Assessing Vowel Quality for Singing Evaluation

Mayank Vibhuti Jha and Preeti Rao

Department of Electrical Engineering,
Indian Institute of Technology Bombay,
Mumbai 400076, India
Email: {mayankjha, prao} @ee.iitb.ac.in

Abstract

The proper pronunciation of lyrics is an important component of vocal music. While automatic vowel classification has been widely studied for speech, a separate investigation of the methods is needed for singing due to the differences in acoustic properties between sung and spoken vowels. Acoustic features combining spectrum envelope and pitch are used with classifiers trained on sung vowels for classification of test vowels segmented from the audio of solo singing. Two different classifiers are tested, viz., Gaussian Mixture Models (GMM) and Linear Regression, and observed to perform well on both male and female sung vowels.

Keywords: MFCC; GMM; Linear Regression; Vowel Quality; Singing Voice; Vowel Classification

1. Introduction

Singing or vocal music, like instrumental performances, is characterised by musical attributes such as melody and rhythm. However in the case of singing, also important are voice quality and the proper articulation of the lyrics. The automatic assessment of singing ability would therefore require processing the audio signal for the underlying acoustic attributes of pitch (related to melody), onsets (related to rhythm), phoneme quality (related to pronunciation) and timbre (related to voice quality). Such a system for singing assessment and feedback could be very useful both for music education and entertainment. Available systems for singing scoring, including popular karaoke games like SingStar [1] and UltraStar [2], are currently restricted to measuring pitch and timing accuracy with respect to a reference, i.e., only melodic and rhythmic aspects are considered. Our present work builds further on the same essential framework by incorporating new methods for the assessment of phoneme quality in singing.

The scenario under consideration has the singer rendering a known song while listening to the song's karaoke (i.e. background music) track. The acoustic characteristics of uttered phones are then evaluated with respect to the expected phones as provided by the song's lyrics. Our aim is to confirm whether the singer has rendered the lyrics accurately. Our aim is to develop a generalized system which should be text-independent. Once trained on sufficient number of vowel samples, it should be usable for testing vowels on any new song, provided the lyrics are known.

The current task is clearly related to Automatic Speech Recognition (ASR). However singing differs from speech in some important ways as presented in the next section. These differences warrant a separate study on features and classification methods for sung phones. In this paper we focus on sung vowel identification using a standard spectral representation and two different methods of classification. While GMM classifiers are widely applied in speech recognition, we also investigate a linear regression approach to classification that has certain advantages in the singing context [3].

2. Singing versus Speech

Singing, compared to speech, has a wider dynamic range in pitch as well as intensity due to the relative importance of expressiveness in singing. Singing tends to be a one-to-many communication at longer distances and hence the need to maintain a loudness balance across sounds [4]. Singing tends to have a higher percentage of sonorants than obstruents so that a singing piece will be largely composed of vowels. In fact, in singing, phonation time can be up to 95%, compared to 60% in normal speech [5]. Hence, restricting phoneme quality assessment to vowels is a reasonable starting point for pronunciation evaluation in singing. Due to the occurrence of high-pitched vowels in singing, it is possible that pitch harmonics do not coincide with the canonical formant locations in some cases. This usually causes singers to modify vowel quality in the interest of maintaining loudness. This dependence of vowel quality on pitch is another distinguishing factor between speech and singing.

3. Database

The data sets used in these experiments were chosen from a database of songs sung by various people in sing-along mode at the venue of a technical exhibition. As these songs were recorded in a public place (with moderate noise levels, SNR of the order of 20-30 dB), the database is representative of real-world scenarios. These songs (of about 1 min duration each) were recorded using a directional microphone, sampled at 8 kHz and stored in 16-bit PCM, mono channel, wave format.

Five popular Hindi movie songs each of male and female playback singers were selected for building the database.

Pitch range of male voices was from 117 Hz to 411 Hz, while that of the female voices was from 188 Hz to 613 Hz. The reference audio tracks were pitch-labelled throughout the vocal segments using a semi-automatic polyphonic pitch detector [6] followed by manual phone segmentation where all vowel instances were marked with the vowel name and segment boundaries.

On average, there were fifty vowel instances per song with five distinct Hindi vowels {/a/, /e/, /i/, /o/, /u/} appearing. Each of these songs was performed in sing-along mode by 8 different test singers. Sing-along was used in data collection although pure karaoke would work in practice. Thus, there were around 5*50*8 = 2000 vowel tokens (across all the vowels) for both male and female datasets. For use in training, all the vowel tokens in the singer audios were manually labelled in PRAAT [7].

3.1 Preprocessing

Test singers were asked to sing along while they listened to the reference song on headphones. This provided segments that were approximately time-aligned with the corresponding underlying lyrics. In this case, singers didn't have any visible cues such as text prompts. This resulted in non-uniform delays of up to 300ms in onset locations of the tokens compared to the corresponding onsets in the reference songs. These delays were longer at the start of the stanzas, and were observed to decrease as the singer caught up with the reference song. Instead of using computationally expensive dynamic time warping (DTW), the onset locations were corrected by using an automatic segmentation algorithm [8].

This algorithm marks the onset of a vowel by searching for rapid changes in selected frequency bands that characterise the formant structure of phones. For a given segment, the onset nearest in time to that in the reference song is selected. In case there is no onset detected within an adapted interval (based on distance from the past segment and the length of the current segment), the onset location is marked as that of the reference song. Based on the refined onsets and segment durations (from the metadata of pre-annotated reference song), the vowel segments are extracted for feature extraction and classification. All the extracted vowel segments were checked for validity via energy and voicing detection. Valid segments were pitch-labelled using a simple autocorrelation function based pitch detector.

3.2 Feature Extraction

Mel Frequency Cepstral Coefficients (MFCCs) were used as features in the vowel classification task. Similar to the widely used ASR features, 13 low-order MFCCs were extracted from the 8 kHz sampled audio waveforms at a frame-rate of 100/sec. Delta and acceleration features were not used. The analysis window was a Hamming window of length 30ms. Motivated by the known dependence of vowel timbre on the pitch, frame-level pitch frequency was included as an additional feature in the classification experiments involving the linear regression classifier. In the case of GMM classification, a 3-way pitch categorization into low/medium/high, as shown in Table 1, was used to train separate vowel models for each pitch category.

Table 1: Classification for Vowel Tokens according to average pitch values

Pitch Classification	Male	Female
Low Pitch	117 - 190 Hz	188 - 300 Hz
Medium Pitch	190 - 240 Hz	300 - 400 Hz
High Pitch	240 - 411 Hz	400 - 613 Hz

3.3 Tokens Selection for Training and Testing

In every song, each instance of a vowel was collected as a token. In some cases vowels, especially /i/ and /u/, were of very small duration. For these tokens, locating the onset and offset of vowels on the spectrogram was very difficult, due to rapidly changing formants locations in the spectrogram. In our database, for male voice out of the total 2497 tokens, 599 tokens (around 24 %) had less than or equal to 10 MFCC vectors (shorter than 132 ms). These tokens were excluded from the experiments described in the next sections. While creating the female database, annotations of tokens of such small durations were generally avoided.

In all the classification experiments for each gender, we used the entire set of 5 songs each sung by 8 test singers. A leave-one-song-out cross validation framework ensured sufficient data for training (4 songs x 8 singers) in each round. Further, the testing data in each round comprised of the one song not represented in the training set.

4. GMM based classification

A Gaussian Mixture Model is used to represent each vowel of the set of 5 distinct vowels. The GMM is trained on the entire training data set's vowel tokens of the specific vowel category. We have used single mixture GMM with full covariance matrix as a classifier. The dimension of each observation vector was equal to 13 (corresponding to 13 MFCC coefficients). Thus, there were 182 parameters (13 means + 13x13 covariance matrix), to be estimated from 2000 observations having 13 dimensions each, to create a model in each case. For the pitch dependent case, separate models were trained for each vowel-pitch category combination.

The classification involves detecting the vowel identity of the test token based on maximizing the likelihood of the observed feature vectors with respect to the models. This is equivalent to computing the Mahalanobis distance between the Gaussian model and the test vector.

Two experiments were performed on both male and female database. In the first experiment (using pitch information), three separate GMMs, corresponding to low, medium and high pitch, were built for each of the five vowels from all the songs in the training set. Then, each MFCC vector, from all the tokens of the test songs, was compared using Mahalanobis distance with all the fifteen

possible GMMs. Mahalanobis distance ensured that the covariance of each GMM distribution along all the dimensions was also factored in the distance from that GMM. If the vector was closest to any of the three GMMs of the ground truth vowel, then it was marked as correct identification. If more than half of the MFCC vectors of the test tokens were correctly identified, then this token was marked as correctly identified; otherwise it was marked as an error. For example, a test token of /a/ having low pitch is marked as correctly identified if more than half of its MFCC vectors are closest to GMMs of vowel /a/ of either of low, medium or high pitch in terms of Mahalanobis distance.

In the second experiment, pitch information was not used. Here, only five GMMs corresponding to five vowels were created by using tokens from all the three pitch classification. Here, an MFCC vector of the test token was compared with only these five GMMs and if this vector was closest to the GMM of the vowel corresponding to ground truth, in terms of Mahalanobis distance, then it was marked as correct identification. Again, if more than half of the MFCC vectors of the test tokens were correctly identified then the token was marked as correctly identified.

5. Linear Regression Classifier

Recently Frostel et al. have used linear regression [3] to project sung vowels on to a continuous articulatory space based on the IPA vowel chart. This method can be extended to vowel classification by dividing the articulatory space into regions corresponding to each of the vowel categories. Further, the position of the test token on the continuous articulatory space could give an additional degree of truth (fuzzy membership) about the quality of classification, telling us how reliable the classification was.

Multiple Linear Regression was used as a predictor in this experiment. The advantage of this method is its simplicity, robustness and efficient implementation for real-time prediction [3].

$$y = \sum_i x_i \beta_i + \varepsilon = \langle \boldsymbol{x}, \boldsymbol{\beta} \rangle + \varepsilon \qquad (1)$$

From equation (1), we can see that Multiple Linear Regression is basically the inner product of feature vector **x** (regressor) and regression coefficients vector **β** plus an error term ε. In this equation, **x** needs to be extended by one dimension to model the constant term.

5.1 Articulatory Space

As database used in this report comprises of Hindi songs, therefore a modified IPA vowel chart for Hindi has been used for linear regression, as shown in Figure 1.

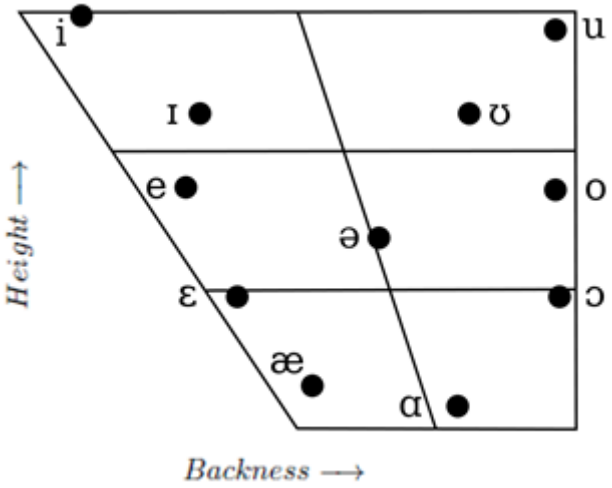

Figure 1: The IPA Vowel Chart for Hindi. The horizontal axis depicts the *vowel backness* and the vertical axis the *vowel height*. [9]

In order to build an articulatory space, this vowel chart was converted into a two-dimensional *Cartesian coordinate system* as shown in Figure 2. Only the five important vowels were retained. The proportions of the vowel chart used here were 2:3:4 for the bottom, right and top sides respectively [3]. The backness coordinates of this space range from 0 (front) to 4 (back), and the height coordinates range from 0 (open) to 3 (closed). Accordingly, separate regression model was generated for both these dimensions.

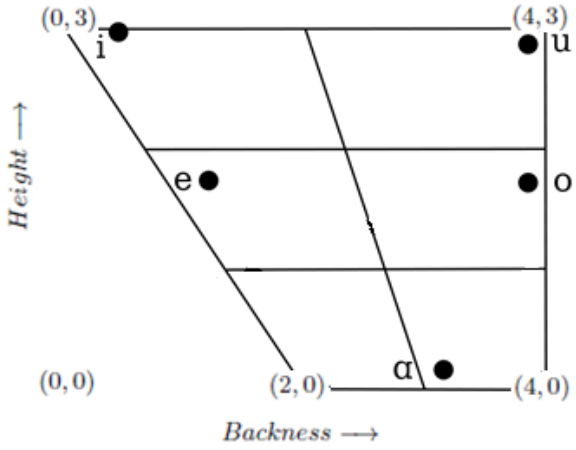

Figure 2: Coordinates of the vowel chart space used for regression. [3]

In this paper, experiments were done on five vowels as mentioned in Section 3. The vowels were mapped onto this space with the coordinates as described in Table 2.

Table 2: Coordinates of Vowels in the articulatory space

Vowels	Backness Coordinates	Height Coordinates
/ɑ/	3.2	0.2
/e/	1	1.6
/i/	0.3	3
/o/	3.8	1.6
/u/	3.8	2.8

5.2 Experiments

For *Multiple Regression Analysis*, two different algorithms were used – *regress* and *robustfit*. These are in-built functions in MATLAB.

- *regress* does simple linear regression
- *robustfit* does iterative linear regression. After each iteration, data points are assigned a weight based on their closeness to the generated model. As the outliers are given lesser weights in every iteration, their influence decreases and a more accurate model is generated.

Using both these methods, separate models were generated for male and female databases in both the dimensions (backness and height) using tokens from all the songs. This process was done twice with different sets of feature vectors – once using pitch values (F0 in Hz) in addition to the MFCCs, and once using only MFCCs as feature vectors. These methods were compared on the basis of Correlation Coefficients (r) and Root Mean Square Error (RMSE) values as shown in Table 3.

Table 3: Comparison of *regress* and *robustfit* algorithms for Male and Female Database

Male Database	**Features Sets: MFCC + F0**		**Features Sets: MFCC**	
	regress	***robustfit***	***regress***	***robustfit***
$\mathbf{r_b}$	0.8952	0.8942	0.8939	0.8938
$\mathbf{RMSE_b(\%)}$	15.2172	14.6641	15.3008	15.0147
$\mathbf{r_h}$	0.8885	0.8848	0.8782	0.8778
$\mathbf{RMSE_h(\%)}$	15.5440	14.8187	16.2010	16.1494

Female Database	**Features Sets: MFCC + F0**		**Features Sets: MFCC**	
	regress	***robustfit***	***regress***	***robustfit***
$\mathbf{r_b}$	0.8871	0.8870	0.8871	0.8870
$\mathbf{RMSE_b(\%)}$	15.4151	15.6810	15.4169	15.6912
$\mathbf{r_h}$	0.8350	0.8343	0.8350	0.8343
$\mathbf{RMSE_h(\%)}$	19.6731	19.9702	19.6730	19.9717

From Table 3, we can see that in most cases, performance of *robustfit* is in fact worse than *regress*. As *robustfit* is computationally intensive and takes about twenty-five times more time to build the model, *regress* was selected for doing the following experiments.

Again, leave-one-song-out cross-validation technique was employed to test the accuracy of the linear regression based model. Two different experiments were done on both male and female database. In the first experiment, 13 MFCCs and F0 constituted a single feature vector, and in the second, only 13 MFCCs constituted a feature vector. In each case the regression coefficients vector (β) in 2 dimensions was calculated using tokens from the four training set songs across all singers. Using this 2-dimensional β, mean and variance of the distribution of each training set vowel on the 2-dimensional vowel chart was calculated. Also, β was used to map the feature vectors of the test tokens on the 2-dimensional vowel chart by calculating their inner product with β. Based on the location estimated using this inner product, the feature vectors were classified as the vowel whose distribution was nearest to it in terms of Mahalanobis distance. If majority of the feature vectors of a test token were classified correctly with respect to the ground truth, then the token was also marked as being correctly identified; otherwise it was termed as an error. This process was repeated by making each song a test song once while rest of the songs were used as the training set.

6. Results and Discussion

From Table 4, we can see that the overall performance of the system is very encouraging. This table allows us to analyse the comparison of the different feature-classifier systems.

Table 4: Accuracy of GMM and LR based classifiers

Database (# Tokens)	**Features Sets**	**Accuracy (%)**	
		GMM	**LR**
Male (1898)	**MFCC + F0**	89.1	87.0
	MFCC	89.5	87.1
Female (1894)	**MFCC + F0**	80.0	76.4
	MFCC	79.5	76.6

Firstly, we can see that there is no significant difference in accuracy for the two features sets (with and without pitch information). Therefore, it is not possible to conclude from these results whether a priori pitch information contributes to the performance of the system. We may need to examine data with higher pitch range to reach a clear conclusion on this. Also, the performance of the system is almost 10% lower on female database in all cases. This may be due to higher contextual dependence and intra-token pitch variation in the vowel tokens in female database.

Secondly, we can see that the accuracy of the Linear Regression based classifier is around 2% to 3% lower than that of the GMM based classifier in all the cases. However the linear regression approach can provide additional information about the relative location of the test utterance in vowel space. This can be used to give more relevant feedback to the singer. An important advantage of the linear regression approach over the GMM approach is that the regression model can be used as such on a new language with a different set of vowels as long as the canonical positions of the vowels in the vowel space chart are known.

In this paper, vowels have been grouped into only five different classes. Normal, nasal and aspirated vowels were not differentiated while preparing training database resulting in large variances in the trained models. It was also observed that in both the methods, /a/, /e/ and /i/ vowels had the better accuracy than /o/ and /u/. We can visualize the distribution of various vowels on the continuous articulatory space by using the linear regression model. Table 5 lists the mean and standard deviation of each vowel in both the dimensions. From this

table we can see that the closeness of vowel pairs like /o/ - /u/ could have led to larger error rate.

Table 5: Coordinates of vowels in the articulatory space, with mean and the standard deviation

Vowels	Backness Coordinates	Height Coordinates
/ɑ/	3.22 ± 0.48	0.38 ± 0.39
/e/	1.08 ± 0.50	1.74 ± 0.38
/i/	0.85 ± 0.53	2.66 ± 0.39
/o/	3.41 ± 0.43	1.60 ± 0.35
/u/	2.80 ± 0.52	2.01 ± 0.38

A comparison of the computational complexity of the two classification methods is useful. With the GMM classifier, computation time for testing a single song of 1 minute duration, with total token duration of around 20 seconds, was about 1.35 second with different pitch based models and about 0.45 second with pitch independent models. With the linear regression classifier, a maximum computation time of around 0.53 second per song was noted indicating real-time application feasibility.

7. Future Work

There is a wide scope for future work on this application. Besides extending the work to all other phone classes, with a much larger training database, it would be possible to investigate HMM based phone recognition on sung audio. Short-duration vowels, consonants and nasality and aspiration detection can later be included in these models to improve the quality of feedback to the user. The system has been tested on 4 kHz bandwidth audio motivated by telephone speech. However it has not been tested for the typical degradations that arise in telephony. Finally, this work can be integrated with melody and rhythm detection system [10] to give a comprehensive feedback score to the user, which can be used for educational purpose as a music tutor or in the entertainment industry as a karaoke based game.

References

[1] SingStar: Karaoke game for Sony's PlayStation 2 & 3. It allows you to evaluate how good you are when you sing by analysing your voice pitch. {http://www.singstar.com/}

[2] UltraStar: Open source PC conversion of famous karaoke game SingStar. {http://www.ultrastargame.com/}

[3] H. Frostel, A. Arzt, and G. Widmer, "The Vowel Worm: Real-time Mapping and Visualisation of Sung Vowels in Music", Proceedings of the 8th *Sound and Music Computing Conference* (SMC 2011*)*, pp. 214 – 219, Padova, Italy, July 2011

[4] Ingo R. Titze, "Speaking Vowels versus Singing Vowels", *Journal of Singing*, pp. 47 – 48, Vol. 52, (September – October 1995)

[5] Alex Loscos, Pedro Cano and Jordi Bonada, "Low-Delay Singing Voice Alignment to Text", *Proceedings of the 1999 International Computer Music Conference*, Beijing, China, October 1999

[6] Sachin Pant, Vishweshwara Rao and Preeti Rao, "A Melody Detection User Interface for Polyphonic Music", Proceedings of *National Conference on Communications* (NCC 2010), Chennai, India, January 2010

[7] Praat: Free scientific software program for the analysis of speech in phonetics. {http://www.fon.hum.uva.nl/praat/}

[8] Pradeep Kumar, Manohar Joshi, Hariharan, S. Dutta-Roy and Preeti Rao, "Sung Note Segmentation for a Query-by-Humming System", Proceedings of *Music-AI (International Workshop on Artificial Intelligence and Music)* in IJCAI, 2007, Hyderabad, India.

[9] Jeffrey Connell, "Hindi Vowel Chart", retrieved from {http://en.wikipedia.org/wiki/File:Hindi_vowel_chart.svg}

[10] Chitralekha Gupta and Preeti Rao, "An Objective Assessment Tool for Ornamentation in Singing", *Proceedings of the International Symposium of Frontiers of Research on Speech and Music and Computer Music Modelling and Retrieval*, Bhubaneswar, India, March 2011