

**E-Hafiz: Intelligent System to Help Muslims in Recitation and Memorization of Quran**Aslam Muhammad<sup>1</sup>, Zia ul Qayyum<sup>2</sup>, Waqar Mirza M.<sup>3</sup> Saad Tanveer<sup>4</sup>, Martinez-Enriquez A.M.<sup>5</sup>, Afraz Z. Syed<sup>6</sup><sup>1,3,4,6</sup> Dept. of CS & E, University of Engineering & Technology, Lahore<sup>4</sup>Department of Computing and Technology, IQRA University, Islamabad, Pakistan<sup>5</sup>Dept. of CS, CINVESTAV-IPN, D.F. Mexico(Corresponding author email: [maslam@uet.edu.pk](mailto:maslam@uet.edu.pk))

**Abstract:** Recitation of the Holy book of Muslims, The Holy Quran, is a regions duty and hence is done with utmost care such that no mistakes are made while reading it. These mistakes may include the wrong utterance of words, misreading words, and punctuation and pronunciation mistakes. Believers of Islam are spread all over the world and hence there also can be a difference in accent. To avoid this, tajweed rules are implemented to ensure that the utterance is done according to some rules. These rules ensure that there is no variance in the recitation of the Holy book for different reciters. For further improvements, the people are encouraged to memorize the whole book and the person who does that is called a Hafiz. Having known the whole book by heart down to every word with Tajweed rules he/she can a guide to correct other learners who intend to learn by listening to learners and correcting their recitation. But the availability of a Hafiz can be a problem where Islam is not a dominant religion. Furthermore the competency and level of expertise are of epic importance. To get around this problem we have designed and developed a system E-Hafiz. It is based on an idea that Tajweed rules are used to train learners how to recite Quran. To achieve this we used on Mel-Frequency Cepstral Coefficient (MFCC) technique. We extract the features of recorded voices using MFCC and compared with experts' voices stored in database. Any mismatch on word level is pointed out and ask the user to correct it.

[Aslam Muhammad, Zia ul Qayyum, Waqar Mirza M. Saad Tanveer, Martinez-Enriquez A.M., Afraz Z. Syed: **E-Hafiz: Intelligent System to Help Muslims in Recitation and Memorization of Quran**. Life Science Journal. 2012;9(1):534-541] (ISSN:1097-8135). <http://www.lifesciencesite.com>. 80

**Keywords:** The Holy Quran, Islam, Muslims, Tajweed, Hafiz, Voice recognition, feature extraction, MFCC.

**1. Introduction**

As Islam is the second largest religion of world's population, and their Holy book being in Arabic with only 3.12% of the world's population speaking Arabic (<https://www.cia.gov>). The adherents of Islam are 1.57 billion in 200 countries all over the world in the year 2009 (<http://pewforum.org>). Such diverse geographical distribution leads to the fact that most of the Muslim population does not have Arabic as their primary language. Regarding to problems in understanding and reading Arabic, the religion binds Muslims strongly to Arabic as their Holy book is in Arabic. The Holy Quran also gives an insight in science, engineering, social sciences, law and management (Abdul Rashid Sheikh, 2000). This fact encourages people to read and understand the Quran for both religious and scientific purposes. The Holy Quran was compiled as book nearly 1400 years ago. During this time Arabic as all other languages evolved and underwent changes. So the text reading is not same as Arabic we know today. To solve this barrier, some rules are implemented to ensure the correct reading of the Holy book. These rules have come to be known as Tajweed rules (H. Tabbal et al., 2006) (M.S. Bashir et al.).

The recitation of Holy Quran by use of Tajweed rules is an art and reciters follow the

Tajweed rules to build their recitation attractive (H. Tabbal et al., 2006). If someone wants to learn the Tajweed then he/she has to consult an expert Hafiz at some Learning institute. Learning is done manually as the student and Hafiz have to sit face to face and the learner recites and the Experts points out and corrects the mistakes if any occur. Presently, there exist institutes and websites that teach the recitation in manual method. The main problem consists here is the accessibility and feasibility of a Hafiz expert. Availability of Hafiz in countries where Islam is not a dominant religion can be quite less to come across. In addition, Hafiz's level of expertise is a big issue. As a human being a Hafiz can also make mistakes while listening, so not all Hafiz can be tutors.

The goal or objective for the development of E-Hafiz system is to facilitate the learning/memorizing of the holy Quran, minimizing errors or mistakes of all kinds, and the systematization of the recitation process. Using this system any reciter can learn the recitation skills at any place and any time. The presence of an expert hafiz would not be needed. Consequently, this system helps the Hafizes in preparation of recitation for 5 times prayers and Traveeh prayer in the month of Ramdhan.

Many audio enabled applications are available in the markets which offer the Holy Quran as audio streams. One of the most popular and commonly used is the Quran Auto Reciter (QAR) (<http://www.searchtruth.com/download.php>). QAR provides a user interface where user can select the verse they intend to listen and the verse can be played stopped and paused. And as the words are played the text also gets highlighted. QAR can help learn the Quran and recitation improvements can be made as well as some basics can be learned. But it cannot judge the user's accuracy and performance as there is no way QAR can indicate or detect the mistakes made by user. If software has the utility to detect and correct the mistake the learning and expertise can be improved substantially. So, the learning process through QAR is one only sided. A user can only listen his desired surah many times to improve recitation ability however he cannot know about the mistakes if any he made during his own recitation. Therefore, to solve this problem, he must consult to a hafiz/expert of Quran to know either he recite the holy Quran correct or make any mistake. So, reliability on other person is still exists in this system. Hence this system fulfill required objective in limited area not completely.

(Hassan Tabbal et al., 2006) introduce an automated delimiter that can extract the verses from the audio file using the open source Sphinx framework and speech recognition techniques. In whole processing, two models acoustic and language are discussed. These models takes parts with feature vectors to generate search space for HMM nodes. These two models are:

**Acoustic Model:** A set of phonemes symbols was used to train the state of HMM that corresponds to the acoustic model. For generating the Acoustic model for the application, the audio recitation of surah Al-Ikhlass (The Holy Quran) (recited by different reciters near about 1 hour) alongside with corresponding dictionary mapping is feed to the sphinxTrain application that generate acoustic representation of each word.

**Language Model:** It was not easy to choose a language model for the holy Quran, as the high precision and accuracy required for recognition. So, for this system, a language model based on the Java Speech Grammar Format (JSGF) specification was chosen that is compatible with sphinx framework and fulfills all the requirements of the system. The JSGF rules used for this system are similar to those used for conversational system and generated them as such to reflect the structure of the Surah.

The sphinx frame work automatically provides the core recognition process by using appropriate language and acoustic models. For this

purpose, the sphinx frame work is configured by an xml based configuration file that includes the feature extraction algorithms and all other aspects that needed by any speech recognition systems. The system design phase is divided into two sub phases.

**Data Preparation:** The frame of 10ms and a threshold of 10db were selected for speech segment extractor. To make the recognition ratio more accurate, a 2-stage pre-emphasis filter with factor values (0.92 and 0.97) is used. A raised cosine widower with the 512 points FFT analysis is used for the system and a Mel Filter Bank followed by a Discrete Cosine transformation is used for extracting MFCC features. MFCC have the ability to transform the frequency from a linear scale to a non linear one which can be achieved by this system by using set of 30 triangular Mel filters. Finally, to reduce the distortion effects produced by the microphone, the Cepstral Mean Normalization (CMN) operation is performed. The CMN is achieved by subtracting the mean vector from each vector.

**System Settings:** The frontend phase output is feed to sphinx core recognizer which uses HMM as a recognizer tool. To translate the result of recognizer into common Arabic language, a hash map was used. The breath first search combined with beam search algorithm is used by the decoder to search to the same words obtained from the recognizer. When the words matched with the stored words, the audio verse correspond to the obtained combination is extracted.

As we have seen, the above discussed application can help users to search a required verse form audio files but unfortunately it is not useful for the users who want to learn the recitation. This application is only used by those persons who know the recitation skill well but not for those who not know how to read the holy Quran. Fortunately, this application helped us very much in our research as it also uses MFCC feature extraction techniques for its implementation. So, the brief study of this application helps us a lot in establishing our ideas and solves the problems we faced.

(Zaidi Razzak et al., 2008) wrote a review paper presents techniques used in Quran Arabic verse recitation recognition and also mention there advantages and disadvantages by comparing them. The objective of this research is to found more effective and efficient technique of Quran Arabic verse recitation recognition for their system that will used to support in j-QAF learning process. According to the paper, process of recitation recognition is commonly divided into: pre-processing, feature extraction, training and testing, feature classification, as well as pattern recognition.

**Preprocessing:** In Pre-processing the information is organized to simplify the task of recognition. Three steps are performed: -End Point Detection specifies the start and end points of recorded words. -Smoothing reduces noise from the speech. -Channel normalization used to train a recognizer with recorded speeches. The recognition process is depending on speeches recorded from different microphones.

**Feature extraction:** To differentiate words, the unique, discriminative, and computation efficient features are extracted from the speech signal. Four techniques are treated: (a) Linear Predictive Coding (LPC) that is not considered as a good method, since LPC reduces high and low order Cepstral coefficient into noise when coefficient are transferred into Cepstral coefficient, (b) Perceptual Linear Prediction (PLP) that is better than LPC, since the spectral features remains smooth within the frequency band in PLP and the spectral scale is non-linear Bark scale, (c) Mel-Frequency Cepstral Coefficient (MFCC), based on the frequency domain of Mel scale for human ear scale. MFCC is considered the best technique because behavior of acoustic system remains unchanged during transferring the frequency from linear to non-linear scale. (d) Spectrographic Analysis is used for Arabic language phoneme identification. Arabic phonemes are identified by spectrograms that are represented by distinct bands.

**Training and Testing:** Speech sample is enrolled in the system database after constructing a model based on features extracted from the speech. Testing process determines similarity between score of newly speech word with the speech stored in DB. Three training and testing methods discussed are: a) Hidden Markov Model (HMM) in which each word is trained independently to get the best likelihood parameters and for this several utterances is performed to train each set of model. b) Artificial Neural Network (ANN) is a based mathematical model, that recognizes speech in such a way that a person applies to visualizing, analyzing, and characterizing the speech to measure its acoustic features. Here, we show that ANN is not well equipped with respect to HMM that solve problems. c) Vector Quantization (VQ) that uses a set of fixed prototype vectors called codebook and each vector in codebook is known as codeword. Quantization is performed by matching input vector against each codeword using distortion measure. d) Features Classification and Pattern Recognition that classify the object of interest into classes. The goal is to know patterns and classes referred to individual words. There are three methods for this purpose: HMM, VQ, and ANN. This process is also referred as feature matching. The author recommends HMM the

best approach for feature extraction and HMM or VQ is for training and testing. HMM is used when Arabic language recognition has to perform and VQ for English language.

There are many different methods suggested by (Zaidi Razzak et al., 2008) in to extract the features from the speech, the system suggests rules and regulations that should be followed during recitation. Again the basic learner is ignored. The system is useful for people who already know the correct pronunciation and Holy Quran rules. But, it is not suitable for non Arabic speakers.

So, a system to help naïve learners to recite the Holy book but also is effective for expert users to know Tajweed rules, pointing out mistakes made during recitation is suitable, tasks achieved by our E-Hafiz system.

We designed, implemented, and tested E-Hafiz application that helps learning like a Hafiz expert. Speeches signals will be gained by a sound speak by a person in microphone. By means of Mel-Frequency Cepstral Coefficient (MFCC) (Noor Jamaliah Ibrahim et al., 2008) transformation, voice features are extracted from the signal emphasized for further processing. MFCC transformation technique produces remarkable results, because the emulation of an auditory system behavior. The linear scale frequency is transformed to non linear one. MFCC is implemented by use of MATLAB framework. The extracted features are used to form a model of speech by use of Vector Quantization (VQ), and are stored in the Database which also contains a large number of speech vectors, obtained from different Quranic verses passed through the above process. Basically, the speech vector is an array of MFCC features. When user utters any verse, it is compared with the stored verse. Verses that are not matched with any registered one are considered as mistakes and pointed to user.

## 2. Material and Methods

Voice content matching is the process of comparing the voice content of a speaker with the relevant voices contents stored in database of system and make decision on the bases of this compression. At abstract level the content matching system has two phases: *Training phase* and *Testing phase*. During the Training phase, the system is trained with the experts' voices and during testing phase a user records his voice and this voice is matched with the experts' voices to generate results.

If we analyses the current method of teaching in existing organizations/institutes, we can find that most of the Islamic institutes follow the manual method of teaching recitations. In manual process, the teacher and student sit in front of each

other and student starts recitation. Whenever student makes mistake, teacher point it out and correct it like a real time system. The developed system does not work in this real time mode, but it can help the user to know his mistakes after he complete his recitation. The reason behind the development of this non-real time/offline mode system is that at initial level we have some kind of issue which we cannot resolve during user recitation like removing silence from the voice and filter out signals etc.

The figure 1 shows the core architecture of E-Hafiz system. In this system, the MFCC feature extraction techniques is used to get the feature vectors of some specific verses read by some experts and store in the system's database. In order to test the performance of the system, the voice sample of same verses read by ordinary persons are taken and the feature of these voices are taken too by using MFCC techniques. Then, these vectors are compare with the vectors stored in system's database to detect the mistakes if exists. The current system has the ability to detect mistakes on word level as the back end database is made at word level. So, any mistake made at word level can be identified. This is done by a word extraction module which extracts the words out the audio stream recorded. Whenever, a mistake is found the system gives option to the user to listen the verse again as in real life a Hafiz does. The user listens to the verse again to comprehend the right pronunciation and after that he again reads it until he passes the minimal criteria .This criteria however, provides more flexibility especially in Tajweed rules for beginners. But, they are stricter for experts. The dataset of E-Hafiz consists of 10 experts with database of first 5 Sorahs of the Holy Quran.

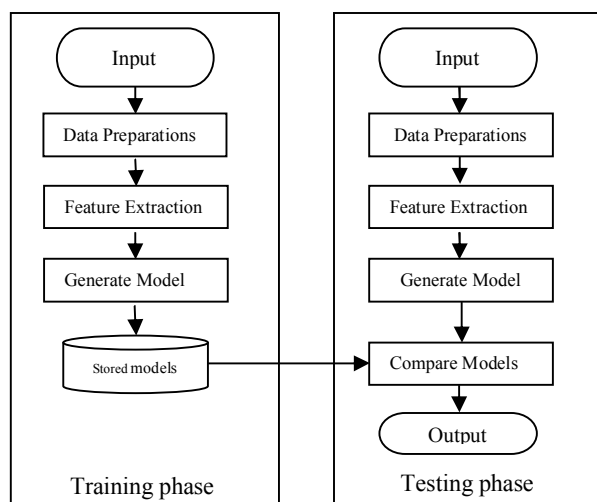


Figure 1. Architectural model of E-Hafiz.

The principal phases in E-Hafiz architecture are: data preparation, feature extraction, and

modeling, storing and comparison.

**Data Preparation:** In data preparation phase, the raw data (input speech signal) contains silences and noises, are filtered out in order to avoid the errors which may occur during processing and disturb the accuracy of results. The data preparation is subdivide into three steps:

*Silence Trimming:* The first verse of Surah Al- Fatiha (sarah-1, verse-1) (The Holy Quran) of holy Quran is uttered as an input audio. The recorder audio signal of this verse is shown in figure 2. As it is widely possible that at start time and end times in which the user may have recorded his/her recitation may contain long or short silence gaps. A module trims these silence gaps. This gives a content rich stream for processing. In order to remove silences trimming, the short-term energy method (Mark Greenwood *et al.*, 1999) is used. In short-term energy method, the energy of a speech signal can be calculated at any instance of time. So, the energy of each frame is calculated and removed all those frames whose energy is near about to zero.

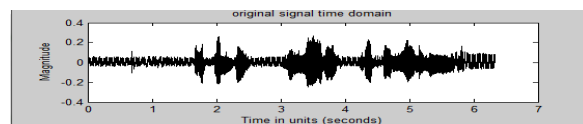


Figure 2. Audio signal of 1st verse of Surah "Al-Fatiha"

*Word Extraction:* The next module separates the words that are content of the audio stream on the basis of silence threshold that enables the system to store all words uttered separately into the database. This is done to ensure that we can match the error on word level as the comparison done is not on the whole audio stream but on the words that have been uttered.

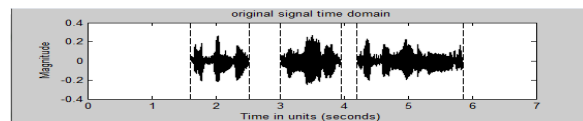


Figure 3. Audio signal of 1st verse of Surah "Al-Fatiha" after extracting words

The detection of words is also performed using short-term energy method (Mark Greenwood *et al.*, 1999). Here, we calculate the energy of each frame and whenever a set of 16 frames having energy approaches to zero is found it is consider an end of word and all the consequent zero energy frames after that are removed. On moving forward where the set of 16 frame having energy greater than threshold founds, it is a sign of starting next word and all the consequent frames are included until again found zero energy frames. This process continues till the



end of speech and finally we got all the available words in the speech (see Figure 3). Once words are extracted each of them undergoes these following steps.

*Pre Emphasis:* After extracting all available words, the next step of data preparation is the pre emphasis each sub-signal/word, giving raise to higher magnitude frequencies with respect to lower in order to improve the signal to noise ratio. The echoes lies in the signal are also eliminated and due to this quality this process is also known as noise canceling filter. To perform the pre-emphasis, the application of first order Finite Impulse Response (FIR) filter is imply on digitized signal. The equation used to apply FIR filter is:

$$H(z) = 1 - \alpha z^{-1} \quad (1)$$

Where  $\alpha$  represents the pre-emphasis parameter, which may have value close to 1, in our case 0.935, it gives rise to the high frequency spectrum to more than 18dB amplification.

**Feature Extraction:** The voice features are extracted by means of Mel-Frequency Cepstral Coefficient (MFCC) (Noor Jamaliah Ibrahim *et al.*, 2008) transformation for further analysis purpose. MFCC is one of the best techniques used for feature extraction purpose that produces remarkable results in the field of voice content matching systems. It is because, it simulate the behavior of human ear and uses Mel Frequency scale. Here the Feature Extraction technique is performed on each extracted word of speech one after another to get their codebooks. The MFCC feature extraction technique consists of seven major components which are:

*Pre-Processing:* This is the process in which we prepare data for processing and make it ready to operate. In section 3.1, some necessary initial processing is performed. So, our date is already prepared and no further operation is needed.

*Framing:* Each word's signal is segmented into 23 mS frames to convert non-stationary signals into qui-stationary format. Also, theses frames are further blocked overlap i.e. every frame contains the 11.5 mS of its previous frame's data. The overlapping is performed to reduce the chances of losing information lies at the end of each frame, which may crash during segmenting of speech.

*Windowing:* Performing windowing means, multiplying each frame with Hamming Window. Windowing shrink the signal values toward zero level at the boundaries of each frame and hence, it reduces the discontinuity. The Hamming Window is obtained by:

$$w_n = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \quad (2)$$

In this equation  $N$  is the total number of

samples in each frame and  $n$  is any value from range 0 to  $N-1$ . Windowing is performed by multiplying each sample of each frame with each corresponding element of hamming window.

*Discrete Fourier Transformation:* In order to transfer each windowed frame from time domain to frequency domain, Discrete Fourier Transformation is applied by use of FFT algorithm. The windowed signal obtained from previous step, is given as an input to DFT and the output of this is a complex number, representing each frequency band (0 to  $N-1$ ) having magnitude and phase of that frequency component in original signal. The DFT is obtained by the following equation:

$$Y_2[n] = \sum_{k=0}^{N-1} Y_1[k] e^{-2\pi jkn/N} \quad (3)$$

Where  $k=0, 1, 2, 3, \dots, N-1$  and  $Y_2[n]$  is the Fourier Transform of  $Y_1[k]$ .

*Mel Filter Bank:* Often low frequencies in speech signal contain more useful and important information as compare to higher ones. So, to emphasize these low frequency components, Mel scale is applied. The formula used to calculate Mels for a frequency  $f$  in Hz is:

$$\text{Frequency (Mel Scale)} = 2592 * \log_{10}\left(1 + \frac{f}{700}\right) \quad (4)$$

*Logarithm:* By use of logarithm, the multiplication effect of the magnitude of Fourier Transform is changed into addition. The Matlab command *log* is used to take the natural log of the Mel filtered speech segments. The effect of taking natural log is that it reduces the values of Mel filter bank.

*Inverse Discrete Fourier Transformation:* IDFT converts the speech signal back to time domain from frequency. As an output we will finally get the Mel Frequency Cepstrum Coefficients (MFCC) in the form of vector known as features vector. The equation used to get MFCCs is:

$$Y[n] = \sum_{k=0}^{N-1} x[k] \cos\left[n\left(k - \frac{1}{2}\right)\frac{\pi}{N}\right], \quad n = 1, 2, 3, \dots, L \quad (5)$$

Here  $x[k]$  is the logged value of each Mel filtered speech segment gained from previous step.  $L$  is the required number of Mel Cepstral Coefficient taken from  $N$  filtertapes of each frame and in our case  $L$  is 12.

### Modeling, Storing and Comparison of Codebooks:

The output of second phase is used to generate model (a features codebook) of speech, and is stored in the Database. Basically, the feature vector is an array of MFCC features. As the speech have a large number of frames and each contains 12 feature vectors. So it is not easy to use all these feature vectors to form codebook. Hence, the numbers of features vectors are reduced by getting highly representative vectors

which is achieved by Vector Quantization (VQ) technique (R. M. Gray *et al.*, 1984).

Vector Quantization is the data compression technique in which probability density functions are modeled by the distribution of prototype vectors. In this technique a large set of vectors having similar number of points are grouped together and these group are represented by their centroid point gained through clustering algorithm. In this system LBG algorithm is used to implement VQ. LBG algorithm works by clustering the similar vectors and finding a single representative value (centroid) for each cluster. This centroid value is also called a code vector and collection of all code vectors correspond to a specific voice is called codebook.

For generating codebook, Mean is calculated for all features vector. So, to calculate the mean of a set of K vectors, the following formula is used:

$$M = \frac{\sum_{i=1}^k X_i}{k}, \quad (6)$$

Here we have a single Mean value representing the whole data. Now this Mean is split into two Means and for this, a very small positive number let say  $\varepsilon$  (read as epsilon) is used as follow:

$$M_1 = M + \varepsilon \quad (7)$$

$$M_2 = M - \varepsilon \quad (8)$$

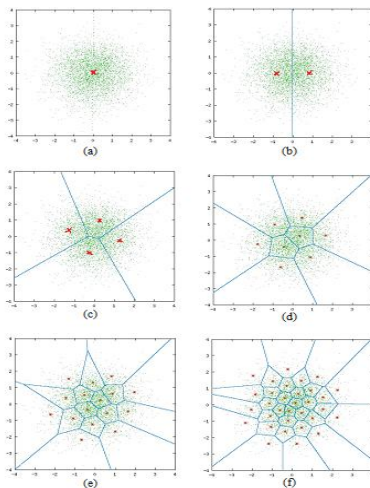


Figure 4. Process of VQ Codebook Generation

Now by use of these two Means two clusters of features i.e. Cluster1 contains all feature vectors having values close to  $M_1$  and Cluster2 contains all feature vectors having values close to  $M_2$ , are created. In order to generate clusters of vectors, the distance between each feature vector and Means values is calculated through Euclidean Distance Formula. Now these two clusters are further divided into four clusters according to method discussed above. This iterative process will continue till 32

clusters are created as shown in figure 4. In every cluster there is a mean value called the code vector of that cluster and all code vectors in all clusters is called a codebook.

Now for testing the system, an interface is given to user to select the Sorah and verse she/he intends to recite as shown in figure 5(a). The system also gives him the option to select his expertise level. On this expertise level, the system decides that with how many experts' voices, his voice is compare. For beginner, the voice of user is compare with all 10 experts' voices.

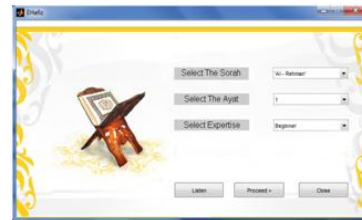


Figure 5(a). Main screen of E-Hafiz application

When user utters any verse, it is compared with the stored verses read by number of different experts; this number however depends on the difficulty level selected. The comparison is performed on word level and in case of any word not matched with any registered one is considered as mistakes and pointed to user. The whole process is as follow:



Figure 5(b). Result screen of E-Hafiz application

1. Gets the utterance of a selected verse recited by any user.
2. Extract the words from the voice sample taken.
3. Extract features of each word using MFCC technique discussed above.
4. Generate the codebook of these words and form an array represents whole verse.
5. From system's database, extract the codebooks array of the same verse recited by experts.
6. For each word of Expert's codebook array and user's codebook array, compute their averages and calculate distance between them.
7. The resultant distance value is compared with the Threshold value. If the distance is less than the threshold value (Dependant on the skill level of user value is set to 2.6 for beginners) it is considered matched, else mismatch word.
8. If all the words of one expert's array are

matched with the user's array, also the number of word in both arrays and their sequence is same then 1 match is considered.

9. In case of beginners at least 3 matches must be found and if not it is consider wrong utterance and pointed to user.
10. In case of wrong utterance result, the word which is found most mismatch is consider as wrong uttered word and that word is highlighted in result screen as shown in figure 7(b).

In result screen option given to user to listen verse in expert voice so that he can found where he misread the word and then he given another option to try it again. User does this process again and again till he recites the whole verse correctly.

### 3. Results

For experimentation, three groups of reciters men, women and children are chosen. Each reciter was asked to read some specific verse of the Holy Quran and his recitations were tested against the expert's recitation. All these experiments are performed in the presence of an expert Hafiz. It is because an expert hafiz knows better either a user recite correct or wrong. So, when any candidate recites any verse, the expert listen his utterance along with E-Hafiz. After evaluating the utterance of candidate, expert tells his decision which is actually the true result of user utterance. The result generated by E-Hafiz is compared with the expert's result. This comparison tells us the accuracy rates of E-Hafiz i.e. how much results generated by E-Hafiz are correct.

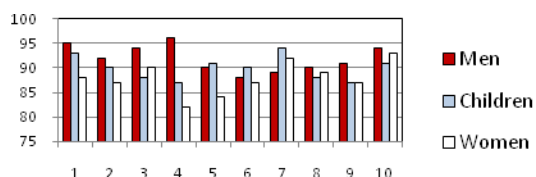


Figure 6. The accuracy rate of E-Hafiz against each candidate

The accuracy rate of E-Hafiz against each candidate is calculated by use of the following formula:

$$\text{Accuracy Rate} = \frac{\text{No. of correct identification by E-Hafiz}}{\text{Total No. of verses}} \times 100 \quad (9)$$

The accuracy rate of each candidate is shown in the figure 6 in which x-axis contains the candidate IDs and y-axis contains results. Also the global mean of all type of candidate is calculated by dividing the sum of all results of one type of candidate to the total number of candidate in that group as shown in table 1.

These results make us inspire and encourage us to make more improvements in this system and

enhance its performance much more than the current ones.

Table 1. Accuracy evaluation of E-Hafiz

Type of Reciters	Number of Reciters	Accuracy Rate
Men	10	92%
Children	10	90%
Women	10	86%

### 4. Discussions

The system solves a huge problem of arranging Hafiz for learning Quran or in cases where Hafiz could not be arranged learning in fear of mistakes. Users of the system have the possibility to finding mistakes and enhance recitation skills. After meticulous testing of system on many verses and test subjects the results obtained inspire us to further develop the system. With addition of word extraction feature E-Hafiz not only tackles the whole verse but also can identify the mistakes on word level. Our next endeavor is going to make this system capable of identifying the notes which are uttered, taking the recognition ability to letter level, where even a miss pronounced letter can be identified, further more a model assistance can be given on how to pronounce the word as per guided by rules in phonetics is underway.

#### Corresponding Author:

Waqar Mirza M., Aslam Muhammad  
Dept. of Computer Science & Engineering,  
University of Engineering & Technology,  
Lahore, Pakistan

E-mail: {[mirzamwaqar\\_mam1713@hotmail.com](mailto:mirzamwaqar_mam1713@hotmail.com)}

### References

1. Abdul Rashid Sheikh, Teachings and reflections of Qur'an and Sunnah about Science, Engineering, Technology and Management. Quaid-e-awan University research journal of engineering, science and technology 2000; 1(1).
2. H. Tabbal, W. El-Falou and B. Monla, Analysis and Implementation of a "Quranic" verses delimitation system in audio files using speech recognition techniques. Proc. of the IEEE Conf. of 2nd Information and Communication Technologies 2006; 2(1): 2979 – 2984.
3. Institute for Research in Islamic education, The New Strait Times online – 26 Sep. 2007.
4. M.S. Bashir, S.F. Rasheed, M.M.Awais, S. Masud and S. Shamail, Simulation of Arabic Phoneme Identification through Spectrographic Analysis. CS, LUMS, Lahore, Pakistan.

5. Mark Greenwood and Andrew Kinghorn: SUVing, Automatic Silence/Unvoiced/Voiced Classification of Speech. Undergraduate Coursework, Department of Computer Science, The University of Sheffield, UK. (1999).
6. Noor Jamaliah Ibrahim, Zaidi Razak, Mohd Yakub, Zulkifli Mohd Yusoff, Mohd Yamani Idna Idris and Emran Mohd Tamil, Quranic verse Recitation feature extraction using Mel-Frequency Cepstral Coefficients (MFCC). Proc. the 4th IEEE Int. Colloquium on Signal Processing and its Application (CSPA) 2008, Kuala Lumpur, MALAYSIA. (2008).
7. Program j-QAF sentiasa dipantau, Berita Harian Online (2005).
8. R. M. Gray, Vector Quantization. IEEE ASSP Magazine 1984:4-29.
9. Satori, H. Harti, M. and Chenfour N. Introduction to Arabic Speech Recognition Using CMUSphinx System. ISCIII '07. International Symposium 2007:31-35.
10. The Holy Quran available in almost all languages.
11. Zaidi Razak, Noor Jamaliah Ibrahim, Mohd Yamani Idna Idris, Emran Mohd Tamil, Mohd Yakub, Zulkifli Mohd Yusoff and Noor Naemah Abdul Rahman, Quranic Verse Recitation Recognition Module for Support in j-QAF Learning: A Review. IJCSNS International Journal of Computer Science and Network Security 2008;8(8).

2/12/12