Interspeech 2018
2-6 September 2018, Hyderabad

Singing voice phoneme segmentation by hierarchically inferring syllable and phoneme onset positions

Rong Gong, Xavier Serra

Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain

rong.gong@upf.edu, xavier.serra@upf.edu

Abstract

In this paper, we tackle the singing voice phoneme segmentation problem in the singing training scenario by using language-independent information – onset and prior coarse duration. We propose a two-step method. In the first step, we jointly calculate the syllable and phoneme onset detection functions (ODFs) using a convolutional neural network (CNN). In the second step, the syllable and phoneme boundaries and labels are inferred hierarchically by using a duration-informed hidden Markov model (HMM). To achieve the inference, we incorporate the *a priori* duration model as the transition probabilities and the ODFs as the emission probabilities into the HMM. The proposed method is designed in a language-independent way such that no phoneme class labels are used. For the model training and algorithm evaluation, we collect a new jingju (also known as Beijing or Peking opera) solo singing voice dataset and manually annotate the boundaries and labels at phrase, syllable and phoneme levels. The dataset is publicly available. The proposed method is compared with a baseline method based on hidden semi-Markov model (HSMM) forced alignment. The evaluation results show that the proposed method outperforms the baseline by a large margin regarding both segmentation and onset detection tasks.

Index Terms: singing voice phoneme segmentation, onset detection, convolutional neural network, multi-task learning, duration-informed hidden Markov model

1. Introduction

1.1. Background

The objective of this work lies in the background of automatic singing voice pronunciation assessment (ASVPA) for jingju music. Jingju singing is sung in the Mandarin language varied by two Chinese dialects. In a professional jingju singing training situation, the teacher would be very demanding of the students regarding a precise pronunciation of each syllable and phoneme.

We design an ASVPA system according to the *learning by imitation* method which is used as the basic training method by many musical traditions and as well by jingju singing [1]. In practice, this method contains three steps in regards to teaching how to sing a musical phrase: (i) the teacher firstly gives a demonstrative singing, (ii) Then the student is asked to imitate it. (iii) The teacher provides the feedback by assessing the student's singing at the syllable or phoneme-levels. Steps (ii) and (iii) should be repeated until the teacher satisfies with the student's singing. We design a three-steps ASVPA system in consideration of the above training method: (a) the teacher's demonstrative and student's imitative singing voice audios are recorded. The former is manually pre-segmented and labeled at the phrase, syllable and phoneme-levels. The latter is manually pre-segmented and labeled only at phrase-level. (b) The student's singing voice is then automatically segmented and labeled at the phoneme-level using the proposed method in this paper. (c) The corresponding phonemes between teacher's and student's recordings are finally compared by a phonetic pronunciation similarity algorithm. The similarity score will be given to the student as her/his pronunciation score.

In this paper, we approach step (a) manually so that the coarse durations and labels of the teacher's demonstrative singing are available as the prior information for step (b). We tackle the phoneme segmentation problem of step (b). Step (c) remains a work in progress.

1.2. Related work

The first topic which is highly related to our research is speech forced alignment. Speech forced alignment is a process that the orthographic transcription is aligned with the speech audio at word or phone-level. Most of the non-commercial alignment tools are built on HTK [2] or Kaldi [3] frameworks, such as Montreal forced aligner [4] and Penn Forced Aligner [5]. These tools implement a part of the automatic speech recognition (ASR) pipeline, train the HMM acoustic models iteratively using Viterbi algorithm and align audio features (e.g. MFCCs) to the HMM states. Brognaux and Drugman [6] explored the forced alignment on a small dataset using supplementary acoustic features and initializing the silence model by voice activity detection algorithm. To predict the confidence measure of the aligned word boundaries and to fine-tune their time positions, Serriére et al. [7] explored an alignment post-processing method using a deep neural network (DNN). The forced alignment is language-dependent, in which the acoustic models should be trained by using the corpus of a certain language. Another category of speech segmentation methods is language-independent, which relies on detecting the phoneme boundary change in the temporal-spectral domain [8, 9]. The drawback of these methods is that the segmentation accuracies are poorer than the language-dependent counterparts [10].

The second topic related to our research is the singing voice lyrics-to-audio alignment. Most of these works [11, 12, 13, 14, 15, 16, 17, 18] used the forced alignment method accompanied by music-related techniques. Loscos et al. [12] used MFCCs with additional features and also explored specific HMM topologies. Fujihara et al. [13] used voice/accompaniment separation to deal with mixed recording, and vocal detection, fricative detection to increase the alignment performance. Additional musical side information extracted from the musical score is used in many works. Mauch et al. [14] used chord information such that each HMM state contains both chord and phoneme labels. Iskandar et al. [15] constrained the alignment by using musical note length distribution. Gong et al. [16], Kruspe [17], Dzhambazov and Serra [18] all used sylla-

10.21437/Interspeech.2018-1224

ble/phoneme duration extracted from the musical score and decoded the alignment path by duration-explicit HMM models. Chien et al. [19] introduced an approach based on vowel likelihood models. Chang and Lee [20] used canonical time warping and repetitive vowel patterns to find the alignment for vowel sequence. Some other works achieved the alignment at music structure-level [21] or line-level [22].

Our research is also related to multi-task learning (MTL) because we want to achieve the segmentation on the jointly learned syllable and phoneme ODFs. MTL means that learning by optimizing more than one loss function [23, 24]. Hard parameter sharing is the most commonly used MTL approach, which applied by sharing the hidden layers between all tasks and keep several task-specific output layers [23]. Baxter argued that hard parameter sharing can reduce the risk of overfitting in an order of the number of tasks. In the music information retrieval (MIR) domain, Yang et al. proposed an MTL framework based on the neural networks to jointly consider chord and root note recognition problems [25]. Vogl et al. showed that learning beats jointly with drums can be beneficial for the task of drum detection [26].

1.3. Contribution

In this paper, we present a new jingju solo singing voice dataset for the phoneme segmentation, which is manually annotated at three hierarchical levels - phrase, syllable, phoneme (section 2). We propose a language-independent phoneme segmentation method which jointly learns syllable and phoneme onsets and hierarchically infers the phoneme boundaries and labels by a duration-informed HMM (section 3). Finally, we build a forced alignment baseline method based on HSMM and compare it with the proposed method (section 4).

2. Dataset

The jingju solo singing voice dataset focuses on two most important jingju role-types (performing profile) [27]: *dan* (female) and *laosheng* (old man). It has been collected by the researchers in Centre for Digital Music, Queen Mary University of London [28] and Music Technology Group, Universitat Pompeu Fabra.

Table 1: *Statistics of the dataset*

	#Recordings	#Phrases	#Syllables	#Phonemes
Train	56	214	1965	5017
Test	39	216	1758	4651

The dataset contains 95 recordings split into train and test sets (table 1). The recordings in the test set are student imitative singing. Their teacher demonstrative recordings can be found in the train set, which guarantees that the coarse syllable/phoneme duration and labels are available for the algorithm testing. Audios are pre-segmented into singing phrases. The syllable/phoneme ground truth boundaries (onsets/offsets) and labels are manually annotated in Praat [29] by two Mandarin native speakers and a jingju musicologist. 29 phoneme categories are annotated, which include a silence category and a non-identifiable phoneme category, e.g. throat-cleaning. The category table can be found in the Github page[1]. The dataset is publicly available[2].

[1] https://goo.gl/fFr9XU

[2] https://doi.org/10.5281/zenodo.1185123

3. Proposed method

We introduce a coarse duration-informed phoneme segmentation method. The syllable and phoneme onset ODFs are jointly learned by a hard parameter sharing multi-task CNN model. The syllable/phoneme boundaries and labels are then inferred by an HMM using the *a priori* duration model as the transition probabilities and the ODFs as the emission probabilities.

3.1. CNN onset detection function

Audio preprocessing: We use MADMOM[3] Python package to calculate the log-mel spectrogram of the student's singing audio. The frame size and hop size of the spectrogram are respectively 46.4ms (2048 samples) and 10ms (441 samples). The low and high frequency bounds are 27.5Hz and 16kHz. We use a log-mel context as the CNN model input, where the context size is 80×15 (log-mel binsframes). Thus the CNN model takes a binary onset/non-onset decision sequentially for every frame given its context: $\pm$70ms, 15 frames in total.

Preparing target labels: The target labels of the training set are prepared according to the ground truth annotations. We set the label of a certain context to 1 if an onset has been annotated for its corresponding frame, otherwise 0. To compensate the human annotation inaccuracy and to augment the positive sample size, we also set the labels of the two neighbor contexts to 1. However, the importance of the neighbor contexts should not be equal to their center context, thus we compensate this by setting the sample weights of the neighbor contexts to 0.25. A similar sample weighting strategy has been presented in Schluter's paper [30]. Finally, for each log-mel context, we have its syllable and phoneme labels. They will be used as the training targets in the CNN model to predict the onset presence.

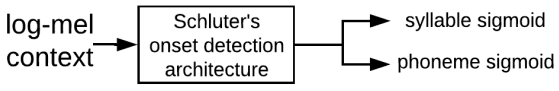

Figure 1: *Diagram of the multi-task CNN model.*

Hard parameter sharing multi-task CNN model: We build a CNN for classifying each log-mel context and output the syllable and phoneme ODFs. We extend the CNN architecture presented in Schluter's work [30] by using two predicting objectives – syllable and phoneme (figure 1). The two objectives share the same parameters, and both are using the sigmoid activation function. Binary cross-entropy is used as the loss function. The loss weighting coefficients for the two objectives are set to equal since no significant effect has been found in the preliminary experiment. The model parameters are learned with mini-batch training (batch size 256), adam [31] update rule and early stopping – if validation loss is not decreasing after 15 epochs. The ODFs output from the CNN model is used as the emission probabilities for the syllable/phoneme boundary inference.

3.2. Phoneme boundaries and labels inference

The inference algorithm receives the syllable and phoneme durations and labels of teacher's singing phrase as the prior input

[3] https://github.com/CPJKU/madmom

and infers the syllable and phoneme boundaries and labels for the student's singing phrase.

3.2.1. Coarse duration and a priori duration model

The syllable durations of the teacher's singing phrase are stored in an array $M^s = \mu^1 \cdots \mu^n \cdots \mu^N$, where μ^n is the duration of the nth syllable. The phoneme durations are stored in a nested array $M_p = M_p^1 \cdots M_p^n \cdots M_p^N$, where M_p^n is the sub-array with respect to the nth syllable and can be further expanded to $M_p^n = \mu_1^n \cdots \mu_k^n \cdots \mu_{K_n}^n$, where K_n is the number of phonemes contained in the nth syllable. The phoneme durations of the nth syllable sum to its syllable duration: $\mu^n = \sum_{k=1}^{K_n} \mu_k^n$ (figure 2). In both syllable and phoneme duration sequences – M^s, M_p, the duration of the silence is not treated separately and is merged with its previous syllable or phoneme.

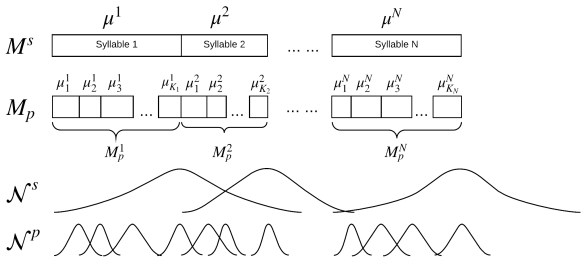

Figure 2: *Illustration of the syllable M^s and phoneme M_p coarse duration sequences and their a priori duration models – $\mathcal{N}^s$, $\mathcal{N}^p$. The blank rectangulars in M_p represent the phonemes.*

The *a priori* duration model is shaped with a Gaussian function $\mathcal{N}(d; \mu_n, \sigma_n^2)$. It provides the prior likelihood of an onset to occur according to the syllable/phoneme duration of the teacher's singing. The mean μ_n of the Gaussian represents the expected duration of nth teacher's syllable/phoneme. Its standard deviation σ_n is proportional to μ_n: $\sigma_n = \gamma\mu_n$ and γ is heuristically set to 0.35. Figure 2 provides an intuitive example of how the *a priori* duration model works. The a priori duration model will be incorporated into a duration-informed HMM as the state transition probabilities to inform that where syllable/phoneme onsets is likely to occur in student's singing phrase.

3.2.2. Duration-informed HMM for segment boundary and label inference

We present an HMM configuration which makes use of the coarse duration and label input (section 3.2.1) and can be applied to inferring firstly (i) the syllable boundaries and labels on the ODF for the whole singing phrase, then (ii) the phoneme boundaries and labels on the ODF segment constrained by the inferred syllable boundaries. To use the same inference formulation, we unify the notations N, K_n (both introduced in section 3.2.1) to N, and M^s, M_p^n to M. The unification of the notations has a practical meaning because we use the same algorithm for both syllable and phoneme inference. The HMM is characterized by the following:

1. The hidden state space is a set of T candidate onset positions $S_1, S_2, \cdots, S_T$ discretized by the hop size, where S_T is the offset position of the last syllable or the last phoneme within a syllable.
2. The state transition probability at the time instant t associated with state changes is defined by *a priori* duration distribution $\mathcal{N}(d_{ij}; \mu_t, \sigma_t^2)$, where d_{ij} is the time distance between states S_i and S_j ($j > i$). The length of the inferred state sequence is equal to N.
3. The emission probability for the state S_j is represented by its value in the ODF, which is denoted as p_j.

The goal is to find the best onset state sequence $Q = q_1 q_2 \cdots q_{N-1}$ for a given duration sequence M and impose the corresponding segment label, where q_i denotes the onset of the $i+1$th inferred syllable/phoneme. The onset of the current segment is assigned as the offset of the previous segment. q_0 and q_N are fixed as S_1 and S_T as we expect that the onset of the first syllable(or phoneme) is located at the beginning of the singing phrase(or syllable) and the offset of the last syllable(or phoneme) is located at the end of the phrase(or syllable). One can fulfill this assumption by truncating the silences at both ends of the incoming audio. The best onset sequence can be inferred by the logarithmic form of Viterbi algorithm [32]:

Algorithm 1 Logarithmic form of Viterbi algorithm using the *a priori* duration model

$\delta_n(i) \leftarrow \max_{q_1, q_2, \cdots, q_n} \log P[q_1 q_2 \cdots q_n,\ \mu_1 \mu_2 \cdots \mu_n]$
procedure LOGFORMVITERBI(M, p)
Initialization:
$\delta_1(i) \leftarrow \log(\mathcal{N}(d_{1i}; \mu_1, \sigma_1^2)) + \log(p_i)$
$\psi_1(i) \leftarrow S_1$
Recursion:
$tmp\_var(i,j) \leftarrow \delta_{n-1}(i) + \log(\mathcal{N}(d_{ij}; \mu_n, \sigma_n^2))$
$\delta_n(j) \leftarrow \max_{1 \leqslant i < j} tmp\_var(i,j) + \log(p_j)$
$\psi_n(j) \leftarrow \arg\max_{1 \leqslant i < j} tmp\_var(i,j)$
Termination:
$q_N \leftarrow \arg\max_{1 \leqslant i < T} \delta_{N-1}(i) + \log(\mathcal{N}(d_{iT}; \mu_N, \sigma_N^2))$

Finally, the state sequence Q is obtained by the backtracking step. The implementation of the algorithm can be found in the Github link[1].

4. Evaluation

The phoneme segmentation task consists of determining the time positions of phoneme onsets and offsets and its labels. As the onset of the current phoneme is assigned as the offset of the previous phoneme, the evaluation consists in comparing the detected (i) onsets and (ii) segments to their reference ones. As the syllable segmentation is the prerequisite for the proposed method, we also report the syllable segmentation results. To compare with the proposed method, we firstly introduce a forced alignment-based baseline method.

4.1. Baseline method

The baseline is a 1-state monophone DNN/HSMM model. We use monophone model because our small dataset doesn't have enough phoneme instances for exploring the context-dependent triphones model, also Brognaux and Drugman [6] and Pakoci et al. [10] argued that context-dependent model can't bring significant alignment improvement. It is convenient to apply 1-state model because each phoneme can be represented by a semi-Markovian state carrying a state occupancy time distribution. The audio preprocessing step is the same as in section 3.1.

We construct an HSMM for phoneme segment inference. The topology is a left-to-right semi-Markov chain, where the states represent sequentially the phonemes of the teacher's singing phrase. As we are dealing with the forced alignment, we constraint that the inference can only be started by the leftmost state and terminated to the rightmost state. The self-transition probabilities are set to 0 because the state occupancy depends on the predefined distribution. Other transitions – from current states to subsequent states are set to 1. We use a one-layer CNN with multi-filter shapes as the acoustic model [33] and the Gaussian $\mathcal{N}(d; \mu_n, \sigma_n^2)$ introduced in section 3.2.1 as the state occupancy distribution. The inference goal is to find best state sequence, and we use Guédon's HSMM Viterbi algorithm [34] for this purpose. The baseline details and code can be found in the Github page[1]. Finally, the segments are labeled by the alignment path, and the phoneme onsets are taken on the state transition time positions.

4.2. Metrics and results

To define a correctly detected onset, we choose a tolerance of $\tau = 25$ms. If the detected onset o_d lies within the tolerance of its ground truth counterpart o_g: $|o_d - o_g| < \tau$, we consider that it's correctly detected. To measure the segmentation correctness, we use the ratio between the duration of correctly labeled segments and the total duration of the singing phrase. This metric has been suggested by Fujihara et al. [13] in their lyrics alignment work. We trained both proposed and baseline models 5 times with different random seeds, and report the mean and the standard deviation score on the test set. Below is the evaluation results table:

Table 2: *Evaluation results table. Table cell: mean score±standard deviation score.*

	Onset F1-measure %		Segmentation %	
	phoneme	syllable	phoneme	syllable
Proposed	75.2±0.6	75.8±0.4	60.7±0.4	84.6±0.3
Baseline	44.5±0.9	41.0±1.0	53.4±0.9	65.8±0.7

We only show the F1-measure of the onset detection results in table 2. The full results can be found in the Github page[1].

4.3. Discussions

On both metrics – onset detection and segmentation, the proposed method outperforms the baseline. The proposed method uses the ODF which provides the time "anchors" for the onset detection. Besides, the ODF calculation is a binary classification task. Thus the training data for both positive and negative class is more than abundant. Whereas, the phonetic classification is a harder task because many singing interpretations of different phonemes have the similar temporal-spectral patterns. Our relatively small training dataset might be not sufficient to train a proper discriminative acoustic model with 29 phoneme categories. We believe that these reasons lead to a better onset detection and segmentation performance of the proposed method.

Fig 3 shows an result example for a testing singing phrase. The syllable/phoneme labels, baseline emission probabilities matrix and alignment path are omitted for the plot clarity, and can be found in the link[1]. Notice that there are some extra or missing onsets in the detection. This is due to the inconsistency

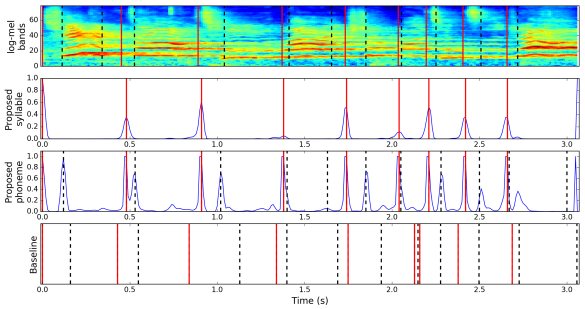

Figure 3: *An illustration of the result for a testing singing phrase. The red solid and black dash vertical lines are respectively the syllable and phoneme onset positions (1st row: ground truth, 2nd and 3rd rows: proposed method, 4th row: baseline method). The blue curves in the 2nd and 3rd row are respectively the syllable and phoneme ODFs.*

between the coarse duration input and the ground truth – students might add or delete some phonemes in the actual singing. Also notice that in the 3rd row, the two detected phoneme onsets within the last syllable are not in the peak positions of the ODF. This is due to that the onsets is inferred by taking into account both ODF and the *a priori* duration model, and the latter partially constraints the detected onsets.

The biggest advantage of the proposed method is the language-independency, which means that the pre-trained CNN model can be eventually applied to the singing voice of various languages because they could share the similar temporal-spectral patterns of phoneme transitions. Besides, the Viterbi decoding of the proposed method (time complexity $O(TS^2)$, T: time, S: states) is much faster than the HSMM counterpart (time complexity $O(TS^2 + T^2S)$). An interactive jupyter notebook demo for showcasing the proposed algorithm is provided for running in Google Colab[4].

5. Conclusions

In this paper, we presented a language-independent singing voice segmentation method by first jointly learning the syllable and phoneme ODFs using a CNN model, then inferring the onsets and segmented labels using a duration-informed HMM. We also presented a jingju solo singing voice dataset with manual boundary and label annotations. For the evaluation, we compared the proposed method with a baseline forced alignment method based on a language-dependent HSMM. The evaluation results showed that the proposed method outperforms the baseline in both segmentation and onset detection tasks. We attribute the performance improvement of the proposed method to the efficient use of the onset and duration information for our relatively small dataset. However, the proposed method is not able to solve the phoneme insertion or deletion problems when there is a mismatch between the prior coarse duration information and the actual singing. We are improving the algorithm to overcome this limitation by using recognition-based methods.

6. Acknowledgements

This work is supported by the CompMusic project (ERC grant agreement 267583).

[4] https://goo.gl/BzajRy

7. References

[1] R. Gong and X. Serra, "Identification of potential music information retrieval technologies for computer-aided jingju singing training," in *5th China conference on sound and music technology*, Suzhou, China, Oct 2017.

[2] S. J. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The HTK Book Version 3.4*. Cambridge University Press, 2006.

[3] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011.

[4] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal forced aligner: Trainable text-speech alignment using kaldi," in *Proceedings of Interspeech 2017*, Stockholm, Sweden, 2017, pp. 498–502. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2017-1386

[5] "Penn phonetics lab forced aligner," https://web.sas.upenn.edu/phonetics-lab/facilities/, accessed: 2018-03-08.

[6] S. Brognaux and T. Drugman, "HMM-based speech segmentation: Improvements of fully automatic approaches," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 24, no. 1, pp. 5–15, 2016.

[7] G. Serrière, C. Cerisara, D. Fohr, and O. Mella, "Weakly-supervised text-to-speech alignment confidence measure," in *International Conference on Computational Linguistics (COLING)*, Osaka, Japan, 2016.

[8] A. Esposito and G. Aversano, "Text independent methods for speech segmentation," in *Nonlinear Speech Modeling and Applications*. Springer, 2005, pp. 261–290.

[9] G. Almpanidis, M. Kotti, and C. Kotropoulos, "Robust detection of phone boundaries using model selection criteria with few observations," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 2, pp. 287–298, Feb 2009.

[10] E. Pakoci, B. Popović, N. Jakovljević, D. Pekar, and F. Yassa, "A phonetic segmentation procedure based on hidden markov models," in *International Conference on Speech and Computer*. Budapest, Hungary: Springer, 2016, pp. 67–74.

[11] A. Mesaros and T. Virtanen, "Automatic alignment of music audio and lyrics," in *Proceedings of the 11th Int. Conference on Digital Audio Effects (DAFx-08)*, Espoo, Finland, 2008.

[12] A. Loscos, P. Cano, and J. Bonada, "Low-delay singing voice alignment to text," in *International Computer Music Conference*, Beijing, China, 1999.

[13] H. Fujihara, M. Goto, J. Ogata, and H. G. Okuno, "Lyricsynchronizer: Automatic synchronization system between musical audio signals and lyrics," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1252–1261, 2011.

[14] M. Mauch, H. Fujihara, and M. Goto, "Integrating additional chord information into hmm-based lyrics-to-audio alignment," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 200–210, 2012.

[15] D. Iskandar, Y. Wang, M.-Y. Kan, and H. Li, "Syllabic level automatic synchronization of music signals and text lyrics," in *Proceedings of the 14th ACM international conference on Multimedia*, Santa Barbara, CA, USA, 2006, pp. 659–662.

[16] R. Gong, P. Cuvillier, N. Obin, and A. Cont, "Real-time audio-to-score alignment of singing voice based on melody and lyric information," in *Proceedings of Interspeech 2015*, Dresden, Germany, 2015.

[17] A. M. Kruspe, "Keyword spotting in singing with duration-modeled hmms," in *23rd European Signal Processing Conference (EUSIPCO)*, Nice, France, Aug 2015, pp. 1291–1295.

[18] G. B. Dzhambazov and X. Serra, "Modeling of phoneme durations for alignment between polyphonic audio and lyrics," in *12th Sound and Music Computing Conference*, Maynooth, Ireland, 2015.

[19] Y. R. Chien, H. M. Wang, and S. K. Jeng, "Alignment of lyrics with accompanied singing audio based on acoustic-phonetic vowel likelihood modeling," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 1998–2008, Nov 2016.

[20] S. Chang and K. Lee, "Lyrics-to-audio alignment by unsupervised discovery of repetitive patterns in vowel acoustics," *IEEE Access*, vol. 5, pp. 16 635–16 648, 2017.

[21] M. Müller, F. Kurth, D. Damm, C. Fremerey, and M. Clausen, "Lyrics-based audio retrieval and multimodal navigation in music collections," in *International Conference on Theory and Practice of Digital Libraries*. Budapest, Hungary: Springer, 2007, pp. 112–123.

[22] Y. Wang, M.-Y. Kan, T. L. Nwe, A. Shenoy, and J. Yin, "Lyrically: automatic synchronization of acoustic musical signals and textual lyrics," in *Proceedings of the 12th annual ACM international conference on Multimedia*. New York, NY, USA: ACM, 2004, pp. 212–219.

[23] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv preprint arXiv:1706.05098*, 2017.

[24] R. Caruana, "Multitask learning," in *Learning to learn*. Springer, 1998, pp. 95–133.

[25] M. H. Yang, L. Su, and Y. H. Yang, "Highlighting root notes in chord recognition using cepstral features and multi-task learning," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, Dec 2016, pp. 1–8.

[26] R. Vogl, M. Dorfer, G. Widmer, and P. Knees, "Drum transcription via joint beat and drum modeling using convolutional recurrent neural networks," in *18th International Society for Music Information Retrieval Conference*, Suzhou, China, 2017.

[27] R. C. Repetto and X. Serra, "Creating a corpus of jingju (Beijing opera) music and possibilities for melodic analysis," in *ISMIR*, Taipei, Taiwan, 2014.

[28] D. A. A. Black, M. Li, and M. Tian, "Automatic identification of emotional cues in Chinese opera singing," in *ICMPC*, Seoul, South Korea, Aug. 2014.

[29] P. Boersma, "Praat, a system for doing phonetics by computer." *Glot International*, vol. 5, no. 9/10, pp. 341–345, 2001.

[30] J. Schluter and S. Bock, "Improved musical onset detection with convolutional neural networks," in *Proceedings of Acoustics, speech and signal processing (ICASSP)*, Florence, Italy, 2014, pp. 6979–6983.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[32] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[33] J. Pons, O. Slizovskaia, R. Gong, E. Gómez, and X. Serra, "Timbre analysis of music audio signals with convolutional neural networks," in *25th European Signal Processing Conference (EUSIPCO)*, Kos, Greece, Aug 2017, pp. 2744–2748.

[34] Y. Guédon, "Exploring the state sequence space for hidden markov and semi-markov chains," *Computational Statistics & Data Analysis*, vol. 51, no. 5, pp. 2379–2409, 2007.