END-TO-END LYRICS TRANSCRIPTION INFORMED BY PITCH AND ONSET ESTIMATION

Tengyu Deng[1] **Eita Nakamura**[1] **Kazuyoshi Yoshii**[1,2]
[1] Graduate School of Informatics, Kyoto University, Japan
[2] PRESTO, Japan Science and Technology Agency (JST), Japan
deng@sap.ist.i.kyoto-u.ac.jp, {eita.nakamura, yoshii}@i.kyoto-u.ac.jp

ABSTRACT

This paper presents an automatic lyrics transcription (ALT) method for music recordings that leverages the framewise semitone-level sung pitches estimated in a multi-task learning framework. Compared to automatic speech recognition (ASR), ALT is challenging due to the insufficiency of training data and the variation and contamination of acoustic features caused by singing expressions and accompaniment sounds. The domain adaptation approach has thus recently been taken for updating an ASR model pre-trained from sufficient speech data. In the naive application of the end-to-end approach to ALT, the internal audio-to-lyrics alignment often fails due to the time-stretching nature of singing features. To stabilize the alignment, we make use of the semi-synchronous relationships between notes and characters. Specifically, a convolutional recurrent neural network (CRNN) is used for estimating the semitone-level pitches with note onset times while eliminating the intra- and inter-note pitch variations. This estimate helps an end-to-end ALT model based on connectionist temporal classification (CTC) learn correct audio-to-character alignment and mapping, where the ALT model is trained jointly with the pitch and onset estimation model. The experimental results show the usefulness of the pitch and onset information in ALT.

1. INTRODUCTION

Automatic lyrics transcription (ALT) refers to a task that aims to estimate the sung texts from music recordings, typically under the presence of accompaniment sounds. Since music composition and sharing have become very popular among non-professional people (e.g., YouTube), the number of non-annotated music data without lyrics transcriptions has been increasing rapidly. ALT has thus gained a lot of attention from the music information retrieval (MIR) community because of its usefulness in karaoke subtitle generation and text-based indexing.

Considering the similarity between ALT and automatic speech recognition (ASR), most studies on ALT have attempted to use ASR techniques, with some modifications if necessary. The hybrid approach based on a hidden Markov model (HMM) enhanced by a deep neural network (DNN), for example, has been used, where only the acoustic model was optimized for ALT [1]. Another way of ALT is to take the end-to-end approach that directly learns a sequence-to-sequence (audio-to-text) mapping. While the connectionist temporal classification (CTC) [2] and/or the attention mechanism [3] have widely been used for ASR [4, 5], the CTC has mainly been used for ALT [6, 7], in conjunction with the attention mechanism [8]. One reason is that the CTC considers only the monotonic audio-to-text alignment, which is relatively easier to infer from a limited amount of training data.

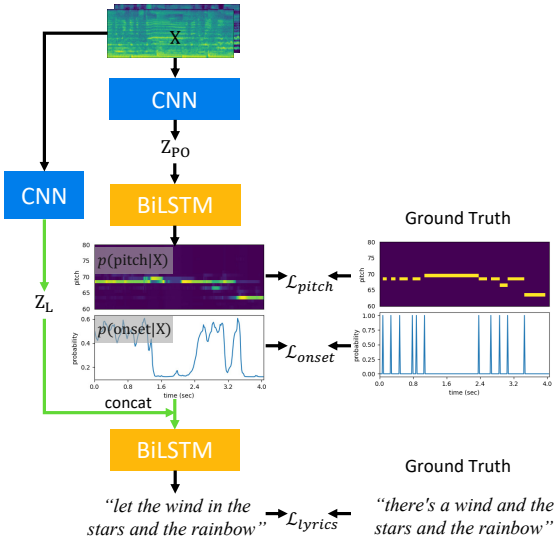

Figure 1. The proposed lyrics transcription method that estimates the pitches and onsets of singing voice and then uses them for character-level lyrics transcription.

The audio-to-text alignment plays a key role in end-to-end ALT and still remains a challenging problem. Firstly, the acoustic characteristics of singing voice vary over time in various ways according to the underlying sung notes and singing expressions. While the phones of speech tend to have particular durations and pitches specific to the speaker, those of singing voice may have time-stretched durations and semi-stepwisely time-varying pitches determined by the singer and the score. Secondly, the acoustic features of singing voice are contaminated by accompaniment sounds

or distorted by singing voice separation. It is, however, difficult to collect as training data a sufficient amount of music recordings with aligned lyrics annotations that cover a wide variety of recording conditions and singing styles.

To solve these problems, we make effective use of the framewise pitches and onset times of sung notes as acoustic features exclusive for ALT. In musical scores, lyrics are typically described synchronously with notes, i.e., words or characters tend to coincide with notes. This implies that the note onset information can be used for guiding an end-to-end ALT model to efficiently find the correct audio-to-lyrics alignment from a limited amount of training data. Multi-conditional training with the note pitch information is also expected to make the ALT model robust against the pitch variations.

In this paper, we propose an ALT method based on a cascading multi-task architecture that estimates the pitches and onset times of sung notes and then transcribes the lyrics at the character level in a pitch- and onset-conditioned manner (Fig. 1). More specifically, a convolutional recurrent neural network (CRNN) is used for jointly estimating the semitone-level pitches and onset times while eliminating the intra- and inter-note pitch variations (e.g., vibrato and glissando). Informed by this estimation, another CRNN is then used with CTC for learning audio-to-character alignment and mapping. These two CRNNs are trained jointly with backpropagation such that the sum of the pitch, onset, lyrics estimation losses is minimized. To mitigate the data insufficiency problem, we also use a domain adaptation method that fine-tunes a baseline ASR model pre-trained from huge speech data.

The main contribution of this paper lies in the first attempt for joint lyrics and pitch transcription towards comprehensive singing voice analysis. We experimentally show that the score information exclusive to music can be used effectively for finding audio-to-text alignment in end-to-end ALT with insufficient training data.

2. RELATED WORK

This section reviews related work on automatic lyrics transcription (ALT) and singing voice transcription (SVT).

2.1 Automatic Lyrics Transcription

ALT for music recordings under the presence of accompaniment sounds is still a difficult task due to the contamination of acoustic features [6–8]. A standard way of mitigating the adverse effect of accompaniment sounds is to take the two-step approach that performs singing voice separation [9] and lyrics transcription in this order. Although the remarkable improvement has been made in terms of the pure signal processing performance typically measured by the signal-to-distortion ratio (SDR) [10], the separated singing voice, however, is hard to transcribe, *i.e.*, the word error rate (WER) might be low, because an ALT model trained with clean isolated singing voice would suffer from the distortion of acoustic features and the mismatch between the training and test conditions. Although even ALT for clean singing voice [1] has much room for performance improvement due to the considerable variation of acoustic features caused by singing expressions, working directly on music recordings without singing voice separation could achieve better performance of ALT [7].

Inspired by the great success of the end-to-end approach to ASR, several attempts have been made for directly learning the mapping between non-aligned input and output sequences (audio and lyrics) of different lengths [6–8]. At the heart of the end-to-end learning is the audio-to-lyrics alignment based on the connectionist temporal classification (CTC) [2] and/or the attention mechanism [3]. The CTC-based approach estimates the posterior probabilities of labels (*e.g.*, words and characters) at the frame level and aims to maximize the total score obtained by efficiently accumulating the posterior probabilities of all possible *monotonic* alignment paths between the estimated and ground-truth label sequences. The attention-based approach is more powerful in that it can consider non-monotonic alignment and is thus useful for a wider variety of tasks (*e.g.*, machine translation). In general, however, the latter needs a larger amount of training data for finding the correct alignment and is thus hard to apply solely to ALT.

Some studies on audio-to-lyrics alignment have reported the effectiveness of using pitch information [11, 12]. Considering the semi-synchronous relationships between notes and characters, joint estimation of the pitches, onset times, and lyrics of singing voice would help an end-to-end model find the correct audio-to-lyrics alignment.

2.2 Singing Voice Transcription

The ultimate goal of SVT, a special case of automatic music transcription (AMT), is to estimate a human-readable vocal score underlying a given music recording. Towards this goal, a lot of efforts have been made for estimating the continuous fundamental frequencies (F0s) or discrete semitone-level pitches of singing voice at the frame or note level [13,14]. In particular, frame-level F0 estimation (*a.k.a.* melody extraction) has conventionally been considered as a subtask of SVT and intensively studied thanks to the great advance of supervised deep learning [15, 16]. There is, however, a big gap between melody extraction and genuine SVT because accurate scores are hard to obtain just by quantizing continuous F0 contours, typically at the tatum level, due to the large fluctuations and smooth transitions of F0s (*e.g.*, vibrato or glissando).

The latest study on SVT attempted to directly estimate a note sequence with discrete pitches and score positions [17]. This method is based on a hierarchical hidden semi-Markov model (HHSMM) that represents the generative process of observed singing spectra from a latent sequence of notes whose pitches and onset positions are assumed to follow a key-dependent Markov model and a metrical Markov model, respectively. The emission model was implemented with a CRNN that is pretrained to estimate the pitches of singing voice at the frame level from music spectra such that the intra- and inter-note F0 variations are eliminated. This technique forms the basis of the pitch and onset estimators used in our study.

3. PROPOSED METHOD

This section describes the proposed ALT method based on a cascading multi-task architecture.

3.1 Multi-task Learning Approach

Our method takes as input the mel-spectrogram of a music recording and that of the singing voice extracted from the recording with a DNN-based music separation method called Open-Unmix [18]. Since the singing voice separation is known to affect ALT, the original recording is thus used as well as the separated singing voice. Let $\mathbf{X} \in \mathbb{R}^{C \times F \times T}$ be the set of the input mel-spectrograms, where C is the number of channels ($C = 2$ in this paper), F is the number of frequency bins, and T is the number of frames.

We use as a basic building block a convolutional recurrent neural network (CRNN) consisting of a convolutional neural network (CNN) working as an encoder and a recurrent neural network (RNN) working as a decoder. The encoder extracts latent features from the input spectrograms and then the decoder estimates an output sequence while considering the sequential dependency of the latent features. The CRNN consists of residual CNN blocks with skip connections and RNN blocks (Fig. 2). Each CNN has a rectified linear unit (ReLU) and implemented with instance normalization. In contrast, each RNN block is a bidirectional LSTM (BiLSTM) [19] with layer normalization.

As shown in Fig. 1, our method uses two CRNNs. One CRNN is used for jointly estimating the posterior probabilities of the semitone-level pitches and those of the onset presence at the frame level. Given the estimated pitch and onset probabilities, the other CRNN is used for transcribing the lyrics from the spectrograms. Specifically, the estimated pitch and onset probabilities are fed together with the latent features extracted from the CNN into the RNN. Both CRNNs are trained jointly such the sum of the framewise pitch and onset estimation losses and the CTC-based lyrics transcription loss is minimized.

3.1.1 Pitch and Onset Estimation

The goal of pitch and onset estimation (supplementary task) is to estimate the framewise pitch and onset probabilities, denoted by $p(\text{pitch}|\mathbf{X}) \in \mathbb{R}^{K \times T}$ and $p(\text{onset}|\mathbf{X}) \in \mathbb{R}^{1 \times T}$, respectively, from the input data $\mathbf{X} \in \mathbb{R}^{C \times F \times T}$, where K is the number of the semitone-level pitches corresponding to MIDI note numbers plus no-pitch ($K = 128 + 1$).

Specifically, $\mathbf{X}$ is first fed to the series of multiple residual CNN blocks as follows:

$$\mathbf{Z}_{\text{PO}} = \text{CNN}(\mathbf{X}), \tag{1}$$

where $\mathbf{Z}_{\text{PO}} \in \mathbb{R}^{C' \times F \times T}$ is the output of the CNN and C' is the number of channels. Note that the zero padding is performed so that the output of each residual CNN block retains the shape of $\mathbf{X}$ except for the number of channels. Then $\mathbf{Z}_{\text{PO}}$ is reshaped into $\mathbf{Z}'_{\text{PO}} \in \mathbb{R}^{C'F \times T}$ and fed to the RNN as follows:

$$\mathbf{Y}_{\text{PO}} = \text{RNN}(\mathbf{Z}'_{\text{PO}}), \tag{2}$$

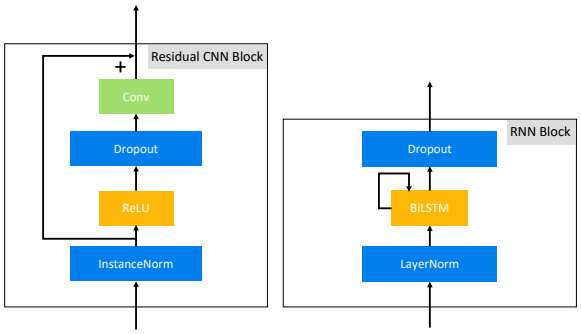

Figure 2. The residual CNN block and the RNN block.

where $\mathbf{Y}_{\text{PO}} \in \mathbb{R}^{C'' \times T}$ is the output of the RNN and C'' is the number of channels. Finally, $p(\text{pitch}|\mathbf{X})$ and $p(\text{onset}|\mathbf{X})$ are computed by feeding $\mathbf{Y}_{\text{PO}}$ to a fully-connected (FC) layer and the softmax and sigmoid functions, respectively, as follows:

$$[\mathbf{Y}_{\text{pitch}}, \mathbf{Y}_{\text{onset}}] = \text{FC}(\mathbf{Y}_{\text{PO}}), \tag{3}$$

$$p(\text{pitch}|\mathbf{X}) = \text{softmax}(\mathbf{Y}_{\text{pitch}}), \tag{4}$$

$$p(\text{onset}|\mathbf{X}) = \text{sigmoid}(\mathbf{Y}_{\text{onset}}), \tag{5}$$

where $\mathbf{Y}_{\text{pitch}} \in \mathbb{R}^{K \times T}$ and $\mathbf{Y}_{\text{onset}} \in \mathbb{R}^{1 \times T}$ are the intermediate outputs from the FC layer.

3.1.2 Lyrics Transcription

The goal of lyrics transcription is to estimate the framewise character probabilities, denoted by $p(\text{character}|\mathbf{X}) \in \mathbb{R}^{V \times L}$ from the input data $\mathbf{X} \in \mathbb{R}^{C \times F \times T}$, where V is the number of characters (dictionary size) including a special blank label used for CTC ($V = 33 + 1$ in this paper).

Specifically, in the same way as pitch and onset estimation, $\mathbf{X}$ is first fed to the series of multiple residual CNN blocks as follows:

$$\mathbf{Z}_{\text{L}} = \text{CNN}(\mathbf{X}), \tag{6}$$

where $\mathbf{Z}_{\text{L}} \in \mathbb{R}^{C' \times F \times T}$ is the output of the CNN. Then $\mathbf{Z}_{\text{L}}$ is reshaped into $\mathbf{Z}'_{\text{L}} \in \mathbb{R}^{C'F \times T}$, stacked with the estimated pitch probabilities $p(\text{pitch}|\mathbf{X}) \in \mathbb{R}^{K \times T}$ and onset probabilities $p(\text{onset}|\mathbf{X}) \in \mathbb{R}^{1 \times T}$, and fed to the RNN as follows:

$$\mathbf{Y}_{\text{L}} = \text{RNN}([\mathbf{Z}'_{\text{L}}, p(\text{pitch}|\mathbf{X}), p(\text{onset}|\mathbf{X})]), \tag{7}$$

where $\mathbf{Y}_{\text{L}} \in \mathbb{R}^{C'' \times L}$ is the output of the RNN and C'' is the number of channels. For computational simplicity, $\mathbf{Z}_{\text{L}}$ as well as $p(\text{pitch}|\mathbf{X})$ and $p(\text{onset}|\mathbf{X})$ are downsampled to $T/2$ frames with a 2-dimensional max-pooling layer. Finally, $p(\text{character}|\mathbf{X})$ is computed by feeding $\mathbf{Y}_{\text{L}}$ to a fully-connected (FC) layer and the softmax function as follows:

$$\mathbf{Y}_{\text{character}} = \text{FC}(\mathbf{Y}_{\text{L}}), \tag{8}$$

$$p(\text{character}|\mathbf{X}) = \text{softmax}(\mathbf{Y}_{\text{character}}), \tag{9}$$

where $\mathbf{Y}_{\text{character}} \in \mathbb{R}^{V \times T}$ is the intermediate output from the FC layer.

3.1.3 Joint Training with Domain Adaptation

The CRNN used for pitch and onset estimation and that for lyrics transcription are mutually dependent and thus trained jointly such that the sum of the framewise pitch and onset estimation losses (cross-entropies) and the lyrics transcription loss (CTC loss) is minimized. The CTC loss is computed from the framewise estimate $p(\text{character}|\mathbf{X})$ by efficiently accumulating the costs of all possible character sequences that can be reduced to the ground-truth character sequence by removing the blank labels. The pitch and onset probabilities $p(\text{pitch}|\mathbf{X})$ and $p(\text{onset}|\mathbf{X})$ and the underlying boundary information are considered to make $p(\text{character}|\mathbf{X})$ consistent with the ground-truth character sequence.

To mitigate the insufficiency of music data with lyrics annotations (DALI dataset [20]), we take the domain adaptation approach based on transfer learning [21]. Specifically, the CRNN used for lyrics transcription is trained using sufficient speech data (LibriSpeech corpus [22]) and then fine-tuned using both speech and music data.

3.2 Decoding

At run-time, we aim to estimate a series of note events with semitone-level pitches (MIDI note numbers) and onset times (frames) and transcribe the lyrics (Fig. 3). Specifically, the pitches with the maximum probabilities are taken from the estimated pitch probabilities $p(\text{pitch}|\mathbf{X})$. The onset times are determined with a peak picking strategy [23] with a window of 50 [ms] and a threshold of 0.4. A frame is counted as the onset time if the onset probability at this frame is maximal within 25 [ms] around this frame, where frames whose probabilities are less than 0.4 are excluded. The lyrics (best character sequence) are determined using a CTC decoder based on beam search [24] with a beam size of 25 frames and a 5-gram language model trained on the LibriSpeech corpus with a vocabulary of 200K words.

4. EVALUATION

This section reports a comparative experiment conducted for evaluating the effectiveness of the multi-task learning with pitch and onset estimation in ALT.

Details about data selection and training conditions can be found in the GitHub repository [1] of this paper.

4.1 Experimental Conditions

We explain the data used for evaluation, network configurations, compared methods, and evaluation measures.

4.1.1 Data

We used the DALI dataset [20] consisting of 5358 pieces of popular music in the English language along with fine-grained lyrics and pitch annotations. This dataset was made by collecting karaoke subtitle data and then searching for the corresponding audio data on the Internet, where an automatic alignment method was used for obtaining aligned

[1] https://github.com/TengyuDeng/lyrics-transcription-with-pitch-onset/

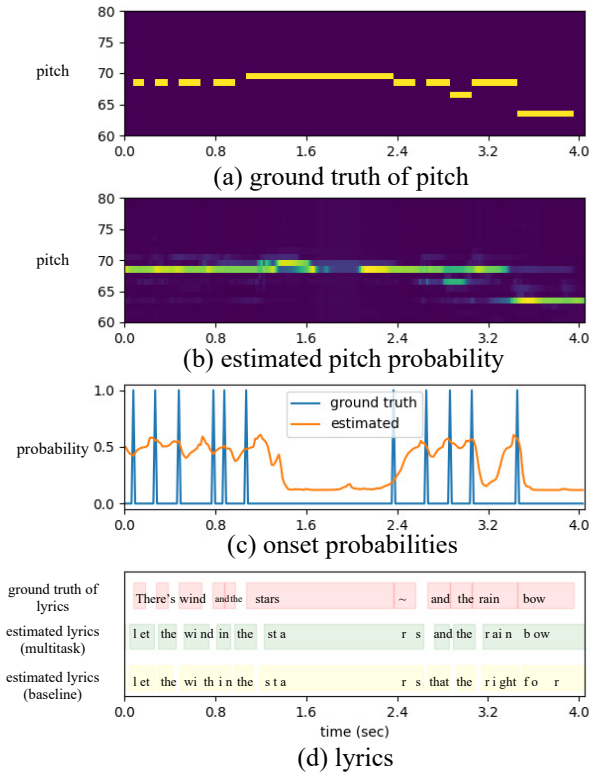

Figure 3. Example of estimations on a segment of a popular song from DALI. In (d), an estimated character is placed where the CTC decoder responds with the peak probability.

annotations. This dataset thus suffers from severe annotation errors [25]. First, the original karaoke data contained many problematic annotations such as global pitch shifts, wrong spellings, and onset time errors. Second, the automatic global alignment method often failed.

We thus filtered out music data with obviously wrong lyrics annotations and/or global pitch shift errors. The data selection procedure was based on the pitch estimation model and RWC Popular Music Database [26, 27]. This dataset contains 80 Japanese songs and 20 English songs, all of which are original popular songs and are provided with careful manual annotations. For simplicity, we used the framewise error rate given by

$$1 - \frac{\#\{\text{Frames Estimated Correctly}\}}{\#\{\text{Total Frames}\}}. \qquad (10)$$

We chose the 80 Japanese tracks of the RWC database and split them into a training set of 64 tracks and a test set of 16 tracks. The CRNN-based pitch estimation was trained on the training set, and attained 26.05% error rate on the test set, which was considered as a standard level.

The model trained on the training set from the RWC database was then exploited to test on the whole DALI dataset. The annotation of each song was shifted by -12, -11, ..., 0, ..., 11, and 12 semitones, and the predicted pitches of the pre-trained model were compared to all 25 pitch-shifted versions of annotations. The pitch shift value with the minimal framewise error rate was considered as the global pitch shift value of the annotation for a song, and

the minimal framewise error rate was recorded. Finally, the DALI dataset was filtered by the recorded framewise error rates. Specifically, songs with framewise error rates larger than a certain threshold were considered to be with problematic annotations.

Of all 5358 songs in the DALI dataset, 3272 songs could be accessed in our region and were in the English language. We selected 2515 songs with the data selection procedure, with a threshold of 50%. They were then split into a train set of 2263 songs, a validation set of 125 songs, and a test set of 126 songs. The total durations were 148.82 hours, 8.16 hours, and 8.47 hours, respectively.

For the transfer learning procedure, we used the LibriSpeech corpus [22] that contains 1000 hours of reading English speech sampled at 16 kHz. The CRNN described in Section 3.1.2 was trained on the train-clean-360 and train-other-500 sets of the LibriSpeech corpus, where all-zero matrices were used as $p(\text{pitch}|\mathbf{X})$ and $p(\text{onset}|\mathbf{X})$ in Eq. (7). This model was then fine-tuned with the training set of the DALI dataset.

4.1.2 Configurations

As described in Section 3.1, the audio signals, sampled at a rate of 48 kHz, were first separated into singing-voice-only signals with model umxhq from open-unmix [18]. Then for computational simplicity, the separated signals and the original mixed signals were resampled to 16 kHz, before they were converted to mel-spectrogram features, respectively. The audio signals were converted to mel-spectrogram with a window size of 32 ms and a hop length of 16 ms, and the resulted mel-spectrogram contained 80 mel-scaled features. We clipped the mel-spectrograms into pieces of 1000 frames, with a duration of about 16 s. Therefore, following the annotations in 3.1.1, we had $C = 2, F = 80, T = 1000$.

In the pitch and onset estimation network, 6 residual convolution blocks were stacked. The kernel sizes were (5,5), (5,5), (3,3), (3,3), (3,3), (1,1), and the numbers of output channels were 64, 32, 32, 32, 32, 1, respectively. After that, only 1 RNN block was applied to obtain the pitch and onsest estimation results. The dropout probability was set to 0 in each layer. In other words, we didn't adopt dropout in the pitch and onset estimation network.

In the lyrics transcription network, 6 residual convolution blocks were stacked. The kernel sizes were (5,5), (5,5), (3,3), (3,3), (3,3), (3,3), and the numbers of output channels were 64, 32, 32, 32, 32, 16, respectively. After that, 3 RNN blocks were stacked, and the number of hidden units in each LSTM was 512. The dropout probability was set to 0.2 in each layer.

When training the model, the Adam optimizer was used, and the parameters were $\beta_1 = 0.9, \beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The model was trained with a learning rate of 5×10^{-4}, and a warming up strategy was used. The learning rate linearly increased from 0 to 5×10^{-4} for the first 100 batches. We used an early stop strategy to manage the learning process. As mentioned in Section 4.1.1, the dataset was split into a training, a validation, and a test set. After training for an epoch, the model was tested on the validation set, and the learning process was terminated when the test statistics for the validation set stopped improving for 10 epochs.

Method	DALI-test	Jamendo
Ours (baseline)	69.22	77.3
Ours (oracle)	**64.41**	/
Ours (multi-task)	68.29	**76.2**
[6]	/	77.8
[7]	/	87.9

Table 1. Comparison of WER (%) with different methods.

4.1.3 Compared Methods

In order to test the performance of our system, we compared a couple of different settings. We also compared the proposed system with previous related works.

For lyrics transcription, in addition to the multi-task architecture, we also trained a model using zero dummy pitch and onset probabilities as a baseline model. Besides, a model was also trained using the ground truth pitches and onset times as oracle information. In order to compare with related works, the multi-task architecture was also tested on the jamendo dataset [6], and the results were compared with the end-to-end models in [6] and [7].

For pitch and onset estimation, we trained the model in Section 3.1.1 without the lyrics transcription part as the baseline model, and the results were compared with that in the multi-task scenario. We also evaluated the test data on VOCANO [28], a note-level vocal melody estimation toolkit available in public.

4.1.4 Evaluation Measures

The lyrics transcription was evaluated using word error rate (WER). The pitch and onset estimation was evaluated using the method first proposed in [29]. We considered the Correct Onset (COn) and the Correct Onset, Pitch (COnP) measures.

4.2 Experimental Results

Before fine-tuning on lyrics data, the lyrics transcription model was trained on LibriSpeech ASR corpus. The model reached a WER of 6.56% on the test-clean dataset and 20.86% on the test-other dataset.

WERs on the DALI-test dataset and the jamendo dataset are shown in Table. 1. On the DALI-test dataset, where the ground truth of pitch and onset information was available, the model reached the best performance when provided with this ground truth information. This shows that correct pitch and onset information can guide the system to find the correct alignment, so that the performance can be increased. In the multi-task architecture, the lyrics transcription model was trained with joint pitch and onset estimation. Although not as good as the oracle-given situation, the performance still gained some improvement. On the jamendo dataset, our multi-task architecture achieved

	COn			COnP		
	precision (%)	recall (%)	F value (%)	precision (%)	recall (%)	F value (%)
Ours(baseline)	53.21	**30.99**	**38.77**	36.92	**21.49**	**26.90**
Ours(multi-task)	**59.84**	28.69	38.41	**40.49**	19.57	26.14
VOCANO [28]	18.78	20.45	19.07	7.46	7.71	7.40

Table 2. Comparison of pitch and onset estimation results.

a similar improvement compared with our baseline model and beat main previous end-to-end ALT systems.

Table. 2 shows the pitch and onset evaluation results. Compared to the baseline model, the pitch and onset estimation jointly trained with the lyrics transcription model remained the same performance. However, for both the COn and COnP statistics, the multi-task scenario had a higher precision but a lower recall value than the baseline model. This shows that being jointly trained with the lyrics transcription model, especially the onset estimation was guided to be in favor of more confident onset positions. This lead to higher precision and lower recall values. It is notable that our models, both the baseline and the multi-task scenario, also gained better results than the results obtained when the VOCANO system was applied to the same test dataset.

5. CONCLUSION

This paper has presented a neural ALT method based on a multi-task learning architecture that estimates the pitch and onset information jointly and then transcribes the lyrics at the character level in a pitch- and onset-conditioned manner. The experiment using the DALI dataset showed that joint pitch and onset estimation can improve the performance of lyrics transcription. Although no significant overall improvement was attained in pitch and onset estimation, higher precision but lower recall rates were observed in the multi-task learning scenario. Our future work includes more comprehensive evaluation by gathering reliable data with accurate aligned lyrics and pitch annotations and using a data augmentation technique.

6. ACKNOWLEDGEMENT

This work is supported in part by JST PRESTO No. JPMJPR20CB and JSPS KAKENHI Nos. 19H04137, 20K21813, 21K02846, 21K12187, 22H03661.

7. REFERENCES

[1] E. Demirel, S. Ahlbäck, and S. Dixon, "Automatic lyrics transcription using dilated convolutional neural networks with self-attention," in *Proc. of International Joint Conference on Neural Networks*, 2020, pp. 1–8.

[2] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proc. of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. of Advances in Neural Information Processing Systems*, 2017.

[4] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang, S. Watanabe, T. Yoshimura, and W. Zhang, "A comparative study on transformer vs rnn in speech applications," in *Proc. of IEEE Automatic Speech Recognition and Understanding Workshop*, 2019, pp. 449–456.

[5] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[6] D. Stoller, S. Durand, and S. Ewert, "End-to-end lyrics alignment for polyphonic music using an audio-to-character recognition model," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 181–185.

[7] C. Gupta, E. Yılmaz, and H. Li, "Automatic lyrics alignment and transcription in polyphonic music: Does background music help?" in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 496–500.

[8] S. Basak, S. Agarwal, S. Ganapathy, and N. Takahashi, "End-to-end lyrics recognition with voice to singing style transfer," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 266–270.

[9] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, D. FitzGerald, and B. Pardo, "An overview of lead and accompaniment separation in music," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 8, pp. 1307–1335, 2018.

[10] C. Raffel, B. McFee, E. J. Humphrey, J. Salamon, O. Nieto, D. Liang, and D. P. W. Ellis, "mir_eval: A transparent implementation of common MIR metrics," in *Proc. of the 15th International Society for Music Information Retrieval Conference*, Taipei, Taiwan, 2014, pp. 367–372.

[11] J. Pons, R. Gong, and X. Serra, "Score-informed syllable segmentation for a cappella singing voice with convolutional neural networks," in *Proc. of the 18th*

International Society for Music Information Retrieval Conference, Suzhou, China, 2017, pp. 383–389.

[12] J. Huang, E. Benetos, and S. Ewert, "Improving lyrics alignment through joint pitch detection," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022.

[13] K. S. Rao, P. P. Das *et al.*, "Melody extraction from polyphonic music by deep learning approaches: A review," *arXiv preprint arXiv:2202.01078*, 2022.

[14] J. Salamon, E. Gomez, D. P. W. Ellis, and G. Richard, "Melody extraction from polyphonic music signals: Approaches, applications, and challenges," *IEEE Signal Processing Magazine*, vol. 31, no. 2, pp. 118–134, 2014.

[15] T.-H. Hsieh, L. Su, and Y.-H. Yang, "A streamlined encoder/decoder architecture for melody extraction," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 156–160.

[16] S. Yu, X. Sun, Y. Yu, and W. Li, "Frequency-temporal attention network for singing melody extraction," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 251–255.

[17] R. Nishikimi, E. Nakamura, M. Goto, and K. Yoshii, "Audio-to-score singing transcription based on a crnn-hsmm hybrid model," *APSIPA Transactions on Signal and Information Processing*, vol. 10, 2021.

[18] F.-R. Stöter, S. Uhlich, A. Liutkus, and Y. Mitsufuji, "Open-unmix – a reference implementation for music source separation," *Journal of Open Source Software*, 2019.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, November 1997.

[20] G. Meseguer-Brocal, A. Cohen-Hadria, and G. Peeters, "Dali: A large dataset of synchronized audio, lyrics and notes, automatically created using teacher-student machine learning paradigm," in *Proc. of the 19th International Society for Music Information Retrieval Conference*, Paris, France, 2018, pp. 431–437.

[21] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[22] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 5206–5210.

[23] S. Böck, F. Krebs, and M. Schedl, "Evaluating the online capabilities of onset detection methods," in *Proc. of the 13th International Society for Music Information Retrieval Conference*, Porto, Portugal, 2012, pp. 49–54.

[24] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[25] G. M. Brocal, R. Bittner, S. Durand, and B. Brost, "Data cleansing with contrastive learning for vocal note event annotations," in *Proc. of the 21st International Society for Music Information Retrieval Conference*, Montreal, Canada, 2020, pp. 255–262.

[26] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "Rwc music database: Popular, classical and jazz music databases," in *Proc. of the 3rd International Conference on Music Information Retrieval*, Paris, France, 2002.

[27] M. Goto, "Aist annotation for the rwc music database," in *Proc. of the 7th International Conference on Music Information Retrieval*, Victoria, Canada, 2006, pp. 359–360.

[28] J.-Y. Hsu and L. Su, "Vocano: A note transcription framework for singing voice in polyphonic music," in *Proc. of the 22nd International Society for Music Information Retrieval Conference*, Online, 2021, pp. 293–300.

[29] E. Molina, A. M. Barbancho, L. J. Tardón, and I. Barbancho, "Evaluation framework for automatic singing transcription," in *Proc. of the 15th International Society for Music Information Retrieval Conference*, Taipei, Taiwan, 2014, pp. 567–572.