MELODY TRAINING WITH SEGMENT-BASED TILT CONTOUR FOR QURANIC TARANNUM

Haslizatul Mohamed Hanum[1*], Luqmanul Hakim Md Abas[2], Aiman Syamil Aziz[3], Zainab Abu Bakar[4], Norizan Mat Diah[5], Wan Fatimah Wan Ahmad[6], Nazlena Mohamad Ali[7], Norshuhani Zamin[8]

[1, 2,3,5] Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia

[4] Faculty of Computer Science and Information Technology, Al-Madinah International University, Shah Alam, Selangor, Malaysia

[6] Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Perak, Malaysia

[7] Institute of IR4.0, Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia

[8] College of Computing and Informatics, Saudi Electronic University, Saudi Arabia

Email: haslizatul@uitm.edu.my[1*](corresponding author), namquluqman@gmail.com[2], aimansyamilaziz@gmail.com[3], zainab.abubakar@mediu.edu.my[4], norizan@fskm.uitm.edu.my[5], fatimhd@utp.edu.my[6], nazlena.ali@ukm.edu.my[7], n.zamin@seu.edu.sa[8]

DOI: https://doi.org/10.22452/mjcs.sp2021no2.1

ABSTRACT

Tarannum, or melodic recitation of Quranic verses, employs the softness of the voice in reading the holy verses of the Quran. Melody training technology allows users to practise repetitively while also providing feedback on their performance. This paper describes an application that captures the pattern of tarannum melodies (from Quranic recitations) and provides feedback to the user. Recordings of Quranic verses are collected from an expert reciting Bayati tarannum. The samples are pre-processed into segmented tarannum verse-contours using pitch sequences. Using the k-Nearest Neighbor (kNN) classifier, the melody patterns are trained on 20 samples. Input vectors are formed by computing the melody verse-contour representation using mean, standard deviation, and slope values and combining them with an identified Tilt-based contour label. A tarannum training prototype is built to test similarity between a user's recitation and the trained patterns. To identify similarity between a pair of verse-contours, the application employs a shape-based contour similarity algorithm. The proposed application also provides feedback in the form of a grade and a percentage of accuracy, as determined by a melody curve similarity algorithm. As results, the current samples have an overall shape-based weighted score of 66%. Some samples are successfully classified with a similarity score as high as 80% individually. The study provides an alternative interactive session for people who want to learn Tarannum, as well as a preliminary step toward understanding the melodic patterns for tarannum. The application provides a repetitive training experience and encourages users to improve their recitations in order to achieve the highest possible score.

Keywords: Audio Similarity, Genre Classification, Melody Training, Tarannum, Quranic Recitations

1.0 BACKGROUND

Melodic contour, or the pattern of rises and falls in pitch, is a critical component of melodic structure and impacts listeners' perceptions and memory for music. Several techniques for defining and computing melodic similarity have been proposed that cover distinct aspects or elements of melodies [1]. There are several ways to define and compute melodic similarity. Intervals, contour, rhythm, and tone are among the aspects that can be quantified [2]. Several applications have employed melody as pitch contour information, such as music query by humming [3], melody similarity matching [4,5], or melodic classification [6,7]. Melody is generally represented as continuous pitch sequences with durational values.

The intonation, modulation, recital, chanting, singing, and humming of Quranic verses are all examples of Quranic chanting (or humming). It creates a distinct melody (known as the tarannum melody) to capture the listener's attention and draw the listener deep into the appreciation of the verses while making the listening experience enjoyable. Adopting tarannum when reciting the Qur'an, requires one to do it according to the correct principles of *tajwid*, or Arabic elocution, with a melodious voice, while observing the meaning of the Qur'an effectively, as to embrace one's body and soul through the recitation [8]. Tarannum practise (also referred as Qur'anic chanting) stressed the importance of understanding the contents of the verses. Training of Quranic chanting is similar to singing and includes series of vocal and melody lessons practised on Quranic verses [9,10]. Focusing on components of *al-Ada'*, which are observation of the law of *tajwid* (elocution rules), *waqaf* (stop or pause) and *ibtida'* (beginning), the *tadabbur* (meditating) of the verse made best with engrossment in reading and correlating of verses with *maqam* tarannum [11].

Tarannum training for students is typically handled using the traditional method of face-to-face repetitive reading of the verses known as the *talaqqi* and *musyafahah* methods [12]. Tarannum learning method requires the process of repeating and correcting the reading, rhythm and tone by the teacher to the students [13]. Each training necessitates the participation of students and teachers in a series of repetitive sessions. Each session consists of repetitive tasks such as listening to pronunciations, correcting recitation rules (*tajwid*), and correcting taranum recitations (as commonly conducted in the j-Qaf classes) [12].

The study by Latif et. al., [14] found that teaching methods using computer technology were the most popular method teachers use in teaching tarannum; here are recommended that further studies should be made on the level of effective teaching of tarannum by teachers. As technology for identifying and analysing melodies evolves and improves, there is a need for an interactive web-based application that can implement the technology for Muslims to learn and practise the tarannum melodies. Furthermore, through multiple listening modules, a computer application can provide personalised materials and frequent practise. An application's immediate response can tell users how closely their humming matches the melody model [15]. Tarannum training can thus be improved by creating an interactive computer-assisted application that provides interactive feedback in order to attract (ease and promote) users (both old and young generations) to learn tarannum.

2.0 MELODIC RECITATION

Melodic recitation of the Quranic verses or tarannum uses the voice’s softness in reading the holy verses of the Quran. According to Dr. Ruuhi Al-Balbaki [16], tarannum can be translated as intonation, singing, and recital. This melodic recitation can make anyone who listens to the recitation of the Qur’an excited and calm. Tarannum is categorized into several types, such as Bayati, Hijaz, Soba, etc. Combinations of two or three types of tarannum are also practiced as different types of tarannum melodies can also be applied to Quranic verses to enhance the meaning. Table 1 lists the seven types of melodies for Quranic recitations, their properties, and the description of the verse types suited for the melody.

Table 1: Melodic recitation type, properties and verse type

Melodic Recitation Type	Property	Verse Type
Bayati	• Soft as well as firm • Harmony • Elements of serenity	• Simple verses
Soba	• Light movements permeate the soul. • A soft, sobbing sound	• Simple verses • Begging, moaning verses
Hijaz	• Slow but effective motion • Firmness of motivation • Agility enlivens presentation	• Any verses • Command, firm, or angry verses
Nahwand	• Effective light motion • Polite gentleness • Simple, captivating soul • Fine Art	• Simple verses
Rast	• Gentle movement along with firmness • Softness • Agile to brighten	• Any verses
Sikah	• Convincing slow motion • Softness pierces the heart. • Characteristic of sadness and grief	• Any verses • Begging, moaning verses
Jiharkah	• Effective light motion • Softness pierces the heart. • Harmony	• Simple verses • Begging, moaning verses

Currently, the way to read the Qur'an is to use manual methods such as *talaqqi* and *musyafahah* [12]. This method is a face-to-face meeting between students and teachers, such as learning in the classroom. This learning requires both students and teachers to be present at the same time and place. So far, it is common for a teacher to administer the tarannum practices in a classroom with a group of 5 to 15 students.

Evaluation of tone contours advances towards examining the similarities of melodic recitation based on the subjects' voices. Other strategies are applicable in processing music signals using artificial intelligence, such as humming music extraction, humming automated transcription, and automatic accompaniment. Most people would practice melodic recitation first by singing the melody without reciting the lines. Humming can intuitively give human beings a basic understanding of how to create the same intonation. This analysis must also provide the mode of recognition of the humming melody. Typically, people's humming performances also require personal expressive shifts, individualized emotional influences, and certain other unstable factors, which enhance the complexity of humming pitch recognition [17].

In this paper, the approach of finding identical melodious segments within the whole Quranic verse is applied. The proposed Tarannum-Tilt model for Quranic verses compares the humming pieces as curves in the pitch-time plane and compare the shape of contours from sub-segments of a verse. The similarity score assigned to each contour remarks the final score for the practice session. The following sections describe the related studies on melody including techniques for feature representation and curve similarity.

2.1 Previous Studies on Feature Representation and Melody Curve Similarity

Several techniques for defining and computing melodic similarity have been proposed that cover distinct aspects or elements of melodies. A study by [18] aims to assess the efficacy of transferring knowledge from related domains to the task of anomalous sound detection by employing various pre-trained neural networks (NNs). This study analysed feature representations for anomalous sound detection (ASD) from the music, image, and environmental sound domains to facilitate fast experimentation. The researchers concluded that the studied representations are suitable for ASD even when the feature extractor and ASD task are different domains.

Research by [19] discusses the different techniques used to classify and retrieve audio signals and proposes a new method for classifying and retrieving audio signals. They mentioned that determining acceptable content-based features for representing the audio signals in question is challenging when constructing an audio retrieval system. One of the techniques discussed is using the pitch extraction approach based on the frequency difference of the audio signal and using the probabilistic neural network (PNN) classifier to classify audio signals. The authors concluded the paper could be utilized as a reference for other researchers to identify the critical features of an input audio signal for their work.

Zhang et al. [20] proposed an unsupervised learning framework to learn a vector representation of an audio sequence for Acoustic Event Classification (AEC). The work applies a Recurrent Neural Network (RNN) encoder to convert a variable-length audio sequence into a fixed-length vector, and an RNN decoder reverses the process. The empirical results from the work concluded that compared to existing state-of-the-art hand-crafted sequence features for AEC, the learned audio sequence representation provides a considerable performance gain by a significant margin.

Nowadays, it is important to have systems that can classify, retrieve the various melodies similar to work done for folk music [21, 22, 23] and humming recitations [10]. The melody variations are produced and influenced by various environments and cultures. Recent research tested several melody similarity techniques used to find identical melodic segments within a whole melody contour [22]. Curve similarity technique is used in the system's similarity measurement by comparing the local features (including distance between tones, intervals, etc.) [24, 25], or comparing shapes of sequences [22, 26] after manipulating them, using existing methods to measure similarities between two vector sequences [24]. In the following section, the k-Nearest Neighbor (kNN) vector similarity technique used to obtain the similarity of melodies will be discussed. The melody similarity technique is needed to understand the similarity of melody curves (shape).

2.2 K-Nearest Neighbors Algorithm

The k-Nearest Neighbor (kNN) algorithm is a non-parametric approach to classification and regression for pattern recognition. kNN is also the most accessible algorithm for machine learning and can conclude that new knowledge and available data are identical. Its advantages are mainly manifested in its simple principle, convenient implementation, support of incremental learning, ability to build a model for ultra-polygon complex decision space, and better classification performance in the situation of the cross-class field [27]. The kNN algorithm has been used to classify the music genre using musical tags to effectively characterize similarities between artists, and the proposed approach outperforms the previous web-based methods for artist genre classification with the highest average accuracy [28].

Similarity algorithms such as k-Nearest Neighbor (kNN) were used to recognise genre from musical signals [29]. The approaches to computing the similarity of two melody sequences are by comparing their local features [24, 30] (such as distance, pitch, intensity, etc.) and by comparing the contour shape [26,31]. Since 2015, Janssen, Kranenburg, and Volk [22] have discussed melody comparison methods used in computational ethnomusicology, including genre classification [29, 32]. The research authors evaluated the algorithms on selected melody segments rather than the entire melody of the datasets. kNN's classification method is based on supervised learning, which is used to determine the similarity of datasets. First, the algorithm calculates the distance between evaluation and training data to select the data mark with the shortest distance. The data is then categorised according to the minimum distances. Thus, in this research, kNN compares the tarannum signal from users' voices and calculates the percentage of their melodic recitations that are correct.

2.3 Melody Similarity using Similarity Matrix Profile

Melodic similarity is useful for evaluating artist's performances. Song cover is the common term used to describe an artist who sing a song track that has already been released. A cover song can refer to a live performance, a remix, or an interpretation of a particular form of music. Besides evaluation of song tracks from the recorded signal, there are many uses of melodic similarity knowledge such as for copyright protection, catalogue organization, and content search [33]. In the research, cover songs were compared to their original version through the signal spectrograms as shown in Fig. 1. From the two spectogram low values of Similarity Matrix Profile (SiMPle) are expected when melodies from different songs are compared, while high values of SiMPle indicate the two profiles belong to the same song.

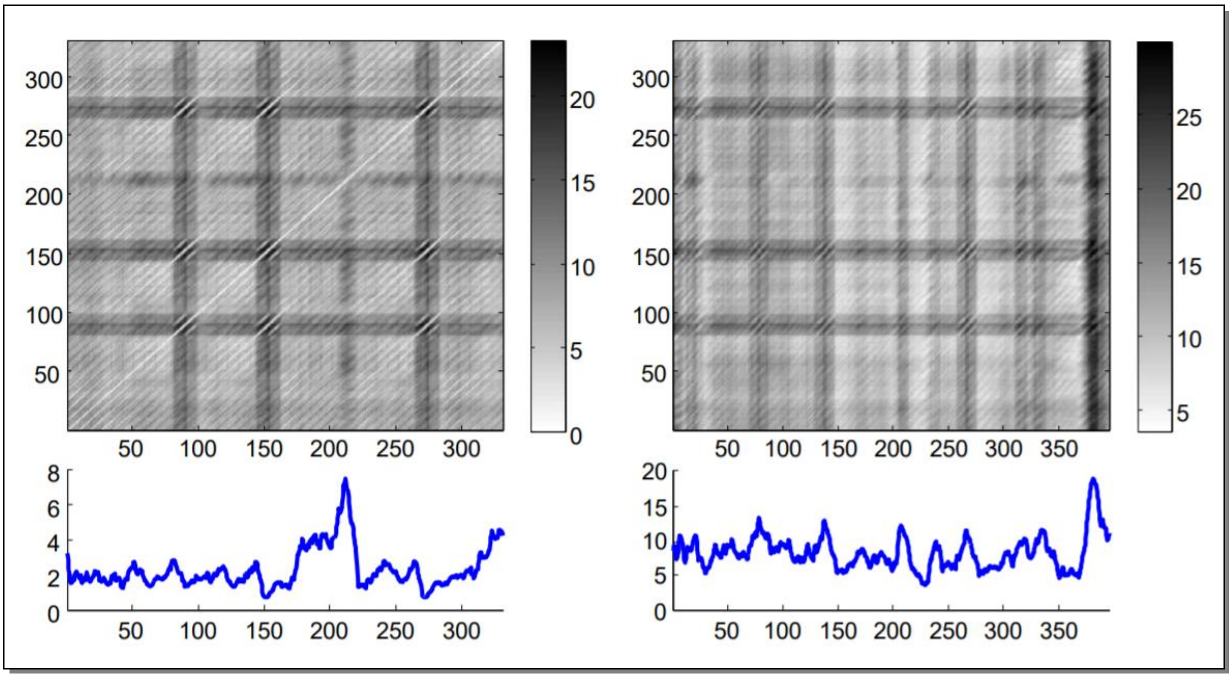


Fig 1: Similarity matrix within (left) and between different songs (right) and their respective SiMPle [33]

3.0` METHODOLOGY

The project's tasks are divided into three phases, as shown in Fig. 2. The first stage involves recording expert recitations and pre-processing the signals through noise reduction and segmentation. In the second phase, pitch features are used to generate the Tarannum melody vector from segmented contours. The Tarannum shape patterns are modelled after the vectors have been trained with a supervised classifier.

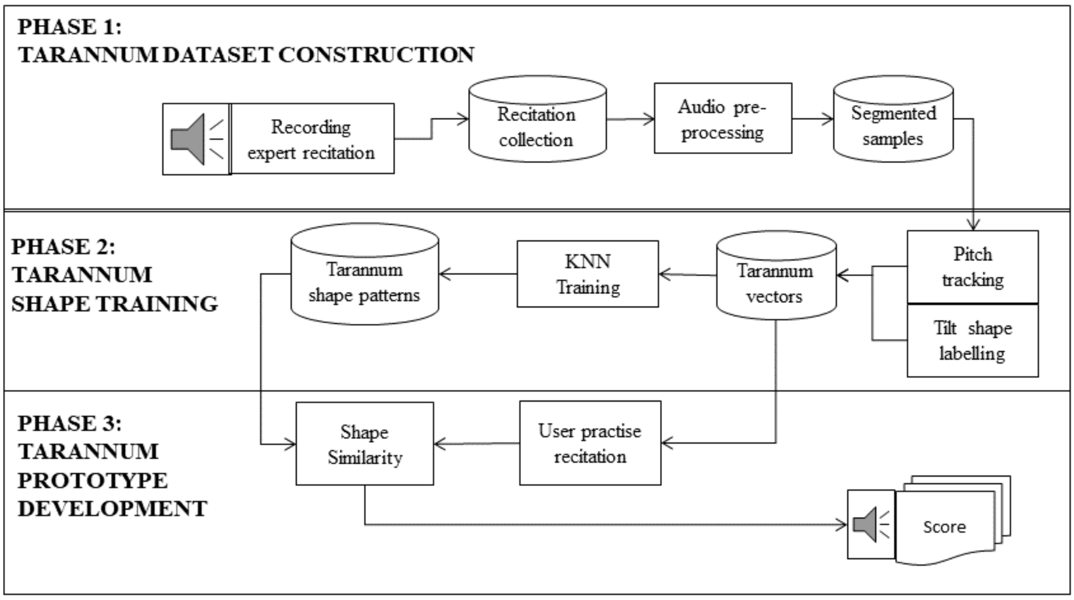


Fig 2: The methodology for Tarannum Melody Training

In the final phase, a prototype is developed to evaluate if the pattern of recitation practiced by user is similar to the model. User is presented with correctness score as interactive response to promote interest to continue the practice. Discussion of results obtained from the prototype is presented in section 4.0.

3.1 Tarannum Dataset

A total of four chapters from the Quran are identified. The chapters are commonly recited Quranic chapters (known as the *Surah Al-Ikhlas, An-Nas, Al-Kafiruun*). The recording files are coded as surah 02, 03, and 04, respectively, following the similar naming standard set for the pilot project using Surah *Al-Fatihah* [34].

A total of 22 samples of Tarannum recitations are recorded for each identified Quranic verse. The *Bayati* melody was chosen because it is the most common and straightforward genre. The recitations were recorded in a secluded room to ensure minimal background noise. A single male reciter who is an expert in Taranum recitation performed the recitation. Using the segment-based curve modelling approach, the first twenty samples are used to construct Taranum's melody pattern. The other two samples are used to put the classified pattern to the test.

3.2 Noise Reduction and Segmentation

Each sample's signals are pre-processed with a noise removal function and manually segmented into sub-verses. The audio tracks are exported to a separate folder after noise reduction. Using Praat's speech analysis tool, each signal is trimmed into smaller segments ranging from two to six. The pitch and intensity sequences are then used to divide each verse into sub-verses with similar melody patterns [22]. For example, the *Surah Al-Ikhlas* (coded as surah-2) contains four verses, the first of which is divided into two sub-segments (named as 20201 and 20102). The boundaries of each sub-verse are then marked using the interaction of pitch with intensity valleys, this step aids in reducing the size and space required to analyse the audio signal. The melody pattern is then constructed by evaluating pitch tracking on each part (sub-verse).

3.3 Pitch Tracking

Pitch tracking is a technique that enables a computer to recognize pitch sequences from an audio signal automatically. Melody recognition, tone recognition, prosody analysis for text-to-speech, intonation assessment, and voice recognition are several audio signal processing applications. Pitch estimation is critical in music signal processing, where monophonic pitch tracking is used to generate pitch annotations for extensive collections of audio tracks [35]. A pitch tracking tool is used for determining the pitch of an audio file which operates on the input signal recorded with the signal's timeline.

In general, the steps for the pitch tracking using CREPE Pitch Tracker [24, 30] are:

i) Segment the audio signal into sequence of 10 mili-seconds (ms) frames.
ii) Compute the pitch of each frame.
iii) Eliminate pitch from silence or unvoiced sounds by using pitch range thresholding.
iv) Extract the pitch sequences and smooth the pitch curve using median filters.

Each resulted curve represents the melody contour. Contour models like Tilt representation can be used to represent the melody contour.

3.4 Tilt-based Tarannum Shape Labeling

Tilt is a dimensionless parameter that describes the shape of an event by comparing the relative sizes of the event's rise and fall components [36, 37]. Before calculating the tilt parameter, F0 sequences are interpolated using the polynomial regression of order two to bridge the time's values. Then, interpolated values are estimated between two known F0 values in a sequence to produce a smooth contour versus the time series data.

From the estimated contour, Tilt parameters are computed using Tilt function as depicted in Fig. 3 and was written in Python using the panda library. First, the tilt vector values; the maximum and minimum pitch values are computed. Next, the rise and fall amplitudes are computed for each segmented verse. These parameters represent the strength of voice conveyed through a verse [38, 39].

```
# Compute RFC-Tilt vectors
startIdx = RFC.F0_.first_valid_index()
endIdx = RFC.F0_.last_valid_index()
startPitch = RFC.F0_.iloc[startIdx]     # start Index of F0_
endPitch = RFC.F0_.iloc[endIdx]
maxPitch = RFC.F0_.max()
maxIdx = RFC.F0_.idxmax()   # frame no.
minPitch = RFC.F0_.min()
minIdx = RFC.F0_.idxmin()   # frame no.

riseAmplitude = maxPitch - startPitch
riseDur = abs(maxIdx - startIdx)
fallAmplitude = endPitch - maxPitch
fallDur = abs(maxIdx - endIdx)
```

Fig. 3: Tilt function for computing contour parameters

Finally, the rise and fall durations are measured and used to obtain the duration for each verse contour. Table 2 shows a sample of shape vector constructed for the first verse of surah Al-Ikhlas. Each vector describes contain parameters that represent the shape of the verse's contour.

Table 2: Sample of computed parameters and label for first verse of Al-Ikhlas

	min	minFrameNo	max	maxFrameNo	rise	rDur	fall	fDur	label
0	117.87	34	190.05	63	55.844	63	-32.21	101	3

As shown in Table 2, the minimum pitch is listed first, followed by the frames number containing the minimum value. The vector also contains the maximum pitch value followed by its frame number, the rise amplitude, the rise duration, the fall amplitude, the fall duration, and finally, the label assigned to each verse contour. The label represented the shape of contour; 1 is a rising shape, 2 is a falling shape, 3 is a rise-fall with concave-down contour, 4 is a fall-rise contour, as shown in Table 3.

A Tilt-based labeling algorithm is adapted so that all samples are identified with a contour label. Four types of contours are identified for the Tarannum melody characterized using a fuzzy algorithm on the tilt parameters: rise amplitude, rise duration, fall amplitude, and fall duration. The tilt parameters are used to label each Tarannum sample. After labeling the contour, we construct the Tarannum vector for each sample.

Table 3: Types of contour pattern for Tarannum melody

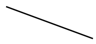
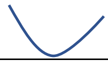
Contour Labels	**Shape Contours**
1	Rise 
2	Fall 
3	Rise – Fall 
4	Fall – Rise 

Table 4 shows the values for each Tilt parameter (the rise amplitude, rise duration, fall amplitude, and fall duration) extracted from each sample. Also shown is the corresponding label assigned to each sample for all the 20 contour samples of surah *Al-Ikhlas*.

Table 4: Tilt parameters and label of the first segment for all 20 samples

Sample No	**Rise Amplitude**	**Rise Duration**	**Fall Amplitude**	**Fall Duration**	**Contour Label**
1	55.844	63	-32.21	101	3
2	45.654	63	-46.175	106	3
3	69.882	64	-35.762	104	3
4	58.171	60	-29.46	103	3
5	66.702	60	-35.031	100	3
6	67.565	64	-37.353	96	3
7	63.782	59	-45.363	103	3
8	104.054	76	-27.111	85	3
9	71.798	61	-42.175	99	3
10	67.28	61	-40.412	100	3
11	67.074	59	-41.198	98	3
12	63.979	64	-38.604	98	3
13	78.089	59	-52.3	97	3
14	66.683	60	-40.309	99	3
15	62.26	63	-40.317	104	3
16	62.134	62	-37.683	102	3
17	124.147	1	-107.409	155	2
18	117.52	63	-36.3	91	3
19	59.178	58	-41.111	93	3
20	68.201	60	-62.457	98	3

As shown in Table 4, for the first segment of all 20 samples, the rising amplitude varies between 45.654 and 124.147. When a new sample is tested, the range can be used to classify the rise contour value. If the rise amplitude is within the model's average value for segment one, then the test sample can be accepted as having a correct (or similar) recitation. Meanwhile, the range of amplitude of the fall contour is between -107.4 and -27.1. The average amplitude of the fall and the duration of the fall are -43.437 and 102.6 seconds, respectively.

Therefore, it is critical to understand that these average values will serve as the label for the Tarannum-Tilt model. As we can see, the average score for the label for segment one in verse one is 2.95, indicating that the estimated contour label is more aligned with label three (label 3 for the rise-fall contour), as opposed to label two (label 2 for the fall contour).

3.5 Mean, Standard Deviation, and Slope Analysis

The mean is calculated as the product of two or more values. The standard deviation is used to describe the dispersion of a dataset in relation to its mean. The standard deviation is a measure of how to spread the data are. In a graph, the slope of a line denotes its steepness. The mean, standard deviation, and slope all contribute to the value of the similarity score when examining a graph containing statistical data. As a result, the Tarannum *Bayati* model must integrate these three qualities.

Three aggregated features and a label score are combined to form the feature vector for each sample. Table 5 shows the mean, standard deviation, and slope values computed for the first sub-segment of verse one (sample 20101) of surah *Al-Ikhlas*.

Table 5: The aggregated features for sub-segment 20101 of chapter *Al-Ikhlas*

Sample No	Mean	Standard Deviation	Slope	Contour Label
1	158.655357575757	23.8700166577929	0.0128889205606468	3
2	159.524470588235	22.6724600589867	0.013040701354436	3
3	160.290420118343	23.5777418303046	0.0134919179536355	3
4	157.410847560975	23.1886068512221	0.0138044092175164	3
5	158.498739130434	24.4868566601354	0.0129097916177077	3
6	157.555838509316	24.5698524695155	0.012416976999826	3
7	157.359552147239	22.6590027267264	0.0134028899989549	3
8	152.881481481481	22.9899854074293	0.0147349660804736	3
9	164.25001863354	23.8606621944132	0.0129016610769061	3
10	162.279895061728	22.8409862770181	0.0123501311489018	3
11	168.135993670886	23.8533371582667	0.0120944888882635	3
12	164.745184049079	23.1898525991618	0.0122904242676869	3
13	165.603866242038	25.0169883417955	0.010576889743837	3
14	160.2226125	22.666766129449	0.0117900660706578	3
15	165.008982142857	24.5741802963842	0.0130524977622335	3
16	161.344442424242	24.2144356436583	0.0135357716802913	3
17	161.460611464968	23.3992344126484	0.00950868745667883	2
18	157.609322580645	22.1776469255646	0.0123274733071612	3
19	163.561467105263	24.1309036851329	0.0120789858205988	3
20	160.47086163522	24.4676678924383	0.00972333925635324	3
Average	160.843498	23.62035921	0.01244605	2.95

The average for each mean, standard deviation, and slope values for sample 20101 is recorded as 161, 24, and 0.0124, respectively. The individual feature values extracted from all samples are tested to compute the similarity of the vector to the average value. Thus, similar vectors will output higher scores as a response to the training application.

4.0 RESULTS AND DISCUSSION

The initial project is to capture and analyse the melodic patterns of Quranic recitations. Twenty-two samples from each of the Quran's four chapters (*Surah Al-Fatihah, Al-Ikhlas, An-Nas, and Al-Kafiruun*) are used to analyse the melody patterns, emphasising the emphasis Taranum *Bayati*. The subsequent subsections examine the performance of the melodic model and the prototype designed to recognise Tarannum patterns.

4.1 Melody Model’s Performance

After running the vector of all the samples on KNN classifier, an approximate score is generated for each sample. The frequency of each type of contour label (as in Table 3) estimated by the classifier is computed. The results are presented in Table 6, which grouped estimated contour labels into two; Label-1 and label-2 based on the frequency score and are validated by a Tarannum teacher expert in the Bayati melody. The grouping help to set a weightage value suitable for modelling other tested contours.

Table 6: Accuracy of kNN classifier on samples from *al-Ikhlas*

Segments	**Label-1**	**Accuracy1**	**Label-2**	**Accuracy2**	**Frequency**
20101	3	100%	1,2,4,5	0%	[3: 19, 2: 1]
20102	2,3	66%	1,4,5	33%	[3: 9, 2: 11]
20201	3	83%	1,2,4,5	16%	[3: 13, 4: 2, 2: 5]
20202	3	50%	1,2,4,5	50%	[3: 11, 2: 5, 4: 4]
20301	3,4	66%	1,2,5	33%	[3: 14, 4: 6]
20302	2,3	66%	1,3,5	33%	[2: 15, 3: 5]
20303	3	83%	1,2,4,5	16%	[4: 3, 3: 14, 2: 3]
20401	3	66%	1,2,4,5	33%	[3: 12, 2: 5, 4: 3]
20402	2,3	100%	1,4,5	0%	[3: 8, 2: 12]
20403	2,3	100%	1,4,5	0%	[3: 10, 2: 10]
20404	3	50%	1,2,4,5	50%	[3: 12, 2: 6, 4: 2]

For example, in Table 6, the second verse is segmented into two sub-segments, labelled with 20201 and 20202. For segment 20201, using kNN model 13 out of 20 samples outputs only a single label 3 with accuracy. Therefore, the sample is weighted with label three in column "Label-1" with 83% accuracy. The frequency column helps to understand how each label is affecting the accuracy score. For example, in segment 20201, the frequency column shows "[3: 13, 4: 2, 2: 5]", which means that the sample is classified as contour label 3 for 13 occurrences, as label 4 for two occurrences, and as label 2 for 5 times. kNN output of 16% accuracy is recorded as the accuracy for the second label (Label-2).

Meanwhile, for segment 20202, the outcomes showed are label 3 with the accuracy of 0.5, the other labels are recorded in column "Label-2" for another 50%. This example explains that the higher the frequency of the label contains for the testing, the higher the accuracy score because it increases the chance for the KNN model to predict the label with a higher frequency correctly.

Another aspect of calculating the final grade is the weightage value. Weightage is calculated by dividing the 100 percent of each verse by its segments. For example, verse 1 has two sub-segments (20101 and 20102). Therefore, the weightage for each segment is approximately 50% each, while the weightage for each segment in verse 4 is approximately 25%. Table 7 shows the weightage values calculated for each of the eleven sub-segments on sample *surah Al-Ikhlas*.

Table 7: Weightage percentage for each segment

Segments	**Weightage**	**Label**	**Accuracy Score**	**Total Accuracy Score by Verse**
20101	50.00%	3	**0.5**	
20102	50.00%	2	**0.33**	**0.83**
20201	50.00%	4	0.08	
20202	50.00%	2	0.25	0.33
20301	33.00%	4	0.2178	
20302	33.33%	2	0.2178	
20303	33.33%	3	0.2739	0.7095
20401	25.00%	3	0.165	
20402	25.00%	2	0.25	
20403	25.00%	3	0.25	
20404	25.00%	3	0.125	0.79
			Overall Accuracy Score	66.49%

Based on Table 7, the total accuracy score for each verse of *surah Al-Ikhlas* is calculated by summation of all the accuracy values computed on the verse's sub-segments. The value for "Accuracy Score" shown for each sample is calculated based on frequency of occurrence for the corresponding label value. For example, the first segment of the first verse of *surah Al-Ikhlas*, segment 20101, is previously labelled as contour 3 with 100% accuracy. Multiplying the value with 50% as the weightage value resulted in an accuracy score of approximately 0.5 assigned to the segment 20101. The total accuracy score of each verse is the summation of the score from all its segments which is 0.83 (calculated as 0.5 for sub-segment 20101 added with 0.33 for sub-segment 20102).

Total score is calculated for all other verse in the collection, such as verse 2 has a total score of approximately 0.33, while verse 3 and 4 has a total score of approximately 0.71 and 0.79, respectively. Therefore, the average overall score for all the verses in sample 5 is valued at approximately 66.49%. In conclusion, this sample's overall score can be regarded as a good sample since the average score is higher than the pre-set overall passing grade (50%) and is assigned as grade B with a minimum of 60%.

4.2 Tarannum Training Application

In terms of the user interface, the system's theme is minimalist and user-friendly. This project aims to create an interactive web application that enables users to view and interact with it. This website is developed using CSS, HTML, and JavaScript. Fig. 4 and 5 below illustrate the output generated when users upload sample recitations.

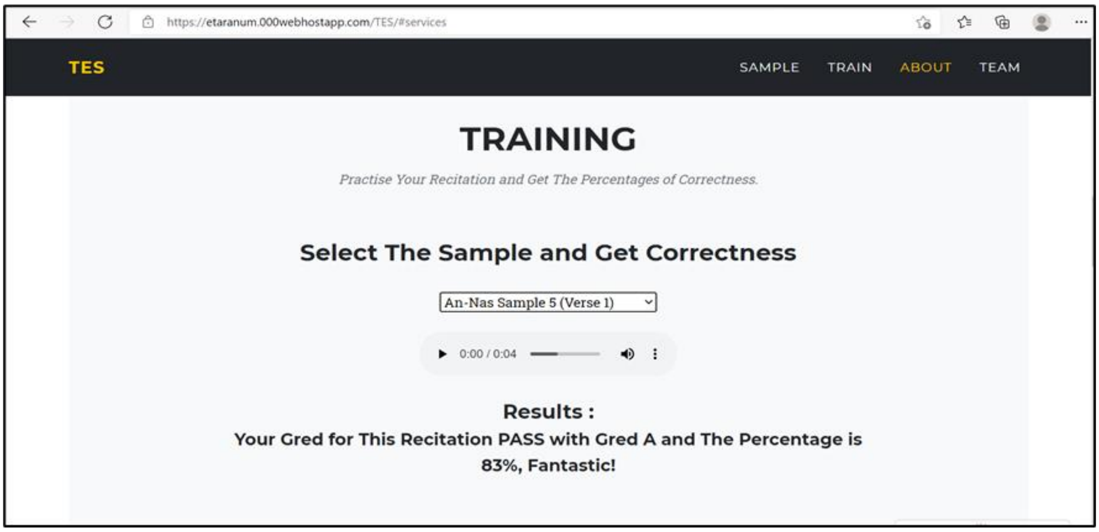


Fig. 4: Sample of a 'Pass' grade response produced by the prototype

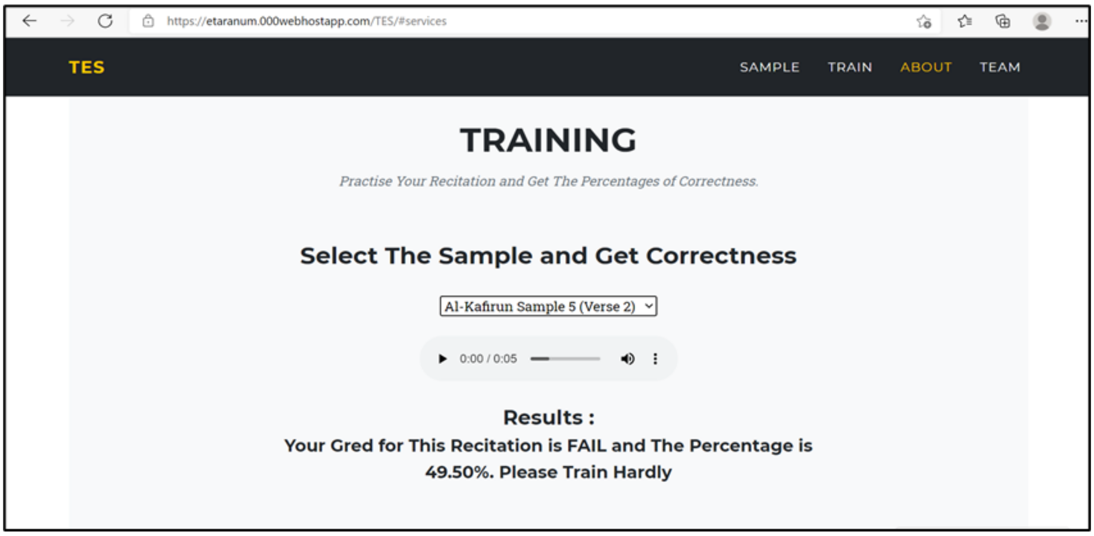


Fig. 5: Sample of a 'Fail' grade response produced by the prototype

Audio playback for both the reciter and the user recording is one of the features. In addition, the prototype enables users to compare the similarity of the sample audio provided by the reciter and the audio uploaded by users. Thus, the user can discern whether the melody of the uploaded file is similar or dissimilar to the sample's audio at a given timeframe. Apart from that, users will be able to upload as many samples as possible using two different upload options: verse-by-verse or entire surah-by-surah, to determine the accuracy of their recordings compared to the model. This application displays the percentage output and melodic recitation for audio samples uploaded by users.

In the current samples, the segmented verses are short, thus, the same label for verse contour occurs more frequently, resulting in the kNN model has a high probability of correctly predicting the same label. Increasing the number of longer verses and more reciters in the melody contour modelling stage can aid in the creation of a generic melody pattern for Tarannum. The current prototype is sufficient for beginners to practise on the melody of Bayati, and the application can recognise melody patterns modelled with a single reciter.

5.0 CONCLUSION

With a new evaluation of Tarannum melody, the project aims to improve knowledge of melody recognition. A tarannum collection is recorded with Bayati melodic recitation for the Quranic chapters *al-Kafirun*, *al-Ikhlas*, and *an-Nas*. The melody contours are then analysed and shape-based tarannum melody pattern is modelled based on the training samples. Pre-processing the audio sample is the first step, followed by feature extraction and contour construction, which includes labelling each verse segment with a contour label.

Then, the melodious recitations are defined using shape-based Tarannum-Tilt contour vectors containing the aggregated mean, standard deviation, slope of pitch values, and the contour label. A weighted accuracy score value is computed for each verse and a mean score measure shape-pattern similarity between any pairs of verses. As results, a 66% of weighted similarity score is reported with the current samples.

This project developed an interactive web tool for listening to and practising the Tarannum melody. Users can practise their recitations by continually listening to recorded recitations while employing the Tarannum melody. Additionally, they can evaluate their performance by examining the application's response score. A score of higher than 80% shows that the user is capable of accurately imitating the melody. A score of less than 50% indicates that the melody given by the user is not consistent with the expected pattern and that additional practise is necessary. Additionally, users can detect a change in their melody compared to the reciter at a particular point in time. The application pushes users to persevere until they master the Tarannum melody.

For future work, the recording collection can be expanded to more chapters, with more verses can be added to the collection. Also, number of reciters can also contribute to the melody patterns to generalise the tarannum contour model. Thus, kNN model can classify a more complex tarannum patterns from the longer verses and generic recitation signals.

ACKNOWLEDGMENT

The project is conducted at and supported by the Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia. The publications of this work are fully funded by Information Retrieval and Knowledge Management Society (PECAMP) and Yayasan Teknologi PETRONAS - Fundamental Research Grant (YUTP-FRG) (2019-2022).

REFERENCES

[1] M. A. Schmuckler, "Melodic Contour Similarity Using Folk Melodies". *Music Perception*, Vol. 28 No. 2, 2010, pp. 169–194.

[2] D. Müllensiefen and K. Frieler, "Evaluating Different Approaches to Measuring the Similarity of Melodies. Studies in Classification". *Data Analysis, and Knowledge Organization*, 2006, pp. 299–306.

[3] V. Makarand and K. Parag, "Unified Algorithm for Melodic Music Similarity and Retrieval in Query by Humming". *In Proceedings of 2nd International Conference Intelligent Computing and Information and Communication (ICICC 2017*), 2018, pp. 373-381.

[4] L. E. Caraballo, J.M. Díaz-Báñez, F. Rodríguez, V. Sánchez-Canales, and I. Ventura, “Scaling and compressing melodies using geometric similarity measures”, *in 36th European Workshop on Computational Geometry, Würzburg Germany, 16-18 March, 2020.*

[5] B. Stasiak, M. Skiba and A. Niedzielski, “FlatDTW - Dynamic Time Warping optimization for piecewise constant templates”. *Digit. Signal Process*, Vol 85, 2019, pp. 86-98.

[6] D. R. Ignatius Moses Setiadi et al., “Comparison of SVM, KNN, and NB Classifier for Genre Music Classification based on Metadata”, in *2020 International Seminar on Application for Technology of Information and Communication (iSemantic)*, 2020, pp. 12-16.

[7] G. Kour and N. Mehan , “Music Genre Classification using MFCC, SVM and BPNN”. *International Journal of Computer Applications*”, Vol. 112, 2015, pp. 12-14.

[8] M. A. Lubis, Y. Bila, H. R. Din, M. Mohamad and S. Sulaiman, “Challenges Faced by Teachers in Teaching Qur’anic Tarannum”. *World Applied Sciences Journal*, Vol. 14, 2011, pp.40-43.

[9] S. Haris@Harith, “Musicology in Islam and Qur'anic chanting (Tarannum)”. Master Degree dissertation, Interdisciplinary Islamic Studies, Syarif Hidayatullah State Islamic University, Jakarta, Indonesia, 2009.

[10] S. Haris@Harith, G. Shanmugavelu, and H. Abdul Bahar, “Musicology in Islam: A preliminary study”, *EPRA International Journal of Multidisciplinary Research (IJMR)*, Vol. 6, No.12, 2020, pp.206-213.

[11] W. H. Wan Abdullah, A. A. Sakat, S. Mohamad and E. A. Jamsari, “Meaning-based Tarannum: Preliminary research on uslub qira'ah of Sheikh Muhammad Rif'at (1880-1950)”. *Middle-East Journal of Scientific Research*, Vol 20, 2014, pp. 2172-2176.

[12] Z. Razak, N. J. Ibrahim, M. Y. I. Idris, E. M. Tamil, M. Y. Z. M. Yusoff, and N. N. A. Rahman, “Quranic verse recitation recognition module for support in j-QAF learning: A review”. *International Journal of Computer Science and Network Security*, Vol. 8, No. 8, 2008, pp. 207-216.

[13] M.R.G. Abbas, “Persepsi Murid Terhadap Minat, Amalan dan Persekitaran Pembelajaran Serta Amalan Pengajaran Guru Tarannum alQuran”. Master Degree dissertation, Universiti Kebangsaan Malaysia, Bangi, Malaysia, 2017.

[14] M. K. Latif, T. Jimaain and K. A.Jasmi, “Competence and Method of Teaching Tarannum Al-Quran Among Teachers of Special Class on Reading and Memorizing Al-Quran Skill (KKQ) in Johor”. *Advances in Social Science, Education and Humanities Research*, Vol. 400, pp. 249-258, 2020.

[15] M. Yenkimaleki and V. J. Heuven, “The relative contribution of computer assisted prosody training vs. instructor based prosody teaching in developing speaking skills by interpreter trainees: An experimental study”. *Speech Communication*, Vol. 107, 2019, pp. 48-57.

[16] N. J. Nik Ismail, *Buku Qawaid al-Tarannum, Cara berlagu*. 5th ed, Kuala Lumpur: Darul Fikir, 2012.

[17] T. Huang and Y. Yu, “Pitch and mode recognition of humming melodies”, 2018 International Conference on Audio, Language and Image Processing (ICALIP), 2018, pp. 366-370.

[18] R. Müller, S. Illium, F. Ritz AND K. Schmid, “Analysis of Feature Representations for Anomalous Sound Detection”, in *Proceedings of the 13th International Conference on Agents and Artificial Intelligence*, Volume 2: ICAART, 2021, pp. 97-106.

[19] N. M. Patil and M. Nemade, “Content-based audio classification and retrieval: A novel approach”, in *2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC)*, 2016, pp. 599-606.

[20] Z. Zixing, L Ding, H. Jing and S. Björn, “Learning Audio Sequence Representations for Acoustic Event Classification”. *Expert Systems with Applications*, Vol. 178, 2021, pp.1-11.

[21] A. Vidwans, P. Verma and P. Rao, "Classifying Cultural Music using Melodic Features," 2020 *International Conference on Signal Processing and Communications (SPCOM)*, 2020, pp. 1-5.

[22] B. Janssen, P. v. Kranenburg, and A. Volk, “A comparison of symbolic similarity measures for finding occurrences of melodic segments”, in *16th International Society for Music Information Retrieval Conference, ISMIR 2015*, 2015, pp. 659-665.

[23] W. Chai and B. Vercoe, “Folk music classification using Hidden Markov Models”, in *International Conference on Artificial Intelligence, Las Vegas, Nevada, USA, June 25 - 28. 2001*.

[24] T. Cheng, S. Fukayama, and M. Goto, “Comparing RNN parameters for melodic similarity”, in *19th International Society for Music Information Retrieval Conference, ISMIR*, 2018, pp. 763–770.

[25] Z. Jiang and R. B. Dannenberg, “Melody identification in standard MIDI files”, in *16th Sound & Music Computing Conference*, 2019, pp. 65-71.

[26] J. Urbano, J. Lloréns, J. Morato, and S. Sánchez-Cuadrado, “Melodic similarity through shape similarity”, in *(eds) Exploring Music Contents. CMMR 2010. Lecture Notes in Computer Science*, S., Ystad, M. Aramaki, R. Kronland-Martinet, K. Jensen, Berlin, Heidelberg, Springer Int. Publishing, 2011. pp. 338-355.

[27] M. Wu and X. Liu, “A Double Weighted KNN Algorithm and Its Application in the Music Genre Classification”, *2019 6th International Conference on Dependable Systems and Their Applications (DSA)*, 2020, pp. 335-340.

[28] J. Hong, H. Deng and Q. Yan, “Tag-based Artist Similarity and Genre Classification”, in *2008 IEEE International Symposium on Knowledge Acquisition and Modeling Workshop*, 2008, pp. 628-63.

[29] D. Kostrzewa, R. Brzeski, and M. Kubanski, “The classification of music by the genre using the KNN classifier”, in *Beyond Databases, Architectures and Structures. Facing the Challenges of Data Proliferation and Growing Variety. BDAS 2018. Communications in Computer and Information Science*, S. Kozielski, D. Mrozek, P. Kasprowski, B. Małysiak-Mrozek, D. Kostrzewa, Springer Cham, Switzerland: Springer Int. Publishing, 2018, pp. 233-241.

[30] Y. Zhu, M. S. Kankanhalli, and C. Zhu, “Pitch tracking and melody slope matching for song retrieval.” *in (eds) Advances in Multimedia Information Processing - PCM 2001 Lecture Notes in Computer Science*, H.Y. Shum, M. Liao, S.F. Chang, Berlin, Heidelberg: Springer Int. Publishing , 2001, pp 530-537.

[31] Y. Zhu, C. Xu, and M. Kankanhalli, “Melody curve processing for music retrieval”, in *IEEE International Conference on Multimedia and Expo*, ICME2001, 2001, pp. 285-288.

[32] N. M. Patil and M. U. Nemade, “Music genre classification using MFCC, K-NN and SVM classifier”. *International Journal of Computer Engineering In Research Trends*, Vol. 4, 2017, pp. 43-47.

[33] D. F. Silva, C. M. Yeh, Y. Zhu, G. E. A. P. A. Batista, and E. Keogh, “Fast similarity matrix profile for music analysis and exploration”. *IEEE Transactions on Multimedia*, Vol. 21, 2019, pp. 29-38.

[34] H. Mohamed Hanum, A. S. Aziz, Z. Abu Bakar, N. Mat Diah, W. F. Wan Ahmad, and N. Mohamad Ali, “Melody Training for Quranic Tarannum”, in *Fifth International Conference on Information Retrieval and Knowledge Management*, 2021, pp. 96-101.

[35] Y. E. Kim, W. Chai, R. Garcia, and B. Vercoe, “Analysis of a contour-based representation for melody”, in *International Symposium on Music Information Retrieval, (ISMIR 2000), Plymouth, Massachusetts, October 23-25, 2000*.

[36] P. Taylor, “The rise/fall/connection model of intonation”, *Speech Communication*, Vol. 15, 1994, pp. 169-186.

[37] P. Taylor, “The tilt intonation model”, in *Fifth International Conference on Spoken Language Processing, ICSLP’98, Sydney, Australia, November 30 - December 4, 1998.*

[38] D. O. Johnson and O. Kang, “Automatic prosodic tone choice classification with Brazil’s intonation model”. *International Journal of Speech Technology*, Vol. 19, 2016, pp. 95-109.

[39] S. Ananthakrishnan and S. Narayanan, “Fine-grained pitch accent and boundary tone labeling with parametric F0 features”, in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, Vol. 2008, 2008, pp. 4545-4548.