ACIT'2017
The International Arab Conference on Information Technology
Yassmine Hammamet, Tunisia
December 22-24, 2017

Automatically Determining Correct Application of Basic Quranic Recitation Rules

Nour Alhuda Damer,* Mahmoud Al-Ayyoub,* and Ismail Hmeidi*

Jordan University of Science and Technology, Irbid, Jordan *

***Abstract*—Quranic Recitation Rules (Ahkam Al-Tajweed) are the articulation rules that should be applied properly when reciting the Holy Quran. Most of the current automatic Quran recitation systems focus on the basic aspects of recitation, which are concerned with the correct pronunciation of words and neglect the advanced Ahkam Al-Tajweed that are related to the rhythmic and melodious way of recitation such as where to stop and how to "stretch" or "merge" certain letters. The only existing works on the latter parts are limited in terms of the rules they consider or the parts of Quran they cover. This paper comes to fill these gaps. It addresses the problem of identifying the correct usage of Ahkam Al-Tajweed in the entire Quran. Specifically, we focus on eight Ahkam Al-Tajweed faced by early learners of recitation. Popular audio processing techniques for feature extraction (such as LPC, MFCC and WPD) and classification (KNN, SVM, RF, etc.) are tested on an in-house dataset. Moreover, we study the significance of the features by performing several t-tests. Our results show the highest accuracy achieved is 94.4%, which is obtained when bagging is applied to SVM with all features except for the LPC features.**

I. Introduction

The Holy Quran is the main sacred book of Islam. It is composed of 30 chapters. The verses of the Quran are 6236 verse grouped into 114 groups called "Surahs." Correct pronunciation of Quran verses during recitation is called in Arabic "Tajweed." Rules for recitation (Ahkam Al-Tajweed) must be considered in order to ensure the delivery of the correct meaning of these verses. There are many things that should be considered for perfect recitation including knowing the principles of Arabic pronunciation in a melodious voice.

There are many issues in teaching Ahkam Al-Tajweed. One of the major problems is that, in order to perfectly learn Ahkam Al-Tajweed, an expert is required and a special kind of private tutoring needs to be practiced, which is called "Talqeen." Such requirements are not always available. In order to tackle this issue, some researchers, turned to Machine Learning (ML) techniques to develop computerized systems to check the proper application of Ahkam Al-Tajweed based on the audio recordings of the user. However, the existing systems are limited in the rules they consider or the Quran verses they cover. In this paper, we address this problem and build a highly accurate and efficient system that consider all Ahkam Al-Tajweed faced by early learners in the entire Holy Quran.

This work consists of many steps. First, we consult with experts on teaching Quran recitation about the rules to take into consideration and the common mistakes made by the students in their early stages. Accordingly, we collect audio recordings of all verses in the Holy Quran that contain these rules. The second step is to apply the feature extraction techniques that are often used with human voice recordings in ML. After extracting the features of each recording, we experiment with different classification techniques commonly used for audio classification. We also perform several tests to determine the significance of the extracted features. The goal is to extract as few features as possible without affecting the accuracy of the system. Our final model has an accuracy exceeding 94%.

This paper is structured as follows. Related work is briefly discussed in Section II. Then, the proposed method and the experiments conducted to evaluate it are explained in Sections III and IV, respectively. Finally, a conclusion of the paper along with future directions is given in Section V.

II. Related Works

For several years, great efforts have been devoted to the study of Arabic speech by many researchers. There already exist detailed reviews of the evolution of Arabic using automatic speech recognition (ASR). In this section, only works related to Holy Quran are highlighted.

The authors of [1] created a system based on ASR techniques to help non-native Arabic speakers learn the correct recitation of some Quran verses. This offline system works in several steps. At each step, the system processes each word separately and saves it into a codebook. The codebook data is then compared with the database of correct recitations. If there is a wrong recitation, the system indicates this to the user and displays the correct recitation. This system gives an excellent accuracy of about 92%. However, it only consider a small number of Quran verses.

A Computer Aided Pronunciation Learning (CAPL) system called HAFSS was developed by [2] based on ASR techniques. The HAFSS system was built to help individuals learn the correct pronunciation of Quran verses. To achieve this, Maximum Likelihood Linear Regression (MLLR) was used in speaker adaptation block diagram, which used acoustic models and compared them to acoustic properties in order to boost system performance. To account for human errors and variance, HAFSS was represented in the form of a linear lattice that is flexible enough to support error hypothesis addition, deletion and overlapping of probable mispronunciation. HAFSS is based on the tri-phone state model HMM,

where each state is modeled after a mix of Gaussians. The database used by this system consists of 507 utterances tested and evaluated by language experts and labeled with actual pronounced phonemes that are used to compare and evaluate the recognized speech. The system achieved 97.87% accuracy surpassing a message based system performance which only achieved 80.13% accuracy. An enhancement on this system based on Speaker Adaptive Training (SAT) was discussed in [3] to address the problem of user enrollment time associated with ASR systems. The authors measured the correlation between the judgments of human experts and HAFSS system. They also measured the proficiencies of beginner users before and after using HAFSS system.

A Quran recognizer that is independent of the speaker is described in [4]. The system considers the 30th chapter of the Quran which includes over 2000 distinct words. Based on MLLR, a speaker independent speech recognition system was then developed. This process allows for the adaptation of recognizing speech without cues of a specific speaker. This is accomplished through global regression class tree (RCT) and MLLR adaptation. The system displayed an obstacle in terms of recognition time due to the large number of states of its HMM. The level of accuracy of the system was in the range 68-85%.

Perhaps, the closest to our work is the work conducted at the J-QAF laboratories, which sought to improve the teaching of the Holy Quran. These laboratories use the traditional methods for teaching the Quran, where the students learn directly from an expert in Quran recitation. A study [5] has been conducted for how this method can become electronic without the need for the presence of expert teacher. Literature related to this work state that MFCC features give the best accuracy in analyzing the verses, making it a possible alternative. This study led to the development of a system that corrects the recitation for the user in two small Surahs: "AlFatiha" and "AlEkhlas", by extracting the features from different recitations and different expert readers for previous small chapters of Quran, then using HMM to map each feature to related text [6]. Multiple recitations from multiple readers are used for reference, raising the accuracy for correct recitation to 98%. Here, the authors only consider four recitation rules in a very small part of the Quran. In our work, we consider the correct as well as incorrect application of eight recitation rules in the entire Holy Quran.

We notice that existing systems on the classification and correction for Ahkam Al-Tajweed are limited in their coverage of the Holy Quran [4], [7], [8] and the recitation rules (the number of rules do not exceed four rules for each system [2], [3], [6], [8]). Also, they assume that the verse being recited is known [6]. Our work comes to fill these gaps.

III. Proposed Approach

Our goal is to build a system capable of determining which one of the Ahkam Al-Tajweed is used in a specific audio recording of a Quranic recitation. The Ahkam Al-Tajweed we consider are eight: "Edgam Meem" (one rule), "Ekhfaa Meem" (one rule), "Ahkam Lam in 'Allah' Term" (two rules) and "Edgam Noon" (four rules). Moreover, we consider both correct and incorrect usage of each rule. Hence, our classification problem involves 16 classes. Finally, unlike previous works, ours covers the entire Holy Quran.

TABLE I
The number of audio recordings for each rule (showing both the correct and incorrect usage of this rule).

Rule	Correct?	Recordings by males	Recordings by females
R1: Edgam Meem	Correct	60	60
	Incorrect	30	-
R2: Ekhfaa Meem	Correct	60	60
	Incorrect	30	-
R3: Tafkheem Lam	Correct	88	512
	Incorrect	55	407
R4: Tarqeeq Lam	Correct	60	195
	Incorrect	30	134
R5: Edgam Noon (Noon)	Correct	138	138
	Incorrect	-	68
R6: Edgam Noon (Meem)	Correct	107	106
	Incorrect	-	53
R7: Edgam Noon (Waw)	Correct	52	52
	Incorrect	-	26
R8: Edgam Noon (Ya')	Correct	221	219
	Incorrect	-	110

We start our discussion of our approach with the dataset description (Section III-A). Then, we discuss the feature extraction (Section III-B) and classification steps (Section III-C) using popular techniques from the speech processing literature.

A. Dataset

Our dataset consists of 3,071 audio files, each containing a recording of one of the eight rules under consideration (in either the correct or the incorrect usage of the rule). We collect these recordings from ten different expert reciters (five males and five females). Table I shows the number of audio recordings for each rule and how they are distributed across the two genders. After their collection, the recordings are preprocessed and the part that contains the rule is extracted.

B. Feature Extraction Techniques

Feature extraction is the process in which sequences of feature vectors are computed to represent an audio signal. In this work, we employ three feature extraction techniques, namely, Linear Predictive Code (LPC), Mel-frequency Cepstral Coefficients (MFCC), and Multi-Signal Wavelet Packet Decomposition (WPD). We explain them in the following subsections.

1) Linear Predictive Code (LPC): LPC is a technique used to extract the audio information by using powerful and simple methods. It creates a vector of coefficients by using previous features to determine future features that have a smooth spectral envelope of a temporal input signal. LPC takes the signal as a vector, and then finds the coefficients of a pth-order linear predictor (FIR filter) that predicts the current value of the real-valued time series based on past samples.[1]

[1]https://www.mathworks.com/help/signal/ref/lpc.html

Removing the redundancy from the signal is the strong point in LPC [9]. In our work, we use MATLAB's (lpc) function for extracting the features from the signal.

2) Mel-Frequency Cepstral Coefficients (MFCC): The use of the MFCC has proven to yield amazing results in the field of speech recognition and it works to copy the behavior of auditory system by transforming the frequency from a linear scale to a nonlinear scale [8]. The first step to compute MFCC is breaking the signal down to windows using a hamming window. Then, Fast Fourier Transform (FFT) is used for each window in order to map it to Mel using a filter bank. Finally, the discrete cosine transform is applied on Mels and the mean and standard deviation (SD) are computed for the coefficients. The mean of the coefficients provides frequency distribution information for the audio signal, while the SD provides information about how much the frequency distribution changed through the audio signal [10].

3) Wavelet Packet Decomposition (WPD): The wavelet packet method is a generalization of wavelet decomposition that offers a vast range of possibilities to analyze signals. In wavelet analysis, a signal is split into an approximation and a detail. The approximation is then split into a second-level approximation and detail, and the process is repeated, further decomposing the signal [11]. We use WPT in a way similar to [12]'s, where a decomposition of complete tree up to six levels is generated for each Stereo record in the dataset. At each level, there are subspaces, where each subspace is a feature space. Then, the features are combined by normalized filter bank energy. The results from this technique are two dimensional matrix of features. In our work, we compute the mean and the SD for this matrix for each audio file.

C. Classification

Several classifiers are considered in our work. We give attention to the classifiers that have done well in other works related to audio classification in general and, specifically, for speech processing. Two types of classifiers are considered as described in the following subsections.

1) Nonlinear ML Algorithms: This type of ML algorithms use one layer processing. They also do not have ability to find the relationship between input attribute and the output attribute being predicted. In the following, we list the most popular nonlinear classification algorithms we consider.

- k-Nearest Neighbors (KNN) [13], [14]: This algorithm works by saving the entire training data, and when a prediction is requested on a certain instance, it tries to locate the k most similar instances in the training data.
- Support Vector Machines (SVM) [15]: SVM works by trying to find the best line that separates the data into several groups. An optimization process is used for this purpose.
- Artificial Neural Network (ANN): The ANN classifier we employ is called Multilayer Perceptron (MLP), in which a set of appropriate outputs are mapped from a set of input data. A MLP is a directed graph with multiple layers of nodes. All nodes in each level are fully connected with the next level. Each node is a processing element with a function. In order to train the model, a supervised learning technique called backpropagation is used. MLP is a developed version of standard linear perceptron, in which the data that are not linearly separable can be distinguished.

2) Ensemble ML Algorithms: More accurate predictions might be obtained by carefully combining the predictions from multiple models. This is done using Ensemble ML algorithms. We list the popular Ensemble algorithms with a short description for each one.

- Random Forest (RF): RF is a strong classifier used in ML field [16]. This classifier has several strong points such as its high classification accuracy, its non-parametric nature and its ability to determine the importance of variables. RF classifier is considered to be a part of the black box type because the classification split rules are unknown.
- Multiclass Classifier: It is a meta classifier used to link multiclass datasets with second class classifier. It can also increase the accuracy by applying error correcting output codes. This classifier has a base classifier that can be selected manually [17]. In our work, the base classifier is set to the RF classifier. This choice is made after an extensive set of experiments.
- Bagging [18]: In order to improve classifier stability, accuracy and reduce variance, bootstrap aggregating approach is used. It works as follow. It uses sampling with replacement to create replicates of the dataset, and, then, for each replica, it builds a model. Then, all created models are used in a classifiers ensemble. A single outcome will be computed by combining individual outcomes.

IV. Experiment and Results

In this section, we describe our experiments, which are split into two parts: with all features and with some features excluded. However, before going into the experiments, we discuss the experimental setup we use.

We use the Weka tool [19] for testing and classification purposes. The testing technique we follow is the 10-fold cross-validation. As for accuracy measures, we report the most popular measures used in the literature. Particularly, for every single classifier, the weighted precision/recall values, F-score, accuracy and AUC are reported, where AUC represents the area under the Receiver Operating Characteristic (ROC) curve [20].

A. Using All Features

In these experiments, we combine all features including LPC, mean and SD of MFCC, and mean and SD of WPD. Many classifiers are used for classification and testing process in these experiments. In Table II, we list the classifiers with the best results on our dataset.

We use the bagging technique with many algorithms such as SMO, RF, etc. However, the best result was when we use it with SMO, which is close to the result when we use SMO classifier alone. Also, we have run multilayer classifier

TABLE II
RESULTS USING ALL FEATURES

Classifier	Prec	Rec	F1	ROC	Build	Acc
Bagging (SMO)	0.944	0.943	0.943	0.996	35.050	94.334
SMO	0.943	0.941	0.941	0.993	4.280	94.138
IBk	0.909	0.908	0.907	0.945	0.030	90.752
Multilayer	0.900	0.897	0.897	0.983	273.260	89.677
Multiclass (RF)	0.887	0.886	0.882	0.994	35.150	88.570
RF	0.837	0.87	0.865	0.993	5.260	87.007

TABLE III
RESULTS USING ALL FEATURES EXCEPT LPC FEATURES.

Classifier	Prec	Rec	F1	ROC	Build	Acc
Bagging (SMO)	0.945	0.944	0.944	0.995	32.560	94.431
SMO	0.943	0.942	0.942	0.993	4.290	94.171
IBk	0.919	0.918	0.918	0.953	0.000	91.794
Multilayer	0.897	0.894	0.895	0.977	245.870	89.449
Multiclass (RF)	0.887	0.886	0.883	0.994	23.590	88.603
RF	0.880	0.877	0.872	0.994	7.360	87.723

many times with different numbers of hidden layer units and the best result is when hidden layer units are 16 (i.e., the number of classes). Multiclass classifier is used with different classifiers and the best result is when it is used with RF and default parameters. Finally, RF classifier is used with different parameter values, but the best results are obtained with its default parameter values.

B. Features' Significance

In this part, we discuss the results of excluding some features. This is done in two phases: the first one by excluding one feature type at a time, and the second one by excluding two feature types at a time.

1) Excluding One Feature: In this part of experiments, we exclude one feature in each experiment and show the performance of the model in each experiment. The effect of each ablation step is verified using statistical hypothesis test.

2) LPC Features Ablation: LPC features are excluded in these experiments. Table III shows the results of this experiment. From these results, we notice that, for all classifiers, the results are better after the ablation step.

3) Mean of MFCC Features Ablation: The mean for MFCC features is excluded in this experiment. Table IV shows the results of this experiment. From the results, F-score are decreased for all classifiers and specifically for the best one bagging, and build time is increased while the accuracy fall down from 94.4% to 87%. The area under the curve also falls down to 0.986. This means that the ablation of these features will affect the system in negative way. Later, we prove this using t-test.

4) Standard Deviation for MFCC Features Ablation: Here, the SD for MFCC features are excluded in this experiment. The experiment results are shown in Table V. The results of this and the previous experiments show that the effect of ablating the mean of MFCC is more significant that ablating the SD. Here, the results being more accurate than the previous experiment but still bad compared with using all features or when ablating LPC.

TABLE IV
RESULTS USING ALL FEATURES EXCEPT MEAN OF MFCC FEATURES.

Classifier	Prec	Rec	F1	ROC	Build	Acc
Bagging (SMO)	0.870	0.871	0.869	0.989	41.590	87.072
SMO	0.866	0.866	0.865	0.983	9.430	86.584
IBk	0.817	0.814	0.813	0.895	0.000	81.439
Multilayer	0.816	0.816	0.814	0.968	276.980	81.569
Multiclass (RF)	0.820	0.820	0.811	0.986	25.080	81.960
RF	0.814	0.810	0.800	0.985	7.910	80.983

TABLE V
RESULTS USING ALL FEATURES EXCEPT STANDARD DEVIATION OF MFCC FEATURES.

Classifier	Prec	Rec	F1	ROC	Build	Acc
Bagging (SMO)	0.911	0.909	0.908	0.992	35.770	90.947
SMO	0.903	0.902	0.901	0.987	7.800	90.231
IBk	0.877	0.875	0.874	0.927	0.000	87.463
Multilayer	0.874	0.869	0.869	0.967	299.350	86.909
Multiclass (RF)	0.862	0.861	0.857	0.991	23.180	86.095
RF	0.854	0.853	0.848	0.991	7.680	85.314

5) Mean for WPD Features Ablation: Mean for WPD features are excluded in these experiments. Table VI shows the results of this experiment, from which, it is evident that ablating the mean of WPD features have no significant effect.

6) Standard Deviation for WPD Features Ablation: Standard deviation for WPD features are excluded in this experiment. Table VII shows the results of this experiment. We notice from this table that ablating SD of WPD features have more effect on the results compared with ablating mean features. The table shows that all of precision, recall, F-score, roc area, accuracy and build time are decreased.

7) Statistical Hypothesis Test for One Feature Ablation: Based on the previous results for classification and testing process, we can determine which features can be excluded by comparing the F-score before and after each ablation process in order to build confidence in our results about which features can be excluded in order to improve the performance and

TABLE VI
RESULTS USING ALL FEATURES EXCEPT MEAN OF WPD FEATURES.

Classifier	Prec	Rec	F1	ROC	Build	Acc
Bagging (SMO)	0.936	0.935	0.935	0.994	25.800	93.487
SMO	0.934	0.933	0.932	0.992	3.030	93.259
IBk	0.919	0.918	0.918	0.952	0.000	91.794
Multilayer	0.896	0.897	0.896	0.985	203.030	89.710
Multiclass (RF)	0.901	0.900	0.896	0.995	20.140	89.970
RF	0.896	0.892	0.888	0.995	6.500	89.221

TABLE VII
RESULTS USING ALL FEATURES EXCEPT STANDARD DEVIATION OF WPD FEATURES.

Classifier	Prec	Rec	F1	ROC	Build	Acc
Bagging (SMO)	0.929	0.929	0.929	0.994	25.890	92.901
SMO	0.930	0.929	0.929	0.991	3.130	92.933
IBk	0.930	0.929	0.929	0.959	0.000	92.868
Multilayer	0.899	0.897	0.897	0.981	205.420	89.677
Multiclass (RF)	0.893	0.893	0.890	0.994	19.720	89.286
RF	0.894	0.892	0.888	0.995	6.090	89.156

TABLE VIII
T-TEST RESULTS FOR ALL EXPERIMENTS.

Ablated Features	LPC	MFCC: Mean	MFCC: SD	WPD: Mean	WPD: SD
Mean	0.926	0.819	0.885	0.919	0.908
t Stat	0.546	6.219	3.964	1.772	2.264
t Critical	2.131	2.131	2.131	2.131	2.131
P-value	0.300	0.000	0.000	0.050	0.020

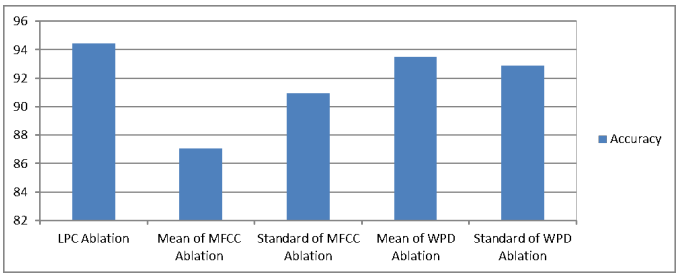

Fig. 1. Accuracy at each feature ablation process.

accuracy for our model. To do this, independent samples t-test is used to determine which feature can be excluded without affecting the results negatively. Such tests are used in many studies for this purpose [21].

In our work, we select the best classifier according to the mentioned measures, which was SMO with bagging applied it. We conduct five t-tests for first level of experiments on F-score. In each one of them, we have hypothesis about using this feature in our work or ablate it. Table VIII shows the results of t-tests for the five experiments.

From Table VIII, we notice that the null hypotheses are accepted regarding the exclusion of LPC features or mean of WPD features, since t Stat is less than t Critical and P-value is less than or equal 5%. When P-value is less than 5%, this means that there is only a 5% chance to make a difference on the results using these features. In contrast with these results, for excluding mean of MFCC features and SD for MFCC and WPD features, the alternative hypothesis is accepted (and the null hypothesis is rejected) since t Stat is always greater than t Critical and P-value is less than the significance level (5%). Figure 1 shows model accuracy at each feature ablation process.

From Figure 1, P-value for excluding LPC is more than 0.05 and the accuracy is 94.43; note that the accuracy was 94.3341%. This means that excluding LPC features upon P-value will increase the model accuracy. According to excluding mean of MFCC features, accuracy fall down to 87% when P-value is nearly equal to zero. And so on for all experiments.

From hypothesis results, we conclude that removing LPC features or mean for WPD features will increase F-score and decrease model build time, which means that the accuracy, which depend on F-score, and the performance, which depends on build time, will be improved. We use these results in the second level of experiments by excluding two features together in order to get more accurate results and determine which features can be selected to build out our model.

TABLE IX
RESULTS USING ALL FEATURES EXCEPT LPC AND MEAN OF WPD FEATURES.

Classifier	Prec	Rec	F1	ROC	Build	Acc
Bagging (SMO)	0.931	0.931	0.930	0.993	24.140	93.064
SMO	0.929	0.929	0.928	0.958	0.000	92.868
IBk	0.929	0.928	0.928	0.991	2.190	92.803
Multilayer	0.905	0.904	0.902	0.993	19.060	90.426
Multiclass (RF)	0.894	0.892	0.892	0.984	164.420	89.189
RF	0.895	0.893	0.889	0.994	6.210	89.254

TABLE X
T-TEST RESULTS FOR EXCLUDING TWO FEATURES TOGETHER.

Classifier	LPC with mean of WPD Ablation
Mean	0.928
t Stat	1.820
t Critical	1.753
P-value	0.040

8) Excluding Two Features: In the previous experiments, we have ablated one feature at a time. Now, we use the features that are rejected from the previous section and exclude two of them at a time. We run our experiments using many classifiers and report the best ones.

Results of ablating LPC and mean of WPD features are illustrated in Table IX. Upon the above results, we select the best F-score which was when we apply bagging to SMO classifier. We take bagging F-scores for all classes and perform a t-test on them. Table X shows the result of the t-tests.

Depending on the results in Table X, we notice that the null hypothesis is rejected according to the value of t Stat, which is greater than t Critical, and the value of P, which is in the significance level (less than 5%). This is an evidence to prove the results in Table VIII, in which the F-score falls down while excluding LPC features and mean for WPD features, and accurate result can be obtained by excluding the LPC features only. Figure 2 shows F-score for our model while excluding LPC features and excluding LPC with the mean of WPD features.

V. CONCLUSION

In this work, we present our efforts towards building computer systems capable of automatically determining the recitation rule used in an audio recording. Unlike existing

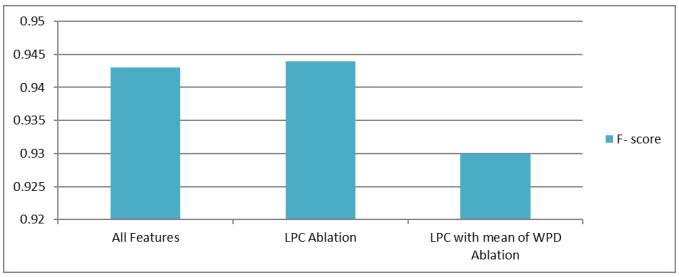

Fig. 2. F-score when we combining all features and ablation of one feature and two features together.

work, we consider eight rules and their (correct as well as incorrect) application throughout the Holy Quran. After experimenting with many feature extraction and classification techniques, we reached the conclusion that the best features can be extracted from the incoming voice are MFCC with WPD (mean and standard deviation for each one), which is in accordance with what is found in the literature of audio processing about MFCC being the best feature extraction technique. As for the classification part, SMO with bagging gives the best results, which is also in accordance with the current literature.

In the future, we plan on expanding our dataset to include more variation in terms of the reciters' background and proficiency levels. We also plan on developing a mobile application in order to make it possible for end users with limited knowledge of computers to benefit from our research.

REFERENCES

[1] W. M. Muhammad, R. Muhammad, A. Muhammad, and A. Martinez-Enriquez, "Voice content matching system for quran readers," in *Artificial Intelligence (MICAI), 2010 Ninth Mexican International Conference on*. IEEE, 2010, pp. 148–153.

[2] S. M. Abdou, S. E. Hamid, M. Rashwan, A. Samir, O. Abdel-Hamid, M. Shahin, and W. Nazih, "Computer aided pronunciation learning system using speech recognition techniques," in *Ninth International Conference on Spoken Language Processing*, 2006.

[3] A. Samir, S. M. Abdou, A. H. Khalil, and M. Rashwan, "Enhancing usability of capl system for qur'an recitation learning," in *Eighth Annual Conference of the International Speech Communication Association*, 2007.

[4] E. Mourtaga, A. Sharieh, and M. Abdallah, "Speaker independent quranic recognizer based on maximum likelihood linear regression," in *Proceedings of world academy of science, engineering and technology*, vol. 36, 2007, pp. 61–67.

[5] N. J. Ibrahim, Z. Razak, Z. Mohd Yusoff, M. Y. I. Idris, E. Mohd Tamil, N. Mohamed Noor, and N. Naemah, "Quranic verse recitation recognition module for support in j-qaf learning: A review," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 8, no. 8, pp. 207–216, 2008.

[6] N. J. Ibrahim, Z. M. Yusoff, and Z. Razak, "Quranic verse recitation recognition module for educational programme," University of Malaya, Tech. Rep., 2008.

[7] H. Tabbal, W. El Falou, and B. Monla, "Analysis and implementation of a" quranic" verses delimitation system in audio files using speech recognition techniques," in *Information and Communication Technologies, 2006. ICTTA'06. 2nd*, vol. 2. IEEE, 2006, pp. 2979–2984.

[8] N. J. Ibrahim, Z. Razak, M. Yakub, Z. M. Yusoff, M. Y. I. Idris, and E. M. Tamil, "Quranic verse recitation feature extraction using mel-frequency cepstral coefficients (mfcc)," in *Proc. of the 4th IEEE Int. Colloquium on Signal Processing and its Application (CSPA), Kuala Lumpur, Malaysia*, 2008.

[9] KohshelanEmail and N. Wahid, "Improvement of audio feature extraction techniques in traditional indian musical instrument," in *Recent Advances on Soft Computing and Data Mining*. Springer, 2014, pp. 507–516.

[10] G. Tzanetakis, G. Essl, and P. Cook, "Audio analysis using the discrete wavelet transform," in *Proc. Conf. in Acoustics and Music Theory Applications*, 2001.

[11] A. Haque, "Fft and wavelet-based feature extraction for acoustic audio classification," *International Journal of Advance Innovations, Thoughts & Ideas*, vol. 1, no. 1, 2012.

[12] R. N. Khushaba, S. Kodagoda, S. Lal, and G. Dissanayake, "Driver drowsiness classification using fuzzy wavelet-packet-based feature-extraction algorithm," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 121–131, 2011.

[13] G. Peeters, "Automatic classification of large musical instrument databases using hierarchical classifiers with inertia ratio maximization," in *Audio Engineering Society Convention 115*. Audio Engineering Society, 2003.

[14] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on speech and audio processing*, vol. 10, no. 5, pp. 293–302, 2002.

[15] J. T. Geiger, B. Schuller, and G. Rigoll, "Large-scale audio feature extraction and svm for acoustic scene classification," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013 IEEE Workshop on*. IEEE, 2013, pp. 1–4.

[16] V. F. Rodriguez-Galiano, B. Ghimire, J. Rogan, M. Chica-Olmo, and J. P. Rigol-Sanchez, "An assessment of the effectiveness of a random forest classifier for land-cover classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 67, pp. 93–104, 2012.

[17] I. H. Witten, E. Frank, L. E. Trigg, M. A. Hall, G. Holmes, and S. J. Cunningham, "Weka: Practical machine learning tools and techniques with java implementations," The University of Waikato, Tech. Rep., 1999.

[18] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[19] C.-Y. Lin and H.-C. Wang, "Language identification using pitch contour information," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, vol. 1. IEEE, 2005, pp. I–601.

[20] M. Al-Ayyoub, M. K. Rihani, N. I. Dalgamoni, and N. A. Abdulla, "Spoken arabic dialects identification: The case of egyptian and jordanian dialects," in *Information and Communication Systems (ICICS), 2014 5th International Conference on*. IEEE, 2014, pp. 1–6.

[21] M. Al-Ayyoub, Y. Jararweh, M. Daraghmeh, and Q. Althebyan, "Multi-agent based dynamic resource provisioning and monitoring for cloud computing systems infrastructure," *Cluster Computing*, vol. 18, no. 2, pp. 919–932, 2015.