression Angle, Left Turn and Right Turn using two 3D face recognition algorithms under the Normal Distribution. While Figure 8. depicts the recognition accuracy of those four different 20° deflection viewpoints the same as Figure 7. using two 3D face recognition algorithms under the Normal Distribution. Both of the two figures show that the proposed-3D face recognition algorithm outperforms the PCA-3D algorithm.

Then we can find that 2D face recognition approaches can only recognize human faces using the Kinect sensor exactly in front of the probe face while 3D face recognition algorithms can recognize human faces from different points of view. The other point is that the proposed 3D face recognition algorithm has better performance compared to the PCA-3D algorithm using the same data set under the Normal Distribution. Moreover, the 3D face recognition accuracy declines with the increase of the deflection angles.

From these experimental results we note that 2D color image along with 3D depth map contains more information than 2D color images, which can utilize the 3D data of face to add one dimensional information. Moreover, the RGB-D images captured by Kinect sensor integrate the RGB information with the depth information and the obtained RGB-D images then can be mapped into the spatial coordinates about human faces, which can be precisely used to extract feature points and recognize the final results. In addition, distribution of the 3D face data are used as the input to the classifier, which directly influences the performance of recognition, e.g. the performance is better over Normal Distribution than over Uniform Distribution.

Further, 2D face recognition apporaches only use 2D color images contained no 3D information about the human faces, which fails to recognize the probe face from different viewpoints except confronting the Kinect sensor. Also, the proposed 3D face recognition algorithm uses the exact 3D spatial coordinates about human faces, which can better recognize the probe face compared to PCA-3D algorithm from different viewpoints under Normal Distribution. Additionly, the deflection angles influence the recognition accuracy due to the 3D face raw data and feature points reduce as the angles of deflection increase and the optimal recognition accuracy can be achieved when the Kinect sensor confronting the probe face.

## V. Conclusion

Although the existing face recognition algorithms can not only utilize the 2D color image but also the 3D information is involved, the recognition performance of these face recognition algorithms is limited due to many factors such as the content of captured information, the cost of acquisition devices and so on.

In this paper, an algorithm used RGB-D images obtained by Kinect is proposed. The algorithm uses the combination of 2D color image and 3D depth map, which can generate the raw spatial coordinate data about the captured human face. Then we also extract seven 3D face feature data stored in XML and use a two-level cascade classifier for classification. The experiments performed on created Kinect 3D-face database demonstrate that the proposed algorithm improve the performance of face recognition compared to the other existing algorithms. Also, we can achieve better recognition performance when the collected data obey Normal Distribution $N(\mu,\sigma^2)$ . Further, 3D face recognition algorithms can recognize the probe face from various points of view. However, 2D face recognition can only recognize the probe face when human faces are exactly confronting the Kinect sensor. In the situation of using 3D face recognition approaches, the proposed algorithm can achieve higher recognition accuracy than the PCA-3D algorithm over the Normal Distribution and the recognition performance declines while the probe face's deflection angles increase.

## Acknowledgment

This work is partly supported by the Wireless Broad Communication and Wireless Sensor network, Embedded system, Department of Education Key lab Open Program (Grant No. NYKL201306, TK214001), Education Natural Science Foundation of Jiangsu Province (13KJB510026), and Nanjing University of Posts and Telecommunications Foundation(Grant No. NY211032).

## References

[1] A. F. Abate, M. Nappi, D. Riccio, & G. Sabatino, "2D and 3D face recognition: A survey," Pattern Recognition Letters, vol. 28, no. 14, pp. 1885-1906, 2007.

[2] Z. Zhang, "Microsoft kinect sensor and its effect," MultiMedia, IEEE, vol. 19, no. 2, pp. 4-10, 2012.

[3] P. Krishnan and S. Naveen, "RGB-D face recognition system verification using Kinect and FRAV3D Databases," Procedia Computer Science, vol. 46, pp. 1653-1660, 2015.

[4] S. Izadi, D. Kim, O. Hilliges, et al., "KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera," Proceedings of the 24th annual ACM symposium on User interface software and technology. ACM, 2011, pp. 559-568.

[5] R. A. El-laithy, J. Huang, and M. Yeh, "Study on the use of Microsoft Kinect for robotics applications," Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION. IEEE, 2012, pp. 1280-1288.

[6] S. Tang, X. Wang, X. Lv, et al., "Histogram of oriented normal vectors for object recognition with a depth sensor," Computer Vision–ACCV 2012. Springer Berlin Heidelberg, 2013, pp. 525-538.

[7] R. I. Hg, P. Jasek, C. Rofidal, et al. “An RGB-D database using Microsoft's Kinect for windows for face detection,” 2012 IEEE Eighth International Conference on Signal Image Technology and Internet Based Systems (SITIS), 2012, pp. 42-46.

[8] T. Huynh, R. Min, J. L. Dugelay, “An efficient LBP-based descriptor for facial depth images applied to gender recognition using RGB-D face data,” Computer Vision-ACCV 2012 Workshops. Springer Berlin Heidelberg, 2013, pp. 133-145.

[9] B. Y. Li, A. S. Mian, W. Liu, et al., “Using kinect for face recognition under varying poses, expressions, illumination and disguise,” 2013 IEEE Workshop on Applications of Computer Vision (WACV), 2013, pp. 186-192.

[10] G. Goswami, S. Bharadwaj, M. Vatsa, et al., “On RGB-D face recognition using Kinect,” 2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS), 2013, pp. 1-6.

[11] Z. Cao, Q. Yin, X. Tang, et al., “Face recognition with learning-based descriptor,” 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 2707-2714.

[12] L. Liang, R. Xiao, F. Wen, et al. “Face alignment via component-based discriminative search,” Computer Vision–ECCV 2008. Springer Berlin Heidelberg, 2008, pp. 72-85.

[13] P. Varcholik, “Real-time 3D Rendering with DirectX and HLSL: A Practical Guide to Graphics Programming,” Addison-Wesley Professional, 2014.

[14] L. Thomason, TinyXML[J]. URL: http://sourceforge.net/projects/tinyxml/[Last accessed 24/08/2007], 2007.