Rev Socionetwork Strat (2013) X:XX-XX

Personal Authentication by Using Kinect Sensor

Xuanang Feng[1)], Yi Zuo[2)], Eisuke Kita [1)] and Fumiya Saito [1)]

1) Graduate School of Information Science, Nagoya University
Furo-chou, Chikusa-ku, Nagoya 464-8601, Japan
2) Institutes of Innovation for Future Society, Nagoya University
Furo-chou, Chikusa-ku, Nagoya 464-8601, Japan
zuo@coi.nagoya-u.ac.jp
kita@is.nagoya-u.ac.jp

Received: Date Month 2017 / Accepted: Date Month 20XX

Abstract. This article proposed a new approach of personal authentication by exploring the features of personal face and voice. Regarding face recognition, Microsoft Kinect sensor is used for detecting images involving faces and complex background information. Facial parts, including eyes, nose and mouth, etc., are analyzed as position vectors. As for voice recognition, Kinect microphone array is adopted for recording personal voice. The mel-frequency cepstrum coefficients, the logarithmic power and related values, which are involved in the analysis of personal voice, are also mentioned in this article. Neural network, support vector machine and principal components analysis are employed and compared for the personal authentication. In order to achieve accurate results, 20 examinees are selected, whose face and voice data are used for training the authentication models. The experimental results show the best accuracy when the model is trained by support vector machine using both face features and voice features.

Keywords: Personal authentication, Kinect, Neural network, Support vector machine, Principal components analysis

1 Introduction

Due to the increasing development of information technology and network service, personal authentication systems based on the collection of personal features have been widely studied and applied. Conventional approached of personal authentication systems, which mainly adopt passwords and numbers, are well known as the knowledge authentication; while those adopting material objects like magnetic cards and IC cards are called goods authentication. In addition to conventional systems, the biometric authentication has attracted more attentions at the moment [1]-[3].

Individual biological features, including finger marks, facial features and iris, etc., are used in the biometric authentication system. Facial authentication, e.g. as digital signage, has been considered a feasible marketing way in department stores [2]. The personal face, which is mixed in complex background in images, is picked out from the picture. Previous researches mostly separated face from background in images by means of skin color and brightness distribution [4]. Manabe et al. [6] studied the personal authentication from the perspective of the sizes of personal facial parts and obtained an average authentication accuracy of 50% based on the data of 7 examinees due to the scaling of feature variables. In this article, Kinect for Windows (Kinect for short) [5], is used as the detecting method. Kinect was initially developed by the Microsoft Corp. for capturing gestures for Xbox360. This article uses Kinect to capture facial parts like eyes, nose, mouse and among others as the position vectors. On the other hand, Kinect can also pick out personal voice in the microphone array. Nakagawa and Mori [7] studied the personal voice as the perspective for studying personal authentication. However, this research not only studies personal facial parts, but also explores personal voice for the personal authentication. To address this issue, this article employ the neural network (NN) and the support vector machine (SVM) to define the personal authentication algorithm according to personal facial parts and voice. The authentication accuracy is verified by comparing with the face and voice data of 20 examinees.

This paper is organized as follows: Section 2 describes the system design concept and related theoretical basis. The numerical examples and results, as well as the performance analysis and discussions, are described in Section 3. Finally, the conclusion and future work are made in the last section.

2 Proposed method

2.1 Basic concept

Kinect [5], which is developed by Microsoft Corp. in 2010, is the motion-capture device for Microsoft Xbox 360 (Fig. 1). The use of the Kinect obtains the personal face region from the pictures including the personal face and the complicated background. The position vectors of the personal face parts are estimated by Web API [8] from the face picture. The Kinect microphone array gathers the personal voice, which are analyzed by HTK [10]. Personal authentication is performed by NN and SVM from the features of the personal face and voice.

Figure 1. Kinect sensor [5]

The process of the personal authentication system is summarized as follows and shown in Fig. 2.

1) Obtain the position vectors of the personal heads through the Kinect.
2) Extract the picture of the face region from the picture including the face and the background by applying Web API from the head position vector.
3) Calculate the features of the personal face from the position vectors of the face.
4) Record the personal voice from the microphone array of Kinect.
5) Calculate the features of the personal voice through HTK.
6) Train the parameters of NN or SVM through the features of the face and voice.

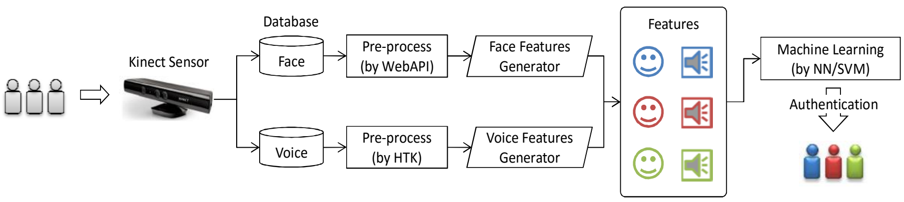

Figure 2. Overview of the proposed personal authentication system

2.2 Description of face features

Kinect takes the personal face picture and then, the use of the Web API [8] estimates the position vector (x, y) of the personal face part such as eyes, nose, mouth, eyebrows and face outline. The position vectors are shown in Fig. 3.

The distances from the base point N1 to the position vector of the personal face parts are calculated and then, the rations between the distances and the pupillary distance p_{d} are calculated as the feature variables [10].

Assuming that the position vectors of the base point N1 and the position vectors of the face part k as (x_{N1}, y_{N1}) and (x_{k}, y_{k}), respectively. The feature variable F_{k} is given as in (1).

$$F_k = \sqrt{\frac{(x_k - x_{\mathrm{N1}})^2 + (y_k - y_{\mathrm{N1}})^2}{p_{\mathrm{d}}}} \tag{1}$$

Web API can estimate the position vectors of 50 face parts.

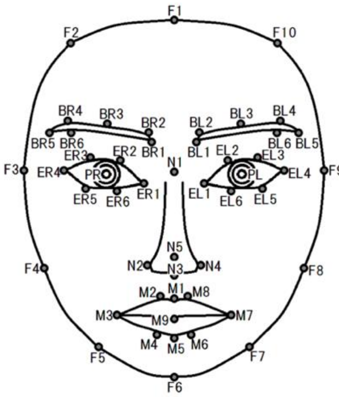

Figure 3. Coordinates of face parts obtained from Web API [8]

2.3 Description of voice features

Personal voice is collected into wave file through the Kinect microphone array.
The use of the voice analysis software HTK [10] estimates the features from the wave file.

As the features, the following 39 quantities are adopted [11]–[13].

- Mel-Frequency Cepstrum Coefficients (MFCC) of 1st to 12th order
- Δ Mel-Frequency Cepstrum Coefficients (ΔMFCC) of 1st to 12th order
- ΔΔ Mel-Frequency Cepstrum Coefficients (ΔΔMFCC) of 1st to 12th order
- Logarithmic power
- Δ Logarithmic power
- ΔΔ Logarithmic power

1) Mel-Frequency Cepstrum Coefficients: The mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency.

Mel-frequency cepstral coefficients (MFCC) are coefficients that collectively make up an MFC. They are derived from a type of cepstral representation of the audio clip (a nonlinear "spectrum-of-a-spectrum") [11], [12]. The difference between the cepstrum and the mel-frequency cepstrum is that in the MFC, the frequency bands are equally spaced on the mel scale, which approximates the personal auditory system's response more closely than the linearly-spaced frequency bands used in the normal cepstrum. This frequency warping can allow for better representation of sound, for example, in audio compression.

The mel scale is a perceptual scale of pitches judged by listeners to be equal in distance from one another. The formula of mel scale to convert f (Hz) into m mel is as in (2).

$$f_{\mathrm{mel}} = 2595 \times \log_{10}(1 + \frac{f}{700}) \quad (2)$$

The MFCC is defined as in (3).

$$c_i = \sqrt{\frac{2}{N}} \sum_{j=1}^{N} m_j \cos(i(j - 0.5)\frac{\pi}{N}) \quad (3)$$

The parameters N, m_j and i denote the flame length, the amplitude of logarithmic filter bank and the order of MFCC, respectively. The function m_j is given as the function of the filter bank energy E_i^x as in (4).

$$m_j = \log(E_i^x) \quad (4)$$

The parameters of low orders (1st to 12th) are used for the voice recognition.

2) Δ Mel-Frequency Cepstrum Coefficients: The Δ Mel-Frequency Cepstrum Coefficients (ΔMFCC) is the dynamic feature which is defined as the finite difference approximation of the time-derivative of the MFCC [12].

The time variation of MFCC is approximated as the time regression model [13]. Then, the ΔMFCC is given as in (5).

$$\Delta c_i(l) = \frac{\sum_{k=-K}^{K} k \cdot c(l_i + k)}{\sum_{k=-K}^{K} k^2} \quad (5)$$

The parameters K and $c_i(l)$ denote the range for calculating the parameters of the regression model and the Mel-Frequency Cepstrum Coefficient of the order i in the flame l, respectively.

Around the starting and the terminal flames, the Mel-Frequency Cepstrum Coefficient is approximated by the starting and the terminal flames as in (6).

$$c_i(l + k) = \begin{cases} c_i(0), & (l - k < K) \\ c_i(L), & (l + k > L - K) \end{cases} \quad (6)$$

The parameter L is the number of the terminal flame.

3) ΔΔMel-Frequency Cepstrum Coefficients: The ΔΔ Mel-Frequency Cepstrum Coefficients (ΔΔMFCC) is the dynamic feature which is defined as the finite difference approximation of the time-derivative of the ΔMFCC.

4) Logarithmic Power: The logarithmic power [11] is defined as in (7).

$$E_i = \log(\sum_{i=1}^{N_l} s_i(l)^2) \tag{7}$$

The parameter $s_i(l)$ and N_l denote the i-th voice information on the flame l and the number of samples on the flame l, respectively.

5) Δ Logarithmic Power and ΔΔ Logarithmic Power: The Δ logarithmic power and the ΔΔ logarithmic power are the dynamic features which are defined as the finite difference approximation of the time-derivative of the logarithmic power and Δ logarithmic power, respectively.

2.4 Algorithm

1) Authentication Algorithm: The authentication algorithms are defined by the multiple regression analysis between the objective variable y and the explanatory variables $\vec{x}$. The objective variable y is the personal name. The explanatory variables $\vec{x} = \{x_1, x_2, \cdots, x_p\}^T$ are the feature variables of the personal face and voice. Since the numbers of the features of the personal face and voice are 47 and 39, respectively, the total number of the explanatory variables is $p = 86$.

The training data are given as follows.

$$(\vec{x}^1, y^1) = \{x_1^1, x_2^1, \cdots, x_{86}^1, y^1\}^T$$
$$\vdots$$
$$(\vec{x}^n, y^n) = \{x_1^n, x_2^n, \cdots, x_{86}^n, y^n\}^T$$

The variable n is the total number of the data sets. The superscript is the number of the data set.

The discriminant analysis determines the relationship between the objective variable y and the explanatory variables $\vec{x}$ from the training data as in (8).

$$y = f(\vec{x}) \tag{8}$$

The use of the above equation discriminates the new data. The function f is determined from the training data by the multiple regression analysis of the NN and the SVM.

2) Neural Network (NN): Artificial neural networks (ANNs, simply NNs) are a network inspired by biological neural network [14], [15]. The NN is composed of the number of formal neurons. The structure of the formal neuron is shown in Fig. 4.

The variables $z_1, z_2, \cdots, z_p$ and $w_1, w_2, \cdots, w_p$ denote the input signals and the weight parameters for the related input signals, respectively.

The relationship between the input variables z_1,z_2, ⋯ , z_p and the output variable z_o in the formal is given as in (9).

$$z_o = \frac{1}{1+e^{-u}} \tag{9}$$

$$u = \sum_{i=1}^{p} w_i z_i - \theta$$

The parameter θ is the threshold.

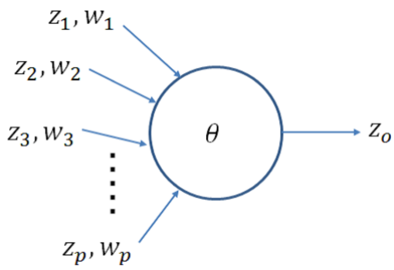

Figure 4. Structure of McCulloch-Pitts model

The three-layered neuron is used for defining the function between the input and output variables (Fig. 5). The parameters of the function are trained according to the error backward propagation algorithm.

a) Prepare the set of the input and the output data for training the parameters of the neural network.
b) Give the random values to the weight parameters and the threshold.
c) Give the input data of the training data set to the neural network and estimate the output data from the network.
d) Update the weight parameters w_1, w_2, ⋯, w_p and the threshold θ so that the error E between the output data of the network and the output data of the training data set are minimized.
e) Repeat the step 4 until the error E is minimized.

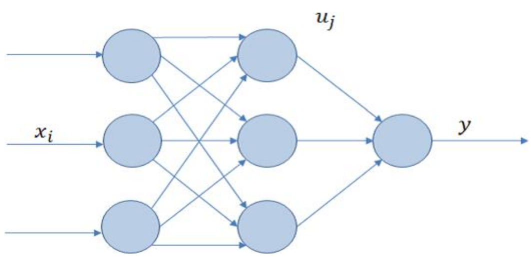

Figure 5. Hierarchical type Neural Network with 3-layer full combination

3) Support Vector Machine (SVM): Support vector machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis [16]–[18].

In this study, the support vector machine of the kernel functions is employed. When the kernel function K is given, the function is given as in (10).

$$y = f(\vec{x}) = \sum_{i=1}^{n} K(x_i, x) + b \tag{10}$$

The support vector function is illustrated in Fig. 5. The parameters a_i and b are determined so that the margin d in Fig. 6 is maximized.

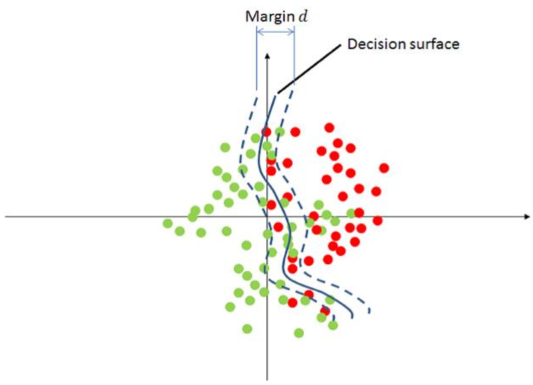

Figure 6. Image of decision surface of SVM

4) Principal Components Analysis (PCA): Principal components analysis is a useful statistical technique that has found application in fields such as face recognition and image compression, and is a common technique for finding patterns in data of high dimension [19]–[21].

The eigenvalues of the correlation coefficient matrix are denoted by λ_i. Algorithm for finding eigenvalues is given as follows.

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_j \geq \cdots \geq \lambda_k \geq 0$$

The proportion of varianceis the rate of each eigenvalue to the whole eigenvalue, as given in (11). The cumulative proportion is the accumulation of proportion of variance, as given in (12).

$$c_j = \frac{\lambda_j}{\lambda_1 + \lambda_2 + \cdots + \lambda_k} \tag{11}$$

$$C_j = \frac{\lambda_1 + \lambda_2 + \cdots + \lambda_j}{\lambda_1 + \lambda_2 + \cdots + \lambda_k} \tag{12}$$

In this study, as the amount of features used to personal authentication is 86, we will discuss the necessary of the dimensionality reduction by using principal components analysis. We choose the principal components whose cumulative proportion is over 0.85.

2.5 Flowchart

The flowchart of the personal authentication system (see Fig. 7) is summarized in a pseudo code as follows.

Input: Face data F, Voice data V

Output: Face features (FFs) using Web API, Voice features (VFs) using HTK

$FFs = Extract\_Features(F), VFs = Extract\_Features(V)$

$M_{F\_NN} = Train\_Model(FFs, "NN"), M_{F\_SVM} = Train\_Model(FFs, "SVM")$

$M_{V\_NN} = Train\_Model(VFs, "NN"), M_{V\_SVM} = Train\_Model(VFs, "SVM")$

$M_{FV\_NN} = Train\_Model(FFs + VFFs, "NN")$

$M_{FV\_SVM} = Train\_Model(FFs + VFFs, "SVM")$

$M_{best} = Select\_Model(M_{F\_NN}, M_{F\_SVM}, M_{V\_NN}, M_{V\_SVM}, M_{FV\_NN}, M_{FV\_SVM})$

While ($CP \leq 0.85$)

  $M'_{F\_NN} = Train\_Model(FFs, "NN" + "PCA")$

  $M'_{F\_SVM} = Train\_Model(FFs, "SVM" + "PCA")$

  $M'_{V\_NN} = Train\_Model(VFs, "NN" + "PPCA")$

  $M'_{V\_SVM} = Train\_Model(VFs, "SVM" + "PCA")$

  $M'_{FV\_NN} = Train\_Model(FFs + VFs, "NN" + "PCA")$

  $M'_{FV\_SVM} = Train\_Model(FFs + VFss, "SVM" + "PCA")$

End While

$M'_{best} = Select\_Model(M'_{F\_NN}, M'_{F\_SVM}, M'_{V\_NN}, M'_{V\_SVM}, M'_{FV\_NN}, M'_{FV\_SVM})$

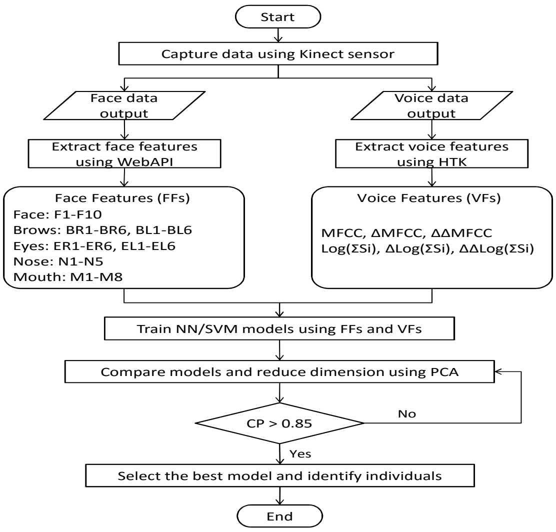

Figure 7. Flowchart of personal authentication system

3 Numerical Examples

3.1 Preprocessing

The evaluation accuracy of the algorithm is confirmed by twenty examinees, which are named from A to T. The experiment environment is shown in Fig. 8. The distance between the examinees is 120 cm and the Kinect is attached at the position of 100 cm height from the floor. The examinees sit on the chair of 45 cm height and face toward the Kinect. Fifteen face pictures are taken from the examinees. In the voice recording, the examinees pronounce ``a e i u e o a o" and the Kinect records it.

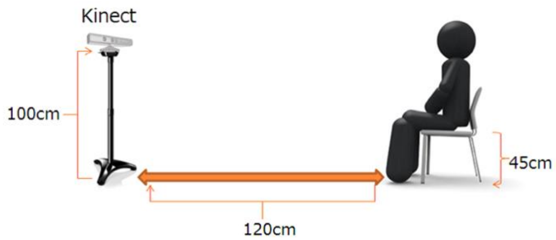

Figure 8. Experiment environment

In order to reduce the overfitting error, we employ the 5-fold cross validation to train the models. To be different from the traditional cross validation, we separate each examinee’s data into five blocks to validate the recognizable performance of models. Therefore, the 5 folds are categorized as Figure 9.

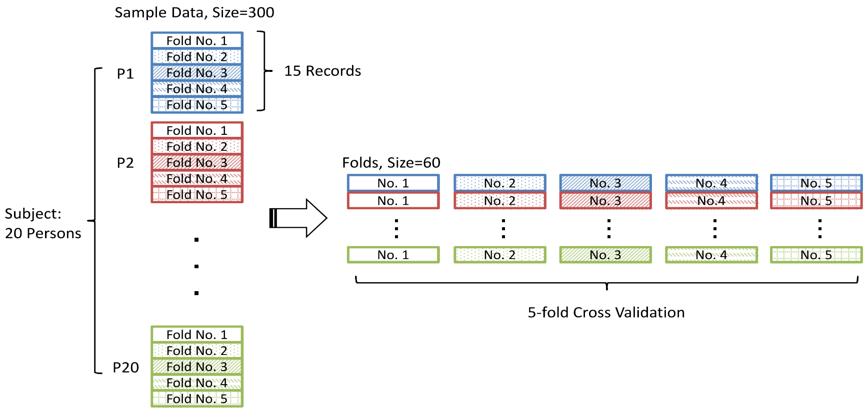

Figure 9. 5-fold cross validation for training models

3.2 Personal authentication with face features (FFs)

1) Neural Network (NN): Personal authentication is performed by using neural network through the personal face features. The results are shown in Table 1.

The "nnet" package[1] in statistical software R [22] is employed for the neural network analysis. Number of nodes on hidden layer is *9*. Parameter for weight decay is *0.1*. Maximum number of the trainings for the neural network training is *100000*.

As shown in Table 1, the best accuracy is 0.7618, in No.3 block. The worst accuracy is 0.6974, in No.1 block. The average accuracy is 0.8633.

2) Support Vector Machine (SVM): Personal authentication is performed by using support vector machine through the personal face features. The results are also shown in Table 1.

The "kernlab" package[2] in statistical software R [22] is employed for the support vector machine analysis. The parameters of Gaussian RBF kernel with σ is *1.0*, and cost is *0.017*.

As shown in Table 1, the best accuracy is 0.9000, in No.1 and No.3 block. The worst accuracy is 0.8333, in No.2 and No.5 block. The average accuracy is 0.8633.

Table 1. Accuracy of NN and SVM by using 5-fold cross validation with FFs

	No.1	No.2	No.3	No.4	No.5	Average
NN	0.6947	0.7568	0.7618	0.7597	0.7403	0.7427
SVM	0.9000	0.8500	0.8333	0.9000	0.8333	0.8633

3.3 Personal authentication with voice features (VFs)

1) Neural Network (NN): Personal authentication is performed by using neural network through the personal voice features. The results are shown in Table 2.

Number of nodes on hidden layer is *9*. Parameter for weight decay is *0.1*. Maximum number of the trainings for the neural network training is *100000*.

As shown in Table 2, the best accuracy is 0.9273, in No.2 block. The worst accuracy is 0.8298, in No.1 block. The average accuracy is 0.8911.

2) Support Vector Machine (SVM): Personal authentication is performed by using support vector machine through the personal voice features. The results are also shown in Table 2.

The parameters of Gaussian RBF kernel with σ is *1.0*, and cost is *0.017*.

---

[1] Package "nnet", https://cran.r-project.org/web/packages/nnet/nnet.pdf.
[2] Package "kernlab", https://cran.r-project.org/web/packages//kernlab.pdf.

As shown in Table 2, the best accuracy is 0.9667, in No.4 block. The worst accuracy is 0.8333, in No.1 block. The average accuracy is 0.8933.

Table 2. Accuracy of NN and SVM by using 5-fold cross validation with VFs

	No.1	No.2	No.3	No.4	No.5	Average
NN	0.8298	0.9273	0.9235	0.8995	0.8752	0.8911
SVM	0.8333	0.8667	0.9500	0.9667	0.8500	0.8933

3.4 Personal authentication with face features (FFs) and voice features (VFs)

1) Neural Network (NN): Personal authentication is performed by using neural network through the personal face and voice features. The results are shown in Table 3.

Number of nodes on hidden layer is *9*. Parameter for weight decay is *0.1*. Maximum number of the trainings for the neural network training is *100000*.

As shown in Table 3, the best accuracy is 0.9530, in No.3 block. The worst accuracy is 0.8348, in No.1 block. The average accuracy is 0.9025.

2) Support Vector Machine (SVM): Personal authentication is performed by using support vector machine through the personal face and voice features. The results are also shown in Table 3.

The parameters of Gaussian RBF kernel with σ is *1.0*, and cost is *0.017*.

As shown in Table 3, the best accuracy is 1.0000, in No.2 block. The worst accuracy is 0.8833, in No.1 and No.5 block. The average accuracy is 0.9333.

Table 3. Accuracy of NN and SVM by using 5-fold cross validation with FFs and VFs

	No.1	No.2	No.3	No.4	No.5	Average
NN	0.8348	0.9323	0.9530	0.9118	0.8808	0.9025
SVM	0.8833	1.0000	0.9833	0.9167	0.8833	0.9333

3.5 Dimensionality reduction by using principal components analysis (PCA)

The principal components analysis can not only extract the significant features, which can also reduce the training time. Furthermore, these benefits can make our proposal more applicable into real-world implications. For example, automatic door access system by using face features. As the face features are too many, which can waste a lot of time on training and recognizing.

Personal authentication is performed by using neural network and support vector machine added with principal components analysis. We compared the results after dimensionality reduction. The results are shown in Table 4.

As shown in Table 4, the dimension of face and voice features were reduced from 86 into 25. The best accuracy on support vector machine through the personal face and voice features dropped from 0.9333 to 0.8900.

Table 4. Accuracy comparisons between NN and SVM by with PCA

	Original Data			Dimensionality reduction with PCA		
	FFs	VFs	FFs + VFs	FFs	VFs	FFs + VFs
Dimension	47	39	86	9	21	25
NN	0.7427	0.8911	0.9025	0.7173	0.8770	0.8883
SVM	0.8633	0.8933	0.9333	0.7700	0.7967	0.8900

3.6 Discussion

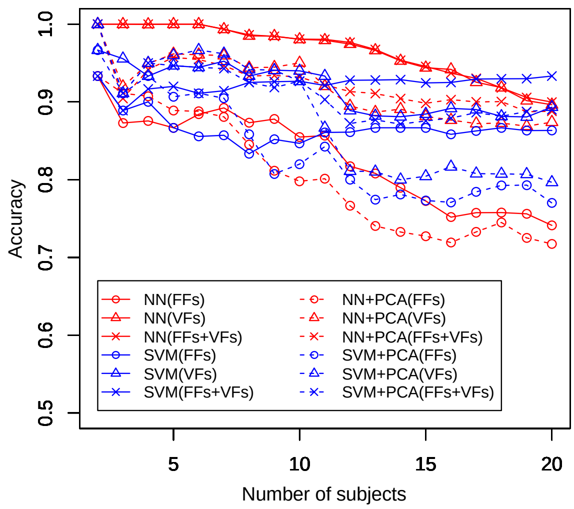

Figure 10. Accuracy of authentication performance for number-specific individuals

The results in Tables 4 and Fig.10 shows that with the increase in the number of subjects, all the accuracy has declined. The average accuracy of original data is higher than dimensionality reduction with principal components analysis. Due to the accuracy decreased significantly, we will select the model without principal components analysis. The average accuracy between neural network and support vector machine is over 0.8 when the personal authentication is performed by the face features alone. When the voice features are adopted, the average accuracy is 0.8922, which is better than that of the face features. While the number of the face features is larger than that of the voice features, the variation of the voice features is bigger than that of the face features. This means that the difference of personal voices is relatively large. When both face and voice features are applied for the support vector machine, the authentication accuracy is 0.9333, which is the best in all results.

4 Conclusion

The personal authentication from the face and voice features observed by Kinect is described in this study. The use of the Kinect detects the face picture area from the picture including both the face and the complicated background. Besides, the personal voice is recorded by the Kinect microphone array. The personal authentication algorithm is defined by the neural network or support vector machine. The accuracy of the authentication algorithms is confirmed by the face and the voice data observed from 20 examinees. The most accurate result is observed when the data of both face and voice are employed and the algorithm is defined by support vector machine. The accuracy is 0.9333.

The future prospects are to increase the security for the development of automatic door systems by using Kinect sensor [23]. And it is also hoped that will be used as a key of vehicle, it enables people to open or lock the door without by themselves when holding luggage in both hands or when taking care of their baby.

References

1. W. Shen and T. Tan, “Automated biometrics-based personal identification", Proceedings of the National Academy of Sciences of the United States of America, vol. 96, no. 20, pp. 11065-11066, 1999
2. S. Ishida, M. Mimura and Y. Seto, “Development of Personal Authentication Techniques Using Fingerprint Matching Embedded in Smart Cards”, IEICE Transactions on Information and Systems, vol. E84-D, no.7, pp. 812-818, 2001.
3. L. O'Gorman, “Fingerprint Verification”, in Biometrics: Personal Identification in Networked Society, A. K. Jain, R. M. Bolle and S. Pankanti (Eds.), Springer US, pp. 43-64, 2006.

4. O. Arandjelovic and R. Cipolla, “Face Recognition from Face Motion Manifolds using Robust Kernel Resistor-Average Distance”, Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004.
5. KINECT for Windows, http://www.microsoft.com/en-us/kinectforwindows.
6. Y. Manabe, R. Saito, Y. Shimada and K. Sugawara, “Frontal View Person Verification by Soft Biometric Features of Gait, Face and Body”, Journal of Japan Society for Fuzzy Theory and Intelligent Informatics, vol. 24, no.5, pp. 988-1001, 2012.
7. S. Nakagawa and K.Mori, “Speaker Change Detection and Speaker Clustering Using VQ Distortion Measure”, IEICE Transactions on information and systems, vol. J85-D-II, no. 11, pp. 1645-1655, 2002.
8. Increment Corporation, Detect Face Web Service, http://detectface.com.
9. U. Park and A. K. Jain, “Face Matching and Retrieval Using Soft Biometrics”, IEEE Transactions on Information Forensics and Security, vol.5, no.3, pp. 406-415, 2010.
10. HTK Speech Recognition Toolkit, http://htk.eng.cam.ac.uk.
11. A. E. Rosenberg, C.-H. Lee and F. K, Soong, “Cepstral channel normalization techniques for HMM-based speaker verification”, Proceedings of Third International Conference on Spoken Language Processing (ICSLP), vol.4, pp.1835-1838, 1994.
12. S.-Y. Chang, B. T. Meyer and N. Morgan, “Spectro-temporal features for noise-robust speech recognition using power-law nonlinearity and power-bias subtraction”, 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 7063 - 7067, 2013.
13. J.-L. Shen, W.-L. Hwang and L.-S. Lee, “Robust speech recognition features based on temporal trajectory filtering of frequency band spectrum”, Proceedings of The Fourth International Conference on Spoken Language, vol. 2, pp. 881 - 884, 1996.
14. W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity”, The Bulletin of Mathematical Biophysics}, vol.5, no.4, pp.115-133, 1943.
15. K. Gurney, “An introduction to neural networks”, UCL Press, 1997.
16. V. N. Vapnik, “The nature of statistical learning theory”, Springer, 1995.
17. C. Cortes and V. N. Vapnik, “Support-vector networks”, Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.
18. A. Ben-Hur, D. Horn, H. T. Siegelmann and V. N. Vapnik, “Support Vector Clustering”, Journal of Machine Learning Research, vol. 2, pp. 125-137, 2001.
19. Herve Abdi, Lynne J. Williams: Principal component analysis, http://www.utdallas.-edu/~herve/abdi-awPCA2010.pdf, 2010
20. Lindsay I Smith: A tutorial on Principal Components Analysis, http://faculty.iiit.ac.-in/~mkrishna/PrincipalComponents.pdf, 26 February 2002
21. Steven M. Holand: PRINCIPAL COMPONENTS ANALYSIS (PCA), http://strata.uga.-edu/8370/handouts/pcaTutorial.pdf, July 2016
22. The R Project for Statistical Computing, http://www.r-project.org/index.html.
23. X. Feng, Y. Yazawa, E. Kita and Y. Zuo, “Control of Automatic Door by Using Kinect Sensor”, Bulletin of Networking, Computing, Systems, and Software, vol. 6, no. 1, pp. 17–21, 2017.