International Journal of the Physical Sciences Vol. 7(7), pp. 1073 - 1082, 9 February, 2012
Available online at http://www.academicjournals.org/IJPS
DOI: 10.5897/IJPS11.1649

Full Length Research Paper

Arabic text-dependent speaker verification for mobile devices using artificial neural networks

Abdulrahman Alarifi[1]*, Issa Alkurtass[1] and Abdul Malik S. Al-Salman[2]

[1]Computer Research Institute, King Abdulaziz City for Science and Technology, P. O. Box 6086, Riyadh 11442, Riyadh, Saudi Arabia.
[2]College of Computer and Information Sciences, King Saud University, P. O. Box 51178, Riyadh 11543, Saudi Arabia.

Accepted 24 January, 2012

Speaker verification is one of the biometric verification techniques used to verify the claimed identity of a speaker. It is mainly applied for security reasons and managing users' authentication. Voiceprint can be used as a unique password of the user to prove his/her identity. In this paper, we propose a new Arabic text-dependent speaker verification system for mobile devices using artificial neural networks (ANN) to recognize authorized user and unlock devices for him/her. We describe system components and demonstrate how it works. We present the performance of our system and analyze its results.

Key words: Speaker verification, neural network, mobile, authentication.

INTRODUCTION

Due to the rapid development in various fields of information technology, data have become more vulnerable to theft and vandalism. This has led to the creation of biometrics as an extra barrier to protect data. Biometrics is a measure of the unique physical characteristics of a person for verifying his/her identity (Jain et al., 2006; Uludag et al., 2004; Campbell, 1997). They are used in identification by comparing stored characteristics with the incoming data and the output score is used in decision making. They began with the idea of the fingerprint in ancient China and extended to include other similar features that distinguish individuals; such as hand geometry, iris scanning, and voiceprint (Bowyer et al., 2008; Jain et al., 2004).

Voiceprint is based on the features of a person's voice which are in turn based on his/her physical characterristics; such as vocal tracts, mouth, nasal cavities and lips that are used in cooperation to create a sound. In general, voiceprint refers to the range of audio frequencies that can be detected and analyzed to identify individuals. It first came under the spotlight in 1935 when it was used as evidence in a crime case (Alghamdi, 1997; Kaji et al., 1997; Ramli et al., 2007). The advantages of using voiceprint are:

1. It does not require specialized equipments (a microphone is sufficient).
2. It can be used remotely.
3. It is a good alternative for people with special needs.

However, voiceprint is affected by surrounding (background) noise, psychological or physical state of the speaker and aging.

Voiceprint can be utilized in speaker recognition, which is the act of validating a speaker's identity using his/her voice. This can be done either by identifying the speaker (Speaker Identification) or by authenticating his/her claimed identity (speaker verification) (Campbell, 1997; Kung and Lin, 2005; Campbell et al., 2006). Speaker verification can be text-dependent or text-independent. In text-dependent, a predefined phrase is used in enrollment and recognition. While in text-independent, any spoken phrase would be sufficient for verifying the speaker's identity. Since the phrase is not predefined, it takes usually longer in text-independent systems to better identify the vocal features. Furthermore, text-independent systems require more training data than text-dependent for the same reason. In general, text-independent systems

*Corresponding author. E-mail: aarifi@kacst.edu.sa.

tend to perform worse than text-dependent and can be more susceptible to recording playback.

There are many applications for speaker verification; mostly in areas where it is desirable to secure actions such as in banking and telecommunication. Due to its low implementation cost and its acceptability by the end users, it is used in access control and transaction authentication. Also it is used in monitoring and forensics. Our work is motivated by the limitations and challenges of using text passwords. Passwords are prone to forgetting and stealing. They often do not fulfill security requirements such as password length, using special characters, and password renewing. On the other hand, using biometrics such as voiceprint overcomes most of these challenges.

In this paper, we can summarize our contribution as follows:

1. We propose a new speaker verification system on mobile devices which uses artificial neural networks in matching speech patterns and recognizing speakers.
2. We present a thorough literature review for similar systems.
3. We have implemented our system on Nexus One mobile device, and have evaluated the performance of our system.

.

RELATED WORK

El-emary et al. (2010) have built voice command system based on hidden markov model and gaussian mixture models using Cepstral coefficients with energy and differentials as features.

Kleynhans and Barnard tested whether using different languages affect the performance of a speaker verification system (Kleynhans and Barnard, 2005). They used a text-independent speaker verification system based on hidden Markov model with 36 dimensional features vector and EM method for training. The test was applied on a multi-language database which represents a huge acoustic corpus of eight different languages (not including Arabic) with a fixed phrase and fixed length of 10 s for each test sample from each language. The error rates varied between languages with Spanish being the best performing language.

Rao et al. (2007) built a text-dependent system for speaker recognition integrated within security access control system. The access control system includes a speech recognition system and a speaker identification system. The hidden Markov model was used to build the system using existing HTK tool for training and testing. Mel-frequency cepstral coefficients were extracted as features from the data set which was collected by recording ten speakers. Each speaker was asked to say any word from a particular Indian language and repeat it 20 times, which was used later as his\her password? In another research, Vaskas et al. (2011) have proposed a new method to optimize the information of an analyzed signal for speaker recognition.

De Lima et al. (2001) aimed to minimize the error rate between false rejection and false acceptance using a text-independent speaker verification system based on Gaussian mixture models. The experiment was done in Brazilian language with a random group of people consisting of 13 males and 26 females. Each person tried to speak continuously without silence for 6 s. The main conclusion was that the higher the number of Gaussians used, the better the results were given enough training time.

Becker et al. (2008) presented a new method for speaker verification based on formant features. After extracting the features, they processed them by using a UBM-GMM verification system. Different feature vectors were used with different dimensions and formant. The best performance was obtained using three formant frequencies.

The Speech Processing Laboratory at National Taiwan University performed tests on three different episodes of a program run by China Central TV (NTUSPL, 2003). This was done to confirm the identity of two people who appeared in these three episodes. Three techniques were used; Eigen voice, Gaussian mixture models and hidden Markov model. Features vectors were extracted using MFCC. The log-likelihood was calculated separately on features vectors for the speaker model and the background speaker model. The performance of HMM was found to be better than the Gaussian Mixture Model, but the GMM can handle a larger training corpus which is more suited for their complicated system. The Eigen voice model performance was not as good as the others.

Olsson used artificial neural networks to estimate hidden Markov model emission posterior probabilities from the speech data in text-dependent systems (Olsson, 2002). They used two common feature parameters: MFCC and LPC coefficients. The Gaussian mixture models gave better results than this HMM/ANN system. Although this system could be further improved by adjusting its many parameters to give an optimal solution. Niesm and Pfister used ANNs to compute the distance between samples in the verification phase of a text-dependent system as an alternative to other distances that are commonly used with DTW (Niesen and Pfister, 2004). The ANN takes a combined pair of feature vectors extracted from the aligned frames of each of the input and stored samples as an input. Features were extracted using the linear prediction cepstral which has 12 coefficients. A fully connected multi-layer perception with hyperbolic tangent activation function ANN was chosen; and the back-propagation algorithm was applied as a training function with adaptive learning rate. For testing

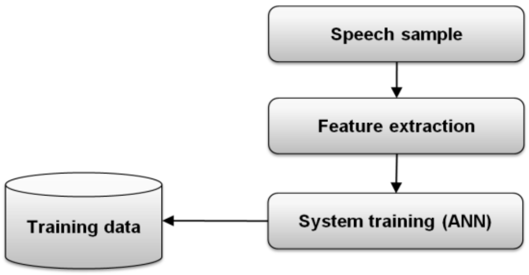

Figure 1a. System flow in enrollment phase.

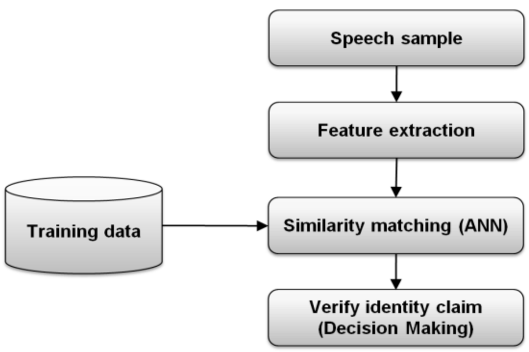

Figure 1b. System flow in verification phase.

and training the system, three hours of samples were obtained from 30 different male speakers. A major improvement in this system is that it does not discover speaker specific features and hence it does not require training for newly input speakers.

Alkanhal et al. (2007) discussed experiments done on an Arabic speaker verification system that uses Gaussian Mixture Models with Saudi accented Arabic telephone speech database (SAAVB); which is a telephony speech database collected over the span of a year. This database contains different speech samples of 1033 Arabic native speakers with Saudi accent collected via mobile service. Different speakers read 59 prompts that contain two different integer numbers twice with two different utterances. The scores of these utterances were then combined in order to reduce error rate. Each frame was processed by using MFCC extraction to get 12 coefficients. Different speakers were asked to say multiple random words through each verification session. For each client and imposter, a score fusion was applied by averaging scores generated by multiple utterances on each single utterance.

THE PROPOSED SYSTEM

We have developed an Arabic speaker verification system with good accuracy which is used in an access control application for mobile devices.

The process of our speaker verification system consists of two phases:

1. The enrollment phase: A speaker *S* repeats a set pass phrase *n* times. The speech signals of the speaker are passed to the neural network as the enrollment data. By adjusting the network, a speaker model for *S* is created. A background model for imposters is created as well; which uses speech signals from a variety of unauthorized speakers.
2. The verification phase: A speaker *X* says the pass phrase and claims to be speaker *S*. The identity claim is then used to fetch the claimed speaker model and background model. The speech features from a spoken utterance are passed as signals alongside the model to a test score evaluation function in the neural network based on how well the utterance fit with the model. The scores are compared with thresholds to accept or reject the claim.

The system flow is shown in Figure 1, which illustrates both enrollment and verification phases and the steps in each phase.

Features extraction

Passing huge speech data as inputs to an algorithm for processing is an exhaustive task with no extra merits for the size of data. In fact, huge data might be an obstacle for enrollment phase which prevents the system from learning, and it is considered one of the main and difficult steps in any learning system which greatly affects the system performance. Since not all data are necessary to derive information needed in recognition, we need a reduced set of these data to pass as feature vectors. This reduction process is called feature extraction.

There are many speech features that are extracted from the acoustic pattern of the speech sample, which is represented as a plot between time on the horizontal axis and loudness on the vertical axis. The features alongside these acoustic patterns differ from one person to another according to the physical characteristics of the airway in the person's vocal tract. To capture these speaker features of a speech represented in waveform, we need to convert it to a type of parametric representation. This parametric representation will be used in the enrollment phase to construct a speaker model to be used in the verification phase to authenticate the speaker. We have used mel-frequency cepstral coefficients (MFCC) as a feature extraction model. We chose MFCC because it is an effective representation (Rao et al., 2007).

Decision making

In general, to confirm if a speech signal *Y* comes from speaker *S*, we need to define the following hypotheses:

1. H_0: *Y* belongs to the speaker *S*.
2. H_1: *Y* does not belong to the speaker *S*.

The speaker verification can be done by applying the following test which is called the likelihood ratio test:

$$\frac{p(Y \mid H_0)}{p(Y \mid H_1)} \begin{cases} \geq \Theta, & \text{accept } H_0 \\ \leq \Theta, & \text{accept } H_1 \end{cases} \tag{1}$$

where $p(Y|H_0)$ and $p(Y|H_1)$ are the likelihoods of the hypotheses H_0 and H_1. Θ is the threshold for accepting or rejecting. In order to implement that, we have used artificial neural network to build and

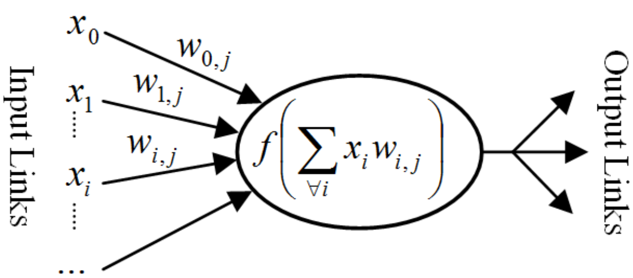

Figure 2. Neuron in neural network.

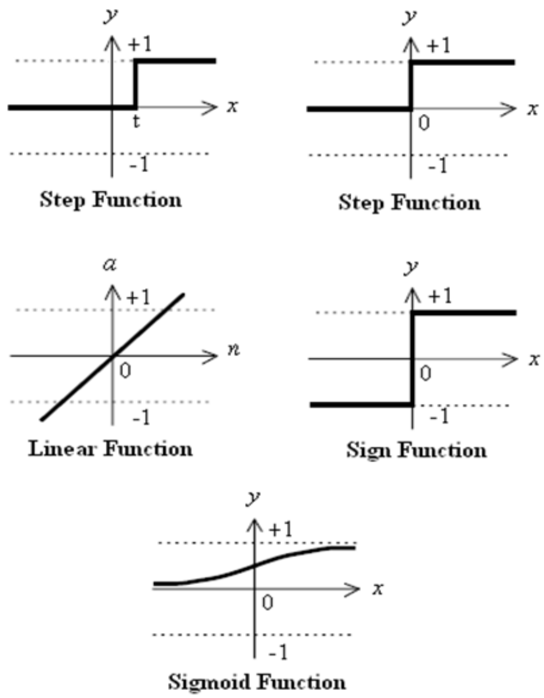

Figure 3. Activation functions in neural network.

match features model.

An artificial neural network (ANN) is a mathematical model that consists of simple interconnected processing units called neurons used to process information (Russell and Norvig, 2003). ANNs are adaptive systems that adjust weights of neuron connections according to knowledge obtained through training. The networks are trained with examples just like children are taught. After the training, ANN may be able to predict and give good approximations to the results of some given data even if it was not included in the training set as a form of generalization. One of the main advantages of using a neural network is its parallelism potential since the computation are mostly independent of each other.

Neural network is a network of neurons connected by directed links, where every $link_{i,j}$ that connects $node_i$ to $node_j$ have a numeric weight $w_{i,j}$, also there is an activation function f associate with every $node_j$. The weight determines how much the input contributes to and affects the result of the activation function as shown in Figure 2. The activation function will be applied to the sum of inputs multiplied by the weights for all incoming links. The activation function - also called transfer function - defines the output of that node given an input or set of inputs. The activation function needs to be nonlinear, otherwise the entire neural network become a simple linear function. Nonlinear activation function allows the neural network to deal with nontrivial problems using a small number of nodes. There are a number of activation functions such as; step function, sign function, and sigmoid function, as illustrate in Figure 3. The sigmoid function is considered the most popular activation function mainly for two reasons - firstly, it is differentiable so it possible to derive a gradient search learning algorithm for networks with multiple layers, and secondly, it is continuous-valued output rather than binary output produced by the hard-limiter.

There are two main types of neural network structures; feedforward networks and recurrent networks. A feedforward network is acyclic network so it has no internal state other than the weights themselves. On the other side, a recurrent network is cyclic network so the network has a state since the outputs are fed back into the network. As a result of that, recurrent networks support short term memory unlike feedforward networks. Feedforward networks are usually arranged in layers, where each neuron receives inputs only from the immediately preceding layer. Feedforward networks can be classified into; single layer feedforward neural networks - also called perceptron - and multilayer feedforward neural networks. For perceptron, all the inputs are connected directly to the output neurons and there is no hidden layer. In this manner, every weight affects only one output neuron. When there is a single output neuron, the network is called single perceptron.

Hidden layers feedforward neural network -- sometime called backpropagation network -- can be used to solve non-linear problems such XOR and many more difficult problems. Hidden layer feedforward neural network currently accounts for 80% of all neural network applications. A simple version of hidden layer feedforward neural network is when a single hidden layer is added and the discrete activation function is replaced with a nonlinear continuous one. The biggest challenge in this type of network was the problem of how to adjust the weights from input to hidden units. Rumelhart and McClelland (1986) have proposed backpropagation learning algorithm, where the errors for the neurons of the hidden layer are determined by back-propagating the errors of the neurons of the output layer. Backpropagation algorithm adjusts the weights in iterative to reduce the error between the output of neural network and the target results. Starting from the output layer and going backward, the weights will be updated as the following:

$$W_{i,j} = W_{i,j} + \alpha \, \Delta_j \, In_i$$

$$\Delta_j = \left(T_j - O_j\right) f'\left(\sum_{k=1}^{n} In_k \, W_{k,j}\right) \quad \text{for output layer}$$

$$\Delta_j = \sum_{k=1}^{m} \Delta_k W_{j,k} \; f'\left(\sum_{l=1}^{n} In_l W_{l,j}\right) \quad \text{for hidden layer}$$

where, α is the learning rate; $W_{i,j}$ is the weight of the *i-th* input for *j-th* node in the layer; In_i is the *i-th* input value for the current node; T_j is the target output; O_j is the actual output and f' is the derivative of the activation function.

Mobile application

The platform for our application is the Android operating system;

Table 1. Hardware specifications and setup.

Parameter	Value
Processor	Qualcomm QSD8250, 1 GHz
Operating system	Android 2.1
Memory	ROM: 512 MB; RAM: 512 MB
Dimensions (L × W × T)	119 × 59.8 × 11.5 mm
Weight	130 grams with battery
Display	3.7-inch AMOLED with 480 × 800 WVGA resolution
Camera	5.0 megapixel color camera
Battery	Rechargeable lithium-ion polymer battery with capacity: 1400 mAh

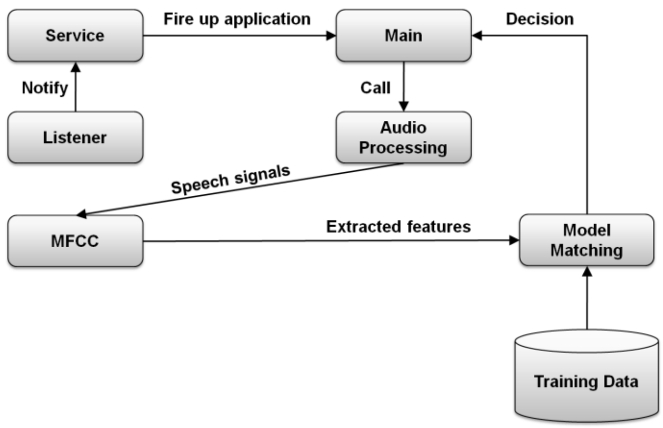

Figure 4. Application system architecture and activity.

which is an open source operating system for mobile devices based on Linux kernel. We designed and tested the application using HTC Nexus One. The hardware specifications and setup are illustrated in Table 1. The application consists of six classes or modules that collaborate to utilize the speaker verification system as follows:

1. Main: Mediates between classes and manages the user interface of the application.
2. Listener: Watches all events occurring in the system. Catches screen locking and unlocking events and signals them to service. Also blocks the buttons when the device is locked.
3. Service: Creates a background running process which receives the unlocking event signal and fires up the application.
4. Audio processing: Prepares the microphones and captures the incoming speech from it. Also adds headers to format raw acoustic data.
5. MFCC: Extracts the speech features from the speech signals.
6. Model matching: Performs the speaker verification using the ANN.

The application design and different modules used with their relationships are shown in Figure 4.

EXPERIMENT

The achieved results for our system were obtained using different configurations. Here, we discuss about samples collection and then present our configurations.

Dataset

All training and testing procedures were done using voice samples recorded from the Android mobile device directly. A corpus of 110 speech samples was collected from 15 different speakers, including people of both genders and of different ages.

The pass phrase chosen was "Naam" ("yes" in Arabic) based on a previous work by Al-Dahri et al. which proved that this phrase is sufficient as a pass phrase to give good results and it has not been affected by the physical condition of the speakers (Al-Dahri et al., 2008).

Configuration

Our system was created using the built-in MATLAB ANN toolbox. Our system was created to accept one speaker. The system was trained, validated and tested using 110 samples: 19 of which are of the accepted speaker saying the pass phrase which should be accepted while 11 samples are random phrases other than the pass phrase; 56 are of random speakers saying the pass phrase; and 16 are of random speakers saying random phrases other than

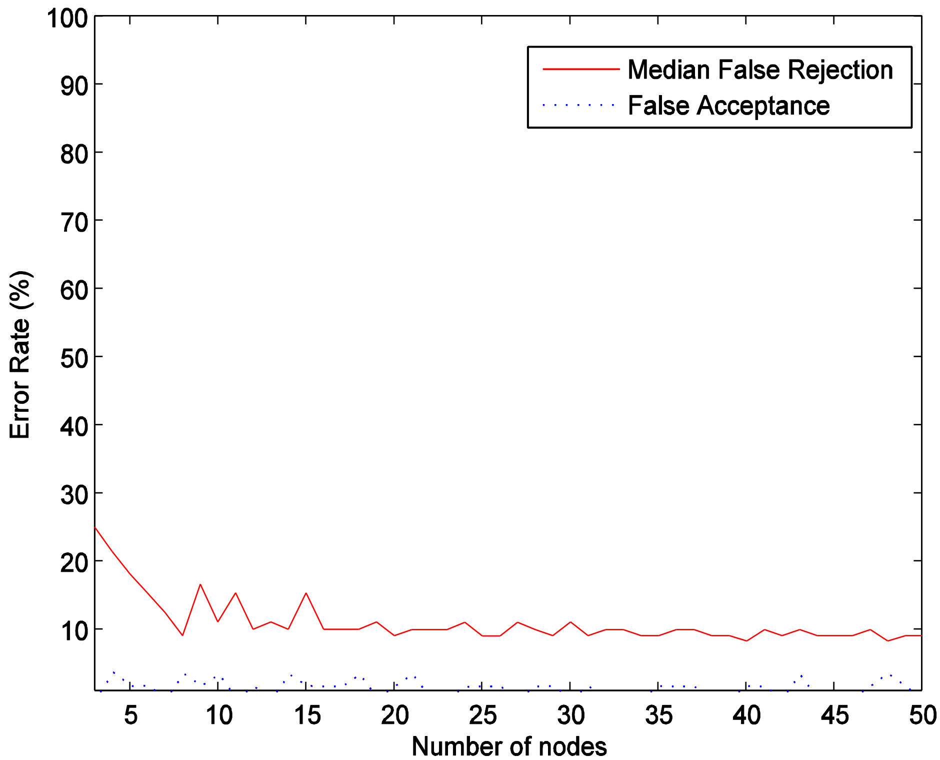

Figure 5a. Comparing median false rejection against the number of nodes for single layer ANN's.

the pass phrases all of which should be rejected. The system was trained using 66 samples, validated using 22 samples, and tested using 22 samples. The selection for the data was done randomly from each sample category. The system was trained to give a value of 1 for acceptance and -1 for rejection. The output is afterwards compared with a threshold for decision making. For all layers, the tan-sigmoid transfer function "tansig" was used.

The batch gradient descent algorithm was used for training the system. The validations samples are used to make sure the network is generalizing and stop training before over fitting. This function was used by G. Venayagamoorthy and N. Sundepersadh (Venayagamoorthy and Sundepersadh, 2000). We have studied the relation between the number of nodes, the threshold and the accuracy in a single hidden layer network; we have also studied the relation between the number of nodes in both layers and the accuracy in a double hidden layers network.

RESULT

In all of the results, each individual test or configuration was run 100 times (trials) since the ANN weighing system depends on random initialization. The medians of these trials were then taken.

The biometric system errors that we used to evaluate the system performance or accuracy are as follows:

1. False rejection rate (FRR): Is the measurement of the likelihood that the system will incorrectly reject an authorized access attempt.
2. False acceptance rate (FAR): Is the measurement of the likelihood that the system will incorrectly accept an unauthorized access attempt.

For the single hidden layer network, we tested the effect of the number of nodes on the accuracy. We also tested the effect of the threshold on the accuracy. We tested 50 different number of node configurations from 1 to 50 and we tested 21 thresholds between -1 and 1 increasing 0.1 at a time. The medians of all thresholds and all nodes are presented in Figure 5. Figure 5a shows the median false rejection rate (FRR) for different numbers of nodes and using a threshold 0. On the other hand, Figure 5b shows the median false rejection rate (FRR) and false acceptance rate (FAR) for different threshold values.

For the double hidden layers network, we tested the effect of the number of nodes in both layers on the accuracy. We tested 50 different numbers of 2nd layer nodes from 1 to 50 and we tested 10 different numbers of nodes for the 1st layer between 5 and 50. The threshold was set to 0. Three of the resulting figures are presented in Figure 6. Table 3 shows the percentage of 100% accuracy (that is. average number of trials that resulted in 100% accuracy) in term of FRR and FAR for both single

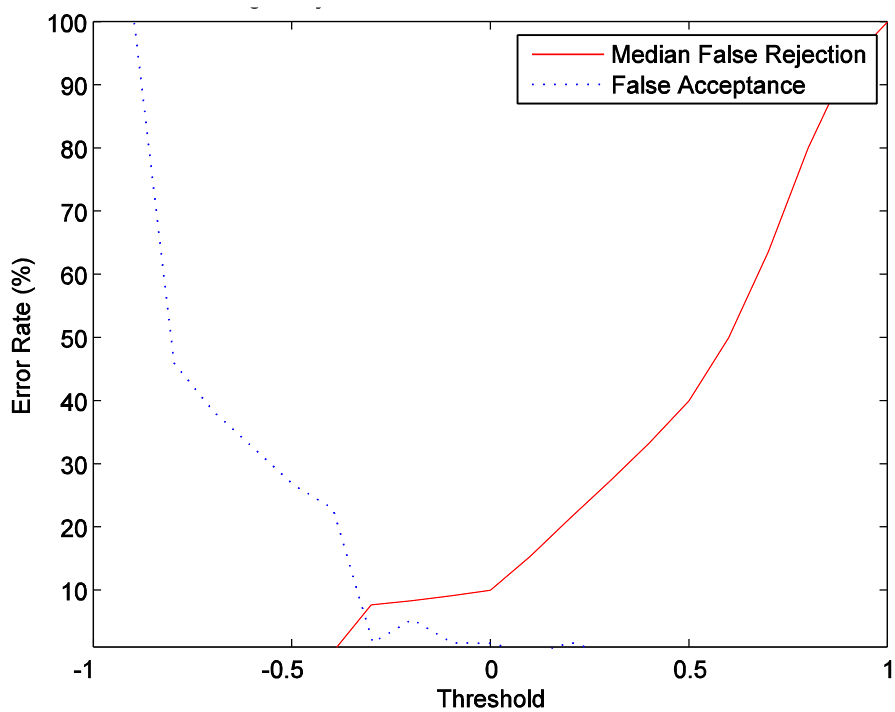

Figure 5b. Comparing median false rejection and false acceptance against the threshold for single layer ANN's.

single and double layers neural networks.

In order to compare our system with different methods, we have implemented the verification system using support vector machine. Support vector machine (SVM) is a statistical based machine learning method originally introduced by Cortes and Vapnik (1995). SVM uses a set of training examples to build a statistical model that is able to classify new examples.

In case of binary labeled data, SVMs separate data classes with a set of hyperplanes that satisfies maximum margin between the two classes. When the data cannot be separated linearly, kernel functions can be used to provide a simple bridge from linearity to non-linearity for algorithms which can be expressed in terms of dot products. In another word, the original data are mapped and transformed using kernel function into linearly separable data. There are many kernel functions, however we have used and tested three kernel functions:

1. Linear kernel function;
2. Quadratic kernel function;
3. Polynomial kernel function with order 3.

Table 2 shows ANN and SVM configurations that have been used in our experiments. The same data set and samples have been used in both ANN and SVM. For each configuration, we run the experiment 100 times (trails) and recorded all results. The average numbers of trails that resulted in 100% accuracy for every configuration are shown in Table 3. The table shows that ANN results in more trails that have 100% accuracy in term of FRR and FAR. Furthermore, all ANN models have smaller median FRR and FAR than any SVM models in our experiments which is shown clearly in Figure 7.

The results show that our system using ANN outperforms the same system using SVM. It appears that much better results in classification were obtained using ANN than SVM. ANNs are very efficient tools for speaker verification and they can be successfully used in accessing control applications.

CONCLUSION AND FUTURE WORK

In this work, we proposed and developed a new Arabic text-dependent speaker verification system for mobile devices based on artificial neural networks. To the best of our knowledge, this is the first Arabic speaker recognition system on mobile devices. Our results have shown that our system is applicable as an alternative access control for mobile devices.

We intend to improve our results by finding the optimal configuration for the neural network. Also, we intend to try different features such as LPC coefficients or even different feature matching models such as recurrent neural

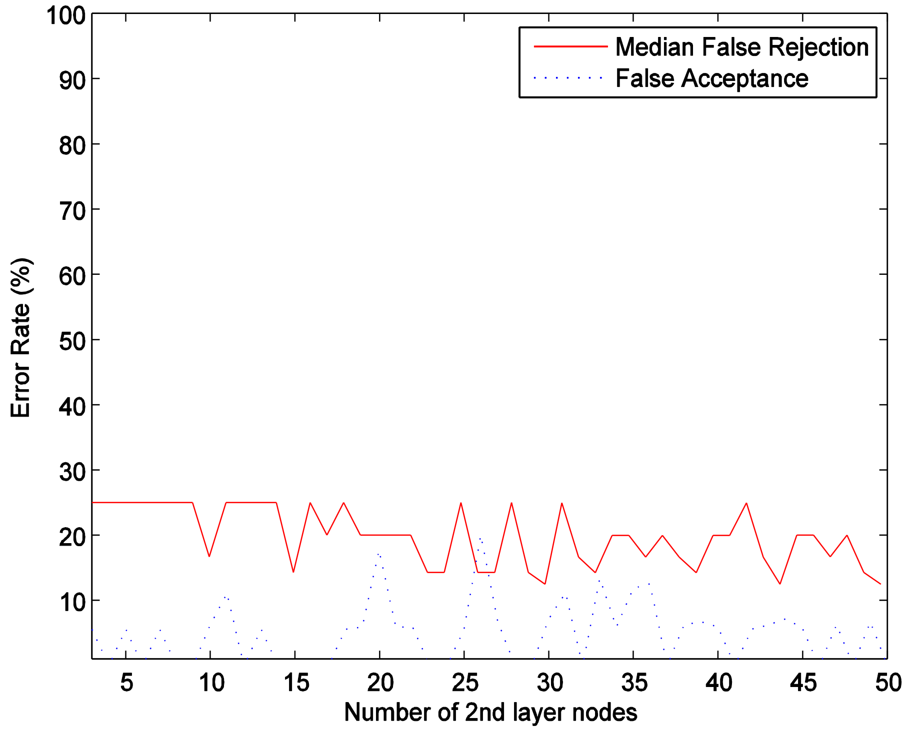

Figure 6a. The performance of double layers ANN using 10 nodes in the first layer and different numbers of nodes in the second layer.

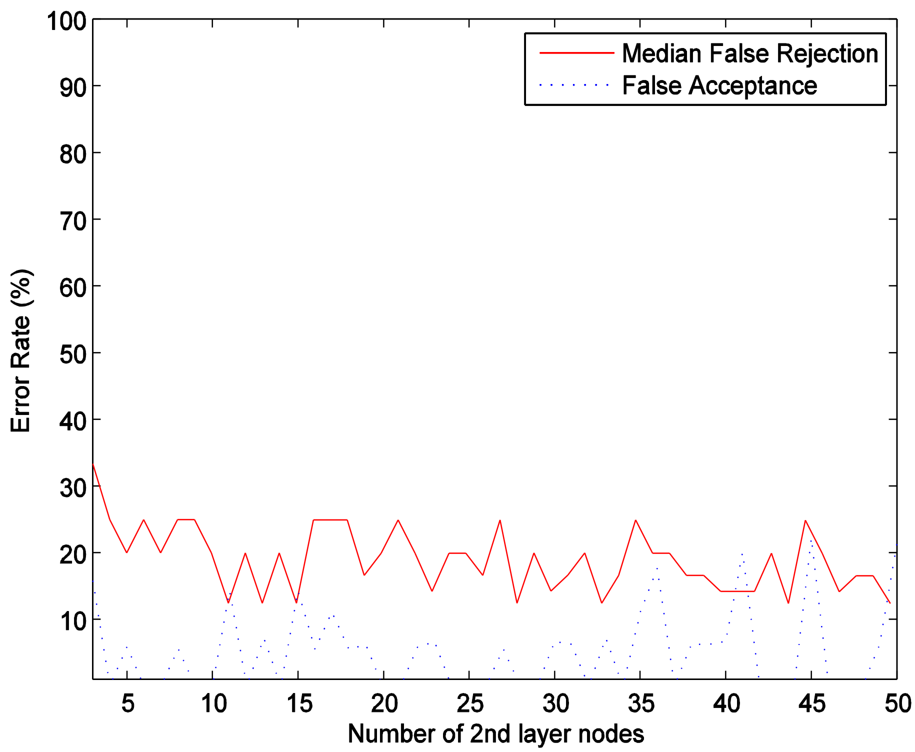

Figure 6b. The performance of double layers ANN using 20 nodes in the first layer and different numbers of nodes in the second layer.

Table 2. Configuration parameters.

Model	ANN	SVM
Tool	MATLAB ANN Toolbox	MATLAB Bioinformatics Toolbox
Samples	110 samples: (accept) 19 accepted speaker saying pass phrase (reject) 11 accepted speaker saying random phrases (reject) 56 random speakers saying pass phrase (reject) 16 random speakers saying random phrases	
Samples distribution	66 training samples 22 validating samples 22 testing sample	65 training samples 45 testing samples
Model configurations	Single layer (1:50 nodes) Double layer (1:50 × 1:50 nodes) Transfer function: Tan-sigmoid	Linear kernel Quadratic kernel Gaussian radial basis function kernel (σ = 1) Polynomial kernel (order: 3)
Training algorithm	Batch gradient descent algorithm	Quadratic programming (two-norm, soft-margin SVM)

Table 3. Average number of trials that resulted in 100% accuracy.

Configuration		Trials with false acceptance rate = 0 (%)	Trials with false rejection rate = 0 (%)
ANN	Single layer	60	29
	Double layers (1st layer 10 nodes)	51	35
	Double layers (1st layer 20 nodes)	47	38
	Double layers (1st layer 30 nodes)	51	37
SVM	Linear	32	11
	Polynomial	37	05
	Quadratic	23	01

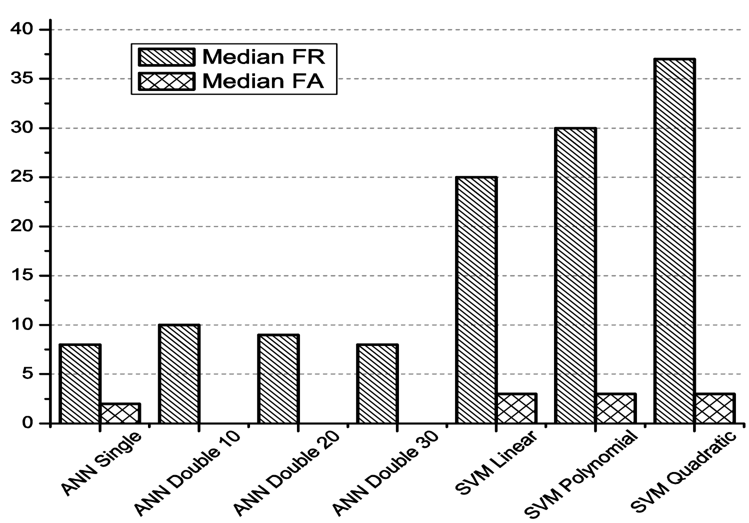

Figure 7. The performance (Median FR and median FA) of ANN vs SVM.

network or hidden Markov model and study the differences between methodologies. Future work may also consider a text-independent version of our speaker verification system.

REFERENCES

Al-Dahri S, Al-Jassar Y, Alotaibi Y, Alsulaiman M, Al-Mamun K (2008). A word-dependent automatic Arabic speaker identification system. IEEE Int. Symp. Signal Proc. Inf. Tech., pp. 198–202.

Alghamdi M (1997). Speaker's identification 'voice print'. King Khalid Military Academy Quarterly, 14(54): 24-28.

Alkanhal M, Alghamdi M, Muzaffar Z (2007). Speaker verification based on Saudi accented Arabic database. International Symposium on Signal Processing and its Applications, Sharjah, United Arab Emirates, pp. 1-4.

Becker T, Jessen M, Grigoras C (2008). Forensic speaker verification using formant features and Gaussian mixture models. Proceedings of INTERSPEECH, Brisbane, Australia, pp. 1505-1508.

Bowyer KW, Hollingsworth K, Flynn PJ (2008). Image understanding for iris biometrics: A survey. Comp. Vis. Image Underst., 110(2): 281-307.

Campbell J (1997). Speaker recognition: a tutorial. Proceedings of the IEEE, 85(9): 1437-1462.

Campbell W, Sturim D, Reynolds D (2006). Support vector machines using GMM supervectors for speaker verification. IEEE Signal Proc. Lett., 13(5): 308-311.

Cortes C, Vapnik V (1995). Support-vector networks. Machine Learning. 20(3): 273-297.

de Lima C, Alcaim A, Apolinario J (2001). Text independent speaker verification using GMM. IEEE South-American Workshop on Circuits and Systems, pp. 1-4.

El-emary I, Fezari M, Attoui H (2010). Hidden Markov model/Gaussian mixture models (HMM/GMM) based voice command system: A way to improve the control of remotely operated robot arm TR45. Sci. Res. Eassys, 6(2): 341-350.

Jain A, Ross A, Pankanti S (2006). Biometrics: a tool for information security. IEEE Trans. Inform. Forensics Security, 1(2): 125-143.

Jain A, Ross A, Prabhakar S (2004). An introduction to biometric recognition. IEEE Trans. Circuits Systems Video Technol., 14(1): 4-20.

Kaji Y, Moriyama T, Ushida T (1997). Recognition on voice-prints of vowels using neural network. Proceedings of the 36th SICE Annual Conference, Guimarães, Portugal, pp. 1151-1154.

Kleynhans NT, Barnard E (2005). Language dependence in multilingual speaker verification. Proceedings of the 16th Annual Symposium of the Pattern Recognition Association of South Africa, Langebaan, South Africa, pp. 117-112.

Kung S, Lin S (2005). Biometric Authentication: A Machine Learning Approach. Prentice Hall, pp. 250-255.

Niesen U, Pfister B (2004). Speaker verification by means of ANNs. Proceedings of ESANN, Bruges, Belgium, pp. 145-150.

NTUSPL (2003). Report on speaker verification. National Taiwan University Speech Processing Laboratory, Taiwan.

Olsson J (2002). Text dependent speaker verification with a hybrid HMM/ANN system. Thesis Project in Speech Technology, Department of Speech, Music and Hearing, Royal Institute of Technology, Stockholm.

Ramli D, Samad S, Hussain A (2007). Preprocessing techniques for voice-print analysis for speaker recognition. Proceedings of the 5th Student Conference on Research and Development, Permati Bangi, Malaysia, pp. 1–5.

Rao RR, Nagesh A, Prasad K, Babu KE (2007). Text-dependent speaker recognition system for Indian languages. Int. J. Comp. Sci. Netw. Security, 7(11): 65-71.

Rumelhart D, McClelland J (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations I. MIT Press, pp. 318–362.

Russell SJ, Norvig P (2003). Artificial Intelligence: A Modern Approach. Pearson Education, pp. 736-748.

Uludag U, Pankanti S, Prabhakar S, Jain A (2004). Biometric cryptosystems issues and challenges. Proceedings of the IEEE. 92(6): 948-960.

Vaskas A, Shamshirband S, Gholami M, Besheli M (2011). Information optimization for speaker recognition using correlation functions. Int. J. Phys. Sci., 6(14): 3398-3408.

Venayagamoorthy G, Sundepersadh N (2000). Comparison of text dependent speaker identification methods for short distance telephone lines using artificial neural networks. Proceedings Int. Joint Conf. Neural Networks Como Italy, pp. 253-258.