Spectral Features and Pitch Histogram for Automatic Singing Quality Evaluation with CRNN

Lin Huang*, Chitralekha Gupta† and Haizhou Li‡
*†‡ Electrical and Computer Engineering, National University of Singapore, Singapore
* lin.huang@u.nus.edu † chitralekha@nus.edu.sg ‡ haizhou.li@nus.edu.sg

***Abstract*—Deep neural networks (DNNs) have been applied successfully to music information retrieval (MIR). In this paper, we design a convolutional recurrent neural network (CRNN) for automatic singing quality evaluation, and present a comparative study over various acoustic features as network input. We optimize the CRNN so that the machine-predicted scores are closer to the human-annotated scores. Furthermore, we augment spectral features with pitch histogram (a musically-motivated representation) as network input. The experiments show that our proposed CRNN framework can learn the underlying discerning properties of singing quality effectively. Moreover, explicit incorporation of pitch histogram further improves system performance, and reduces the system's dependency on song content.**

I. Introduction

Singing is a popular entertainment and a desirable skill to develop. Traditional singing evaluation methods rely heavily on professional music teachers or experts, which is often inaccessible for ordinary people. Therefore, a system for automatic and reliable evaluation of singing quality would be useful for singing pedagogy, singing competitions, and karaoke systems.

Many studies on automatic music performance assessment utilize hand-crafted features to characterize different perceptual music parameters, such as intonation accuracy, rhythm consistency, and appropriate vibrato. These parameters are defined based on music knowledge and perceptual preferences, which are important for reliably predicting the overall singing quality [1]. Gupta et al. [2] proposed various perceptual features that measured the similarity between a test singing rendition and a reference singing rendition to help predict the overall singing quality. Nakano et al. [3] used pitch interval accuracy that measured the offset of the pitch values from the musical semitone grid to assess singing quality without a reference. Gupta et al. [4], [5] calculated statistical features that characterized the pitch histogram's shape to evaluate the intonation accuracy of a music piece. For capturing information about other perceptual parameters like rhythm and timbre, Gupta et al. [5] additionally designed inter-singer relative measures based on the concept of "veracity", that is, the singing vocals of good singers are similar while those of bad singers are different.

As hand-crafted features are extracted under simplified assumptions and rely on individual signal processing techniques, evaluation systems may draw conclusions from incomplete information. Deep neural network (DNN) is a feature learning approach for the effective characterization of meaningful features in complex and non-linear tasks. Ref. [6] showed that DNN-based methods can capture more relevant aspects of music than hand-crafted features. Therefore, in this study, we are motivated to use DNN models for feature learning and then evaluating singing quality reliably.

Convolutional neural networks (CNNs) are based on convolving the input with learnable kernels and are efficient at learning local features [7]. With their success in image classification tasks, CNNs have now been applied to audio processing tasks as well. For example, Hershey et al. [8] used AlexNet and VGGs for audio classification. Takahashi and Mitsufuji [9] used DenseNet for audio source separation. However, CNNs lack the ability to learn temporal dependencies, which is essential for modeling sequential data like audio, speech, or music. Recurrent neural networks (RNNs) are another type of neural networks that calculate the output of a time step from both the input of this time step and the hidden state of the previous step. This models the temporal dependency in the input. Moreover, RNNs can process the output of a CNN to form a convolutional recurrent neural network (CRNN). In this case, the early convolutional layers capture local information, and the recurrent layer summarises it along time. Some studies used CRNN for music information retrieval (MIR) tasks such as music transcription [10], music classification [11] and music emotion recognition [12].

The success of deep learning methods relies on the design of network architecture and feature representation. Zhang et al. [13] created a CNN architecture named Bi-DenseNet processing fft-spectrograms to discriminate the good singings from the poor singings. Pati et al. [6] trained a fully convolutional neural network on pitch contours and a CRNN model on Mel-scaled spectrograms (Mel-spectrograms) to assess music performances of pitched wind instruments. Wang and Tzanetakis [14] utilized CNNs in a siamese architecture trained on both Mel-spectrograms and constant-Q transformed spectrograms (CQT) to investigate singing style.

In this study, we would like to apply deep neural networks to singing quality evaluation. We adopt a CRNN architecture to learn features from input and predict evaluation scores of singers. To fine-tune the input representation, we compare the system performance among three types of spectral features, i.e. Mel-spectrogram, CQT and chromagram. In addition, we propose incorporating pitch histogram, a musically-motivated representation, as a conditioning vector, in the neural network.

This paper is structured as follows. In Section II, we

introduce the audio data representations. Section III describe the neural network architectures employed in this study. The experimental setup and results are discussed in Section IV and V. Section VI concludes the study.

II. Audio Data Representations

We study different methods of audio and music motivated acoustic features. As we aim to develop a singing quality evaluation system, we consider spectral features that characterize singing quality, such as Mel-spectrogram, CQT, and chromagram. In addition, we use a combination of spectral features and pitch histogram as network input to capture both spectral and prosodic patterns of singing.

A. Mel-spectrogram

Mel-spectrogram is a time-frequency representation that is optimized for human auditory perception (Fig. 1(a)). It consists of two concepts: the Mel-scale and the spectrogram. The spectrogram is a bunch of fast Fourier transforms (FFTs) stacked on top of each other. It varies with time at different frequencies and therefore can visually represent the amplitude of audio signals. The Mel-scale is the result of some nonlinear transformations of the frequency scale, which is consistent with known human perception [15].

Mel-spectrogram can preserve perceptually important information and therefore has been popular in many audio processing tasks, such as automatic tagging [16], onset detection [17], and learning features of music recommendation [18].

B. Constant-Q Transform (CQT)

CQT provides a time-frequency representation with logarithmic-scale center frequencies (Fig. 1(b)). It employs constant-Q transform instead of FFT to transform audio signals from time domain to frequency domain. Constant-Q transform uses geometrically spaced frequency bins to ensure that the Q factors (the ratio of the center frequencies to bandwidths) of all bins are constant [19]. This makes CQT well suited for music data, since the Q factor is approximately constant in most of the audible frequency range of the human perception system, and the fundamental frequencies of the tones in Western music are geometrically spaced along the standard 12-tone scale [19].

CQT is essentially a wavelet transform, which means that the frequency resolution is better at low frequencies and the temporal resolution is better at high frequencies. Therefore, it can capture essential audio information from both low and high frequencies in sufficient resolution, and has been widely used in music signal processing [20].

C. Chromagram

Chromagram (also called the pitch class profile) provides a 2D representation of the energy distribution over a set of pitch classes (often 12 pitches in Western music) (Fig. 1(c)) [21], [22]. Compared with Mel-spectrogram and CQT, chromagram is rarely used as input of the neural network. But it can efficiently capture the existence of each tone in a short-time music segment, which is useful for the harmonic and melody characterization of music signals. Birajdar and Patil [23] explored the features related to music tones from the chromagram for the speech/music classifcation.

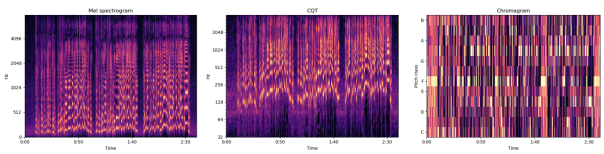


Fig. 1. Three types of spectral features of the same singing rendition: (a) Mel-spectrogram (x-axis is time in seconds, y-axis is frequency with Mel-scale in Hz), (b) CQT (x-axis is time in seconds, y-axis is frequency with logarithmic-scale center in Hz), (c) chromagram (x-axis is time in seconds, y-axis is pitch class).

D. Pitch Histogram

The spectral features, namely Mel-spectrogram, CQT and chromagram, as shown in Fig. 1, capture music-related information. Meanwhile, pitch histogram is also an effective indicator of singing quality [5]. Pitch is an auditory sensation, and pitch correctness is important for a good singer. The pitch histogram is a global statistical representation of the pitch content of a music composition, representing the distribution of pitch values in a music piece [24]. A pitch histogram is computed as the count of pitch values (in the units of cents) folded over 12 semitones in an octave (one semitone represents 100 cents on equi-tempered octave). To compute the pitch histogram from the input audio, we first extract the pitch contour (in Hz), and convert it to an equi-tempered scale (in the unit of cents) using the following equation:

$$f_{cent} = 1200 \times log_2 \frac{f_{Hz}}{440} \tag{1}$$

where f_{Hz} is the pitch value in Hz, 440 Hz (pitch-standard musical note A4) is considered as the base frequency, f_{cent} is the resulting pitch value in cents.

Then, we subtract the median of obtained pitch values (in the unit of cents) in a singing rendition to remove the key of the song, and then transpose all pitch values to a single octave. Next, the pitch histogram H is calculated by placing the pitch values into their corresponding bins [25]:

$$H_k = \sum_{n=1}^{N} m_k \tag{2}$$

where H_k is the k^{th} bin count, N is the number of pitch values, $m_k = 1$ if $c_k \leq P(n) \leq c_{k+1}$ and otherwise $m_k = 0$, where $P(n)$ is the n^{th} pitch value in an array of pitch values and (c_k, c_{k+1}) are the bounds on k^{th} bin.

Fig. 2 gives the pitch histogram of two different levels of singers (both performing the same song). To obtain a fine histogram representation, we divide each semitone into 10 bins. Therefore, for each pitch histogram, we have 12 semitones x 10 bins = 120 bins in total (each bin represents

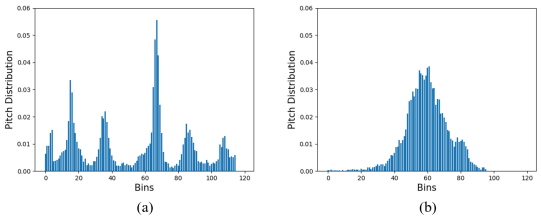


Fig. 2. Normalized pitch histogram of (a) a good singer and (b) a poor singer. (1 bin = 10 cents).

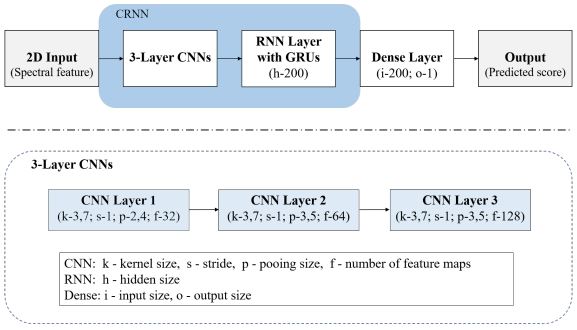


Fig. 3. CRNN model with spectral features as input.

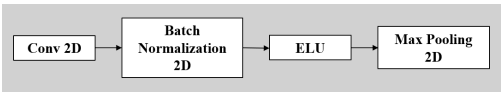


Fig. 4. CNN sub-structure.

10 cents). We also normalize the pitch histogram to scale the range of its y-axis in [0, 1].

A song's melody usually consists of a set of dominant notes (or pitch values), which are hit frequently in the song and sometimes last a long time. In the pitch histogram of a good singer, there are several sharp and narrow peaks, suggesting that these dominant notes are hit more frequently and consistently than the rest of pitch values (Fig. 2(a)). In other words, this singer sings in tune. On the other hand, the poor singer has a dispersed distribution of pitch values, reflecting that this singer cannot consistently hit the song's dominant notes (Fig. 2(b)).

As pitch histogram has strong relevance to "sing-in-tune" quality, we propose incorporating pitch histogram to the neural network. In this way, the network learns music-related information, such as timbre and rhythm, from the spectral features, and "sing-in-tune" quality from pitch histogram.

III. Neural Network Architecture

Convolutional recurrent neural network (CRNN) takes advantage of CNN for local feature extraction and RNN for temporal summarization of the features extracted by CNN. In this work, we train CRNN models on different kinds of spectral features mentioned in Section II. In addition, we incorporate the pitch histogram as a conditioning vector to the neural network, which we call a hybrid CRNN.

A. CRNN with Spectral Features

The CRNN used in this paper is motivated by the structure for assessing music performances of pitched wind instruments [6]. The network consists of 3-layer CNNs, 1 recurrent layer and 1 fully-connected dense layer, as shown in Fig. 3. Each CNN sub-structure has 4 components: (i) a 2D convolutional layer, (ii) a 2D batch normalization layer, (iii) an exponential linear unit (ELU) activation function, and (iv) a 2D max-pooling layer, as shown in Fig. 4. The 2D input representation is fed into the first convolutional layer and abstracted to a feature map. Batch normalization and max-pooling can control overfitting during training. The last CNN is followed by a RNN with gated recurrent units (GRUs). Compared with other RNN units like long short-term memory (LSTM), GRUs are simpler to implement and are equally well suited to capture long-term dependencies [26], [27]. We remove the last ReLU activation function behind the dense layer in [6], since the range of our manual annotations is [-1, 1]. Then, the hidden state of the last GRU is passed to a fully-connected linear layer to directly obtain the predicted scores of singers.

The spectral features like Mel-spectrogram, CQT and chromagram are directly fed into the CRNN architecture. Therefore, we build 3 CRNN models: (i) Mel-CRNN using Mel-spectrogram as input, (ii) CQT-CRNN using CQT as input, and (iii) Chro-CRNN using chromagram as input.

B. CRNN Conditioned on Pitch Histogram

The pitch histogram is a musically-motivated acoustic feature that encodes pitch accuracy information. However, it is not straightforward to use the pitch histogram directly as the input of the neural network because it takes a more compressed form than spectral features. In addition, the input of the network framework used in this paper should be a 2D representation, while the pitch histogram is a 1D audio representation. Therefore, we condition the CRNN on pitch histogram by concatenating the pitch histogram vector with the output vector of its intermediate layer (the recurrent layer), as shown in Fig. 5. We insert the pitch histogram here, because the output here has dimensions comparable to the histogram. Then, the concatenated feature vector is passed to the dense layer to obtain the output of the model. Such a configuration, which we call the hybrid CRNN, aggregates the features learned by the original CRNN along with the pitch accuracy related information captured by the pitch histogram, thereby improving the discrimination ability of the network for singing quality evaluation.

We construct 3 hybrid CRNN models: (i) MPH-CRNN using Mel-spectrogram and pitch histogram as input, (ii) CPH-CRNN using CQT and pitch histogram as input, and (iii) ChPH-CRNN using chromagram and pitch histogram as input.

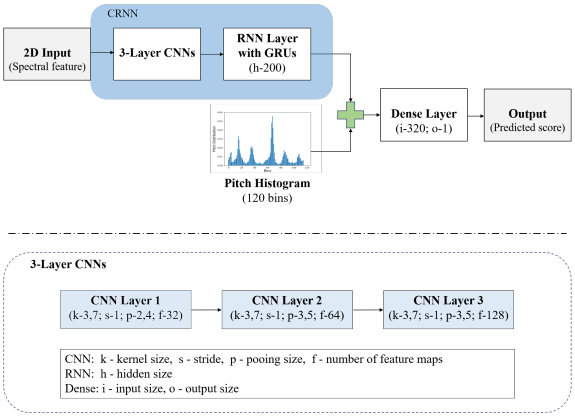


Fig. 5. Hybrid CRNN model conditioned on pitch histogram.

IV. Experimental Setup

We conduct experiments to evaluate the performance of CRNN and hybrid CRNN for automatic singing quality evaluation. We also evaluate the effect of different input representations. We further compare the performance of the proposed system with prior work about music assessment in literature. Moreover, we design the "leave-one-song-out" experiment to test the framework on unseen songs and visualize the features learned by the neural network.

A. Dataset

1) Singing Voice Dataset: We use the dataset used by [5] that consists of solo-singing recordings of 4 popular Western songs (*Let it go (Idina Menzel), Cups (Anna Kendrick), When I was your man (Bruno Mars), Stay (Rihanna)*) in Smule's DAMP dataset[1]. Each song is sung by 100 unique singers, including 50 males and 50 females, which can avoid gender bias. There were no common singers in different songs. For data augmentation, we divide every singing rendition into 5 snippets, where each snippet is of 20-30 seconds in duration, and full-length audio files are also used. Therefore, there are 2400 audio files in total.

The experiment consists of two phases: (i) training and validation phase to train and determine the model; and (ii) testing phase to evaluate the final model. Therefore, we divide the dataset into training, validation, and test sets. For each song, we select 80 singers for training, 10 singers for validation and 10 singers for testing. We first rank the 100 singers according to their ground-truth scores, i.e. rank 1 represents the best singer with the highest subjective score, and rank 100 represents the worst singer with the lowest subjective score. Next, we pick the singers with ranks [1,11,$\cdots$,81,91] as the test singers, the singers with ranks [2,12,$\cdots$,82,92] as the validation singers, and the rest are used for training models. Such a data configuration method ensures similar distribution of singing quality in all of these subsets. Note that the test singers are not in training or validation sets, but the songs in the three sets are the same. The dataset division is summarized in Table I. There are 4 songs in the audio dataset, and each singer's singing voice is represented by 6 audio files (5 audio snippets plus the entire file), so there are a total of 1920 audio files for training, 240 for validation, and 240 for testing.

[1] https://ccrma.stanford.edu/damp/

TABLE I
Summary of the number of audio samples in training, validation and test sets

Dataset division	# Songs	# Singers per song	# Snippets	# Total samples
Training	4	80	6	$4 \times 80 \times 6 = 1920$
Validation	4	10	6	$4 \times 10 \times 6 = 240$
Test	4	10	6	$4 \times 10 \times 6 = 240$

2) Ground-Truth Scores: The ground-truth subjective ratings (manual annotations) provided with the dataset in [5] were Best-Worst Scaling (BWS) scores obtained from pairwise BWS tests on Amazon mechanical turk (MTurk, one crowdsourcing platform), by asking listeners to choose the better singer between a pair of singers singing the same song. The BWS score is defined as follows:

$$B = \frac{n_{best} - n_{worst}}{n} \tag{3}$$

where n_{best} and n_{worst} are the number of times a singer is marked as preferable and otherwise, and n is the total number of times the singer appears in the pairwise BWS tests.

The BWS score is reliable since people are better at relative judgments, i.e. choosing the preferable singer between a small set of singers, rather than giving an absolute rating [28], [29]. For further improving the reliability of the pairwise BWS test, some rules were set to screen MTurk users. For example, the accepted attempts required users to have some formal training in music and be able to write the musical notations successfully, and the less serious attempts (users did not finish listening to snippets) would be removed through monitoring the time spent by MTurk users in performing the test.

B. Input Representation Computation

The spectral features (Mel-spectrogram, CQT and chromagram) are calculated with Librosa. For Mel-spectrogram, the window length and hop size are set to 2048 and 1024 respectively, and the number of Mel bins is fixed to 96. For CQT, the hop size is set to 512. There are 96 bins per CQT and 24 bins per octave to capture sharp/flat pitches. The calculated Mel-spectrogram and CQT are both squared and then scaled into decibels (dB). For chromagram, the hop size is 512 and the number of chroma bins is 96.

For computing the pitch histogram, we extract the pitch using the autocorrelation-based pitch estimator PRAAT [30], [31] with a hop size of 10 ms, and calculate the pitch histogram as described in Section II-D. We have 120 bins in total (12 semitones x 10 bins = 120 bins) for each pitch histogram, and we use the min-max normalization method to scale the range of the pitch histogram (values on the y-axis) in [0, 1].

C. Training of Networks

1) Network Configuration: The hybrid CRNN (Fig. 5) is a modified version of the CRNN (Fig. 3). These two neural network frameworks have the same hyperparameters of the 3-layer CNNs and the RNN layer. The number of feature maps increases from 32 in the first CNN to 128 in the third CNN. The hidden size of the GRU is 200. For the dense layer of the basic CRNN model (Fig. 3), the input size is 200 and the output size is 1. For the hybrid CRNN model (Fig. 5), the input size of its dense layer is changed to 320, since we concatenate the 200-dimensional output vector of the RNN layer with the normalized 120-dimensional pitch histogram vector before the dense layer.

2) Training: The neural network framework for the training process is developed using PyTorch. We use the adaptive moment estimation (Adam) optimization algorithm with a learning rate of 0.0001 to update the parameters and the mean square error (MSE) as the loss function. The batch-mode is used to train the model, which means dividing the dataset into small batches and updating the model parameters based on the MSE calculated in each batch. All spectral features in a mini-batch are matched to the same sequence length (the longest length in that mini-batch) by zero paddings. The batch size is 10 for the model using Mel-spectrogram as input and 5 for the model using CQT/chromagram as input. The maximum number of epochs is set to 250 for the model using Mel-spectrogram/CQT as input and 100 for the model using chromagram as input. We select the trained model from the epoch that shows the best performance on the validation set.

D. Evaluation Metrics

We utilize two types of statistical metrics (calculated using Scipy) to evaluate the performance of the framework: **(i) Pearson correlation coefficient:** Measure the correlation between the scores predicted by the machine and the BWS scores annotated by humans. **(ii) Spearman's rank correlation coefficient:** Measure the correlation between the rank-order obtained from the machine-generated scores and the rank-order obtained from the manually-annotated BWS scores.

V. Experiments and Results

A. Performance of CRNN with Various Spectral Features

The manual annotations (human-annotated BWS scores) are reliable and can be considered as "development" data. By comparing the correlation between the output results generated by the CRNN model (Fig. 3) and the manual annotations, we can fine-tune (or select) the better input representation for the CRNN to learn. In this experiment, we use 3 types of spectral features as input to train the CRNN model, and then obtain 3 CRNN models: Mel-CRNN, CQT-CRNN and Chro-CRNN, as described in Section III-A.

From Table II, we see that all 3 models can converge on the training set with a high correlation. While on the test set, we observe that the CQT-CRNN outperforms the other 2 CRNN models. This implies that CQT can capture the underlying aspects of singing quality more effectively than Mel-spectrogram and chromagram. Therefore, we select the CQT spectrogram as the better input representation.

TABLE II
Performance of the CRNN framework with different spectral features as input in terms of Pearson and Spearman correlation between machine-predicted scores and human-annotated BWS scores (Mel-CRNN uses Mel-spectrogram as input, CQT-CRNN uses CQT as input, Chro-CRNN uses chromagram as input)

Model	Pearson correlation			Spearman correlation		
	Train set	Vali. set	Test set	Train set	Vali. set	Test set
Mel-CRNN	0.99	0.68	0.56	0.99	0.67	0.56
CQT-CRNN	0.99	0.72	**0.72**	0.99	0.72	**0.73**
Chro-CRNN	0.98	0.73	0.70	0.98	0.73	0.70

TABLE III
Performance of hybrid CRNN models: (i) MPH-CRNN using Mel-spectrogram and pitch histogram as input, (ii) CPH-CRNN using CQT and pitch histogram as input, and (iii) ChPH-CRNN using chromagram and pitch histogram as input

Model	Pearson correlation			Spearman correlation		
	Train set	Vali. set	Test set	Train set	Vali. set	Test set
MPH-CRNN	0.99	0.64	0.63	0.99	0.65	0.61
CPH-CRNN	0.99	0.73	**0.76**	0.99	0.73	**0.76**
ChPH-CRNN	0.99	0.69	0.73	0.99	0.71	0.74

B. CRNN Vs. Hybrid CRNN

CRNN is designed to learn the discriminatory characteristics of singing quality from the spectral features. The hybrid CRNN is expected to outperform CRNN, because pitch histogram reflects "sing-in-tune" quality, as discussed in Section II-D. In this experiment, we train 3 hybrid CRNN models: MPH-CRNN, CPH-CRNN and ChPH-CRNN (as described in Section III-B) to test our hypothesis that the hybrid CRNN conditioned on the pitch histogram can improve the network performance.

From Table III, we see that all 3 hybrid CRNN models converge on the training set, and on the test set, the performance of the model with CQT as input (CPH-CRNN) is still better than the other 2 models. We compare the performance of the hybrid CRNN and the CRNN on the test set. Then, as Table IV shows, the hybrid CRNN outperforms the CRNN. This means that explicitly encoding pitch accuracy related information via the pitch histogram supports the network to learn other aspects of singing quality from the spectral features, hence conditioning the network on such acoustic feature (pitch histogram) can boost the system performance.

TABLE IV
Comparison of the performance of CRNN with that of Hybrid CRNN on the test set

	Model	Pearson corr.	Spearman corr.
CRNN	Mel-CRNN	0.56	0.56
Hybrid CRNN	MPH-CRNN	0.63	0.61
CRNN	CQT-CRNN	0.72	0.73
Hybrid CRNN	CPH-CRNN	**0.76**	**0.76**
CRNN	Chro-CRNN	0.70	0.70
Hybrid CRNN	ChPH-CRNN	0.73	0.74

C. Performance of Model with Only Pitch Histogram

Hybrid CRNN combines spectral features and pitch histogram as input, improving the performance of singing quality evaluation system. However, it is uncertain whether it is the pitch histogram or the combination of spectral features and pitch histogram that improves system performance. Therefore, we conduct an experiment that uses only the pitch histogram as input to train a model and evaluate the singing quality. As described in Section III-B, the pitch histogram cannot be as input of CRNN since it is a 1D audio representation. Therefore, we pass the pitch histogram directly to the dense layer. That is, we remove the CRNN part from the hybrid CRNN model (Fig. 5), and modified the input size of the dense layer from 320 to 120. The other configuration is the same as the hybrid CRNN.

The Pearson correlation coefficient and Spearman's rank correlation coefficient of this model on the test set are **0.39 (Pearson)** and **0.37 (Spearman)**. Therefore, we can conclude that it is the combination of spectral features and pitch histogram that improves system performance. The pitch histogram helps capture information related to pitch accuracy, while spectral features can learn other music-related information, such as rhythm and timbre.

D. Comparison with Prior Studies

The previous studies of Gupta et al. [5] and Pati et al. [6] are similar to ours. Gupta et al. explored various hand-crafted features from the pitch histogram to generate a rank-order of singers. Pati et al. trained a CRNN model that used Mel-spectrogram as the input representation to assess music performances of pitched wind instruments. In this experiment, we compare the performance of our proposed hybrid CRNN model CPH-CRNN (using CQT and pitch histogram as input) with the absolute scoring system using hand-crafted features (computed from pitch histogram) of [5]. Our framework is similar to the absolute scoring system of [5] because both are direct evaluations of singing quality. Additionally, we train the CRNN model of [6] on our dataset. This model is also our previously trained model Mel-CRNN (see Section V-A).

In Table V, we see that our proposed hybrid CRNN performs better than the absolute scoring system of Gupta et al. [5]. This implies that the neural network conditioned on the pitch histogram can capture more discriminatory information about singing quality than hand-crafted features. Moreover, the hybrid CRNN outperforms the work of Pati et al. [6]. This means that the neural network performs better by using the combination of the CQT spectrogram and pitch histogram as input rather than using only the spectrogram.

E. Cross-Validation

Since our test set is small (only 240 samples, see Table I), we perform cross-validation to take advantage of all audio samples. In this experiment, the cross-fold 1 is of the same data configuration as described in Section IV-A1. The other folds are set in a similar way, e.g. for cross-fold 2, the singers with ranks [2,12,$\cdots$,82,92] are test singers, the singers with ranks [3,13,$\cdots$,83,93] are validation singers, the rest are training singers. Note that for cross-fold 9, the test singers are ranked in [9,19,$\cdots$,89,99], the validation singers are ranked in [0,10,$\cdots$,90]. We conduct this experiment on the hybrid model CPH-CRNN which has the highest correlation in this work (0.76, see Table IV). We also compute the average result of the 9 folds.

In Table VI, we observe that the results of all folds (including the average result) have slight difference, and each Spearman's rank correlation is higher than that of the work in [5] and [6], as given in Table V. Therefore, the correlation coefficients of the hybrid CRNN model CPH-CRNN in Table IV are reliable.

TABLE V
COMPARISON OF THE PERFORMANCE OF OUR PROPOSED HYBRID CRNN WITH THAT FROM PREVIOUS WORK ON THE SAME DATASET

Framework	Model description	Spearman corr.
Gupta et al. [5]	The absolute scoring system with hand-crafted features computed from pitch histogram	0.48
Pati et al. [6]	The CRNN model using Mel-spectrogram as input, i.e. model Mel-CRNN	0.56
This work	The hybrid CRNN model using CQT and pitch histogram as input, i.e. model CPH-CRNN	0.76

TABLE VI
PERFORMANCE OF THE HYBRID CRNN MODEL CPH-CRNN IN CROSS-VALIDATION

Cross fold	1	2	3	4	5	6	7	8	9	Average
Pearson corr.	0.76	0.73	0.73	0.69	0.67	0.68	0.71	0.66	0.66	0.70
Spearman corr.	0.76	0.72	0.73	0.66	0.67	0.68	0.71	0.65	0.64	0.69

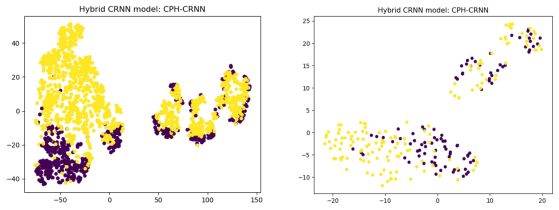
F. Evaluation on Unseen Songs

As mentioned in Section IV-A1, the songs in 3 sets (training, validation and test sets) are the same. To test whether the proposed model is song-independent, we need to test the performance of the trained model on unseen songs, i.e. the songs in the test set are not in the training and validation sets. Therefore, we conduct the "leave-one-song-out" experiment, i.e. leave one of the 4 songs to test the trained model, and the remaining 3 songs are used to train and tune the neural network. We perform this experiment on both the CRNN model (CQT-CRNN) and the hybrid CRNN model (CPH-CRNN). Since there are 4 songs in the dataset, we need to perform this experiment 4 times and then compute the average result.

From Table VII, we see that hybrid CRNN outperforms CRNN. This implies that the pitch histogram is a powerful acoustic feature of singing quality which reduces the dependence of the network on the song content.

TABLE VII
EVALUATION OF THE CRNN AND THE HYBRID CRNN MODEL ON UNSEEN SONGS

Framework	Average Pearson corr.	Average Spearman corr.
CRNN (CQT-CRNN)	0.48	0.48
Hybrid CRNN (CPH-CRNN)	0.56	0.56



(a) (b)

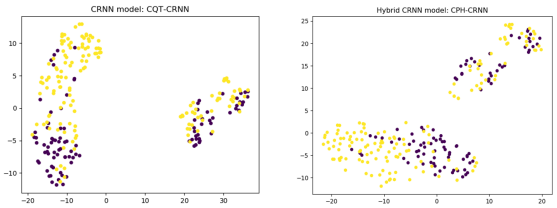
Fig. 6. Feature visualization of the hybrid model CPH-CRNN (using CQT and pitch histogram as input) on (a) the training set and (b) the test set. Purple points represent good singers, and yellow points represent poor singers.

G. Feature Visualization

The neural network proposed in this study learn features that can distinguish singing quality. For CRNN, the feature vector is of 200 dimensions, and for hybrid CRNN, it is 320-dimensional, as mentioned in Section IV-C1. To visualize how learned features capture the singing characteristics of individual singers, we first extract these features before the final dense layer, and then project them onto a 2D space using t-SNE [32]. The t-SNE plots often show clustering (which means that the samples belonging to the same category will cluster together), so we need to label each sample based on their ground-truth BWS scores (range in [-1, 1]), i.e. singers with BWS scores higher than 0.25 are good singers, and the rest are poor singers. The visualization is shown in Fig. 6 and Fig. 7.

In Fig. 6, we extract the features learned by CPH-CRNN (the hybrid CRNN using CQT and pitch histogram as input) from the training set and test set. The purple points represent good singers, and the yellow points represent poor singers. We see that the feature vectors from the training set and test set can cluster singers of the same label together. The size of the training set is larger than the test set, and the correlation of the training set is higher than the test set, so the clustering effect is more obvious on the training set.

In Fig. 7, we visualize the learned features extracted from the CRNN model (CQT-CRNN) and the hybrid CRNN model (CPH-CNN) on the test set. We observe that both CRNN and hybrid CRNN have the ability to cluster singers belonging to the same category together. This further proves that the features learned by our proposed neural network frameworks can well capture discriminatory information about singing quality.



(a) (b)

Fig. 7. Feature visualization of (a) CRNN model and (b) hybrid CRNN model on the test set.

TABLE VIII
COMPARISON OF THE PERFORMANCE OF MODEL USING THREE SPECTRAL FEATURES WITH MODEL USING ONE SPECTRAL FEATURE

	Model	Pearson corr.	Spearman corr.
CRNN	CQT-CRNN	0.72	0.73
	Model fusion	0.76	0.76
Hybrid CRNN	CPH-CRNN	0.76	0.76
	Model fusion	0.78	0.78

H. Model Fusion

This paper explored three types of spectral features: Mel-spectrogram, CQT and chromagram. Both the Mel-spectrogram and CQT are time-frequency representations of audio, which can preserve perceptual information. The chromagram can be regarded as a CQT folding in the frequency axis [33]. Since various spectral features cause different evaluation results, we conduct an experiment to combine models with the three spectral features. For CRNN, we have three models: Mel-CRNN, CQT-CRNN and Chro-CRNN. Each model generates a score for every singer. We calculate the average of the scores obtained from the three CRNN models as the result of model fusion. For Hybrid CRNN, we also have three models: MPH-CRNN, CPH-CRNN and ChPH-CRNN, and compute the average score from the three hybrid CRNN models.

From Table VIII, we see that model fusion improves the evaluation results for both CRNN and Hybrid CRNN with only one spectral feature (CQT). Therefore, model fusion can aggregate information captured by different types of spectral features.

VI. CONCLUSIONS

In this work, we build a CRNN framework for automatic singing quality evaluation. We compare the correlation between machine-predicted scores and manual annotations when using Mel-spectrogram, CQT and chromagram as network input respectively, to fine-tune the spectral features. The experimental results show that CRNN can learn more discriminatory information about singing quality from CQT compared to Mel-spectrogram and chromagram. We also incorporate the musically relevant pitch histogram representation in the CRNN to build a hybrid CRNN framework, which shows to improve the performance and song-independence of the neural network. In future work, other types of audio data representations can

be explored, since different acoustic features capture different information from the raw audio.

ACKNOWLEDGMENT

This research work is supported by Academic Research Council, Ministry of Education (ARC, MOE), Singapore. Grant: MOE2018-T2-2-127. Title: Learning Generative and Parameterized Interactive Sequence Models with RNNs.

REFERENCES

[1] C. Cao, M. Li, J. Liu, and Y. Yan, "A study on singing performance evaluation criteria for untrained singers," in *2008 9th International Conference on Signal Processing*. IEEE, 2008, pp. 1475–1478.

[2] C. Gupta, H. Li, and Y. Wang, "Perceptual evaluation of singing quality," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2017, pp. 577–586.

[3] T. Nakano, M. Goto, and Y. Hiraga, "An automatic singing skill evaluation method for unknown melodies using pitch interval accuracy and vibrato features," in *Ninth International Conference on Spoken Language Processing*, 2006.

[4] C. Gupta, H. Li, and Y. Wang, "Automatic evaluation of singing quality without a reference," in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2018, pp. 990–997.

[5] ——, "Automatic leaderboard: Evaluation of singing quality without a standard reference," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 13–26, 2019.

[6] K. A. Pati, S. Gururani, and A. Lerch, "Assessment of student music performances using deep neural networks," *Applied Sciences*, vol. 8, no. 4, p. 507, 2018.

[7] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[8] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold *et al.*, "Cnn architectures for large-scale audio classification," in *2017 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE, 2017, pp. 131–135.

[9] N. Takahashi and Y. Mitsufuji, "Multi-scale multi-band densenets for audio source separation," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017, pp. 21–25.

[10] S. Sigtia, E. Benetos, and S. Dixon, "An end-to-end neural network for polyphonic piano music transcription," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 5, pp. 927–939, 2016.

[11] K. Choi, G. Fazekas, M. Sandler, and K. Cho, "Convolutional recurrent neural networks for music classification," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2392–2396.

[12] M. Malik, S. Adavanne, K. Drossos, T. Virtanen, D. Ticha, and R. Jarina, "Stacked convolutional and recurrent neural networks for music emotion recognition," *arXiv preprint arXiv:1706.02292*, 2017.

[13] N. Zhang, T. Jiang, F. Deng, and Y. Li, "Automatic singing evaluation without reference melody using bi-dense neural network," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 466–470.

[14] C.-i. Wang and G. Tzanetakis, "Singing style investigation by residual siamese convolutional neural networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 116–120.

[15] B. C. Moore, *An introduction to the psychology of hearing*. Brill, 2012.

[16] S. Dieleman and B. Schrauwen, "Multiscale approaches to music audio feature learning," in *14th International Society for Music Information Retrieval Conference (ISMIR-2013)*. Pontifícia Universidade Católica do Paraná, 2013, pp. 116–121.

[17] J. Schlüter and S. Böck, "Improved musical onset detection with convolutional neural networks," in *2014 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE, 2014, pp. 6979–6983.

[18] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 2011, pp. 315–323.

[19] C. Schörkhuber and A. Klapuri, "Constant-q transform toolbox for music processing," in *7th Sound and Music Computing Conference, Barcelona, Spain*, 2010, pp. 3–64.

[20] M. Todisco, H. Delgado, and N. W. Evans, "A new feature for automatic speaker verification anti-spoofing: Constant q cepstral coefficients." in *Odyssey*, vol. 2016, 2016, pp. 283–290.

[21] T. Fujishima, "Real-time chord recognition of musical sound: A system using common lisp music," *Proc. ICMC, Oct. 1999*, pp. 464–467, 1999.

[22] G. H. Wakefield, "Mathematical representation of joint time-chroma distributions," in *Advanced Signal Processing Algorithms, Architectures, and Implementations IX*, vol. 3807. International Society for Optics and Photonics, 1999, pp. 637–645.

[23] G. K. Birajdar and M. D. Patil, "Speech/music classification using visual and spectral chromagram features," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 1, pp. 329–347, 2020.

[24] G. Tzanetakis, A. Ermolinskyi, and P. Cook, "Pitch histograms in audio and symbolic music information retrieval," *Journal of New Music Research*, vol. 32, no. 2, pp. 143–152, 2003.

[25] G. K. Koduri, J. Serrà Julià, and X. Serra, "Characterization of intonation in carnatic music by parametrizing pitch histograms," in *Gouyon F, Herrera P, Martins LG, Müller M. ISMIR 2012: Proceedings of the 13th International Society for Music Information Retrieval Conference; 2012 Oct 8-12; Porto, Portugal. Porto: FEUP Ediçoes, 2012.* International Society for Music Information Retrieval (ISMIR), 2012.

[26] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[27] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *International conference on machine learning*, 2015, pp. 2342–2350.

[28] J. J. Louviere, T. N. Flynn, and A. A. J. Marley, *Best-worst scaling: Theory, methods and applications*. Cambridge University Press, 2015.

[29] A. Marley, T. N. Flynn, and V. Australia, "Best worst scaling: theory and practice," *International encyclopedia of the social & behavioral sciences*, vol. 2, no. 2, pp. 548–552, 2015.

[30] P. Boersma, "Praat, a system for doing phonetics by computer," *Glot. Int.*, vol. 5, no. 9, pp. 341–345, 2001.

[31] L. Rabiner, "On the use of autocorrelation analysis for pitch detection," *IEEE transactions on acoustics, speech, and signal processing*, vol. 25, no. 1, pp. 24–33, 1977.

[32] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[33] K. Choi, G. Fazekas, K. Cho, and M. Sandler, "A tutorial on deep learning for music information retrieval," *arXiv preprint arXiv:1709.04396*, 2017.