Music Chord Recognition From Audio Data Using Bidirectional Encoder-decoder LSTMs

Takeshi Hori* and Kazuyuki Nakamura† and Shigeki Sagayama‡
* Meiji University, Tokyo, Japan
E-mail: hori@meiji.ac.jp
† Meiji University, Tokyo, Japan
E-mail: knaka@meiji.ac.jp
‡ Meiji University, Tokyo, Japan
E-mail: sagayama@meiji.ac.jp

***Abstract*—In this paper, we discuss some methods for chord recognition based on long short-term memory recurrent neural networks (LSTM, LSTM-RNN). Chord progressions play an important role in the generation process of music. Actually, music processing systems containing a model for chord progressions achieve high accuracies in tasks like music structure analysis, multi pitch analysis an automatic composition or accompaniment.**

In previous research, chord progressions were obtained rule-based or have been modeled using stochastic methods like hidden Markov models or probabilistic context-free grammars. Pitch patterns were then regarded as the observations resulting from the hidden states of the chord progression model. Recently, convolutional neural networks have been used for chord recognition with considerable success. On the other hand, LSTM networks have been shown to be suitable for generating chord progressions, since these neural networks can process time series data very well.

The purpose of this study is to evaluate and compare three types of LSTM networks based on the bidirectional and encoder-decoder structure with regards to their chord recognition performance. In order to extract more effective data for chord recognition, we use a constant-Q transform and specmurt analysis to suppress overtone components, and chroma vectorization to reduce the feature dimensionality.

The evaluation results show that the encoder-decoder-based LSTM can learn the relationship between the observed chroma vectors and the associated chord progression more effectively than simpler LSTM networks.

I. Introduction

Harmony, which is the foundation of Western music, is an important element in music analysis. Although in recent years several genres without tonality like twelve-tone music and free jazz have become popular, music analysis based on harmony has not lost its importance, since most of the currently produced music is still bound to the concept of tonality.

Stochastic models that take harmonic structure into account are utilized for various tasks, including automatic harmonization of melodies [1], automatic arrangement for guitar [2], automatic music transcription [3], multi pitch analysis of polyphonic music [4], sound source separation [5] and other disciplines in the field of music information retrieval [6]. These approaches often use an inverse problem formulation based on a generative model of music including concepts like tonality and chords.

Such models of harmonic structure can be regarded as equivalent to language models for voice recognition, where concatenation of words can be probabilistically modeled using n-grams or formal grammars. Similarly, chord progressions are traditionally based on musical rules and patterns: While for instance the sentence structure "subject - verb - object" is that of a viable sentence in the English language, cadences like "I - IV - V - I" occur frequently in western music and provide the basis for functional harmony theory.

Due to this similarity, it is reasonable to assume that one can apply methods of the field of research of speech and language analysis to music analysis problems as well. In fact, several such methods have already been applied successfully for music information processing tasks. This includes n-grams and hidden Markov models (HMM) [7] as well as probabilistic context-free grammars (PCFG) [8]. One of the recently very successful methods for speech and language processing is the long short-term memory recurrent neural network (LSTM, LSTM-RNN) [9]. On the basis of these previous successes, it is reasonable to assume that LSTM networks could achieve high precision in the field of music information retrieval such as the chord recognition task discussed in this paper.

II. Chord recognition

A human trained in musical theory can recognize a tonic key or chords from a melody (pitch information). If a computer could similarly extract precise pitch information from audio material and learn the relationship between pitches and chords, it could estimate chords from audio input.

Fig. 1 illustrates the processing steps of the chord recognition system used for this paper. The system receives an audio signal in WAVE format as input, which is then preprocessed to extract information that is most relevant to chord recognition. The signal is first converted to a logarithmically scaled spectrogram using a constant Q transform, projecting musical intervals on constant distances in the constant Q transform spectrogram. Since overtone frequencies produced by musical instruments complicate the estimation of precise pitches, we use a specmurt analysis [10], which accounts for overtone components by treating the audio signal as a convolution of the "clean" spectrogram (containing only the fundamental

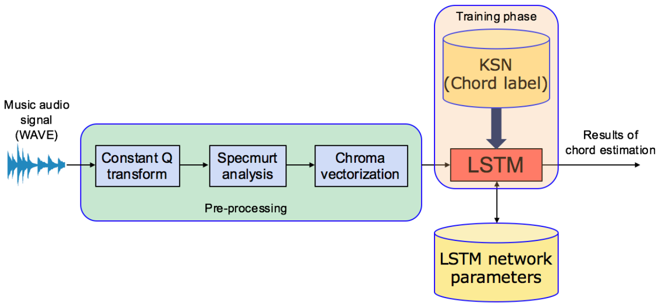

Fig. 1. Processing steps of the chord recognition system. Using a constant Q transform, specmurt analysis and chroma vectorization, a time series of vectors is computed from an audio signal. The relationship between the vectors and chords is learned by LSTM networks.

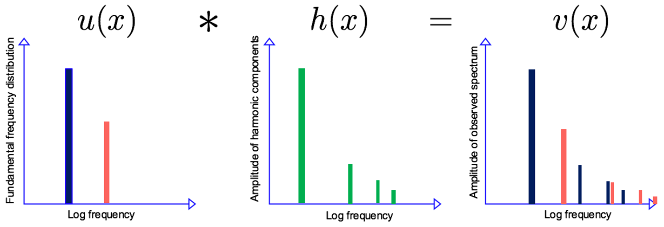
$u(x)$
$*$
$h(x)$
$=$
$v(x)$
Fundamental frequency distribution
Log frequency
Amplitude of harmonic components
Log frequency
Amplitude of observed spectrum
Log frequency

Fig. 2. Given the common harmonic structure pattern $h(x)$, the fundamental frequency distribution $u(x)$ can be estimated from a logarithmically scaled spectrum $v(x)$ using the specmurt analysis

frequencies of each note) with the overtone spectrum of the performing instrument. In addition, we compute chroma vectors [11], [12] from the resulting spectrograms. A chroma vector contains 12 elements, each containing the sum of a tone's (e.g. C#) spectrum magnitudes at all octaves. The computer then learns the relationship between the time series of the resulting chroma vectors and labeled chords.

A. Specmurt analysis

Specmurt analysis is used for multi pitch analysis of spectrograms of audio signals of recorded musical instruments which usually produce overtone frequencies. In the specmurt analysis, the power ratio of the harmonic overtones is assumed to be the independent from the absolute value of the respective fundamental frequency. Although this assumption is only an approximation, it allows to easily suppress a significant amount of overtone frequencies in spectrograms of recordings of a single instrument. If one scales the frequency domain logarithmically, the distances between the overtone frequencies become constant and independent from the fundamental frequency. Therefore, in a logarithmically scaled frequency domain x, one can approximate the spectrum of an audio signal of a single instrument $v(x)$ using the common harmonic structure pattern $h(x)$ and convolving it with the fundamental frequency distribution $u(x)$ as shown in Fig. 2:

$$v(x) = u(x) * h(x). \quad (1)$$

Due to the convolution theorem of the Fourier transform $\mathcal{F}$, the following equation holds for the Fourier transformed signals $\mathcal{F}[u(x)](y), \mathcal{F}[v(x)](y)$ and $\mathcal{F}[h(x)](y)$:

$$\mathcal{F}[u(x)](y) = \frac{\mathcal{F}[v(x)](y)}{\mathcal{F}[h(x)](y)}. \quad (2)$$

implying that the fundamental frequency distribution $u(x)$ can be obtained as follows:

$$u(x) = \mathcal{F}^{-1}\left[\frac{\mathcal{F}[v(x)](y)}{\mathcal{F}[h(x)](y)}\right](x). \quad (3)$$

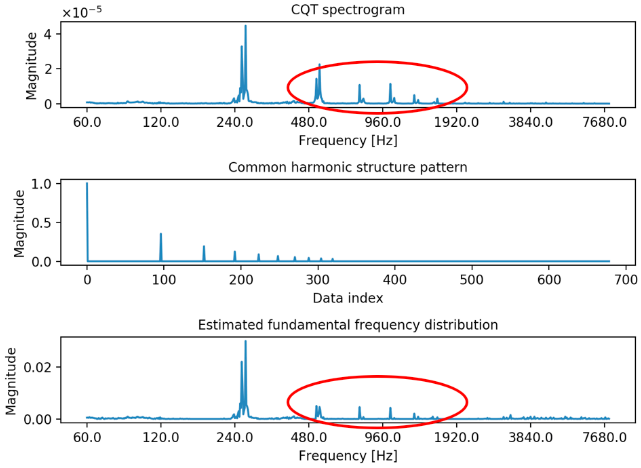

Fig. 3. Experimental results of applying the specmurt analysis to a recording of the note C4 played on a piano (taken from the RWC Music Database [14]).

TABLE I
PARAMETERS USED FOR THE SPECMURT ANALYSIS

Frame shift	10 ms
Lowest frequency	60 Hz
Highest frequency	8000 Hz
Frequency resolution	12.5 cent
Overtone magnitude attenuation factor used in $h(x)$	$\alpha = 1.5$
Number of harmonics considered in $h(x)$	10

To obtain a logarithmically scaled power spectrum as required for the specmurt analysis, one could rescale a linear spectrum obtained using the short-time Fourier transform (STFT). However, it is more effective to directly compute such a logarithmically scaled spectrum using a constant Q transform (CQT) [13]. We used a CQT filter bank with 96 filters per octave in a range between 60 Hz and 8000 Hz. The spectrograms were computed with a time resolution of 100 frames per second, i.e. a frame shift of 10 ms.

Fig. 3 shows the results of the specmurt analysis of a recording of the note C4 (261.6Hz) played on a piano (taken from the RWC Music Database [14]). The common harmonic structure pattern $h(x)$ used for this paper contains peaks (with a width of a single bin) for 10 harmonics (including the fundamental frequency). The magnitude of these peaks decreases with distance from the fundamental frequency according to the following formula:

$$\frac{h(f_n)}{h(f_0)} = (n+1)^{-\alpha}$$

where f_n is the index of the n-th harmonic and α is an attenuation factor for which the value 1.5 was chosen for this paper. All parameters of the specmurt analysis are listed in table I. As can be seen in Fig. 3, the specmurt analysis does not completely remove the overtone frequencies due to its approximative character, but it significantly reduces their magnitudes. Negative values obtained from the specmurt analysis are set to 0 before further data processing is applied.

B. Chroma vector

In chord recognition, using the combined magnitude of pitch classes like C or D can be more effective than using detailed magnitude information of every individual pitch C4, C5, or D4. The combined magnitude information is computed as 12-dimensional chroma vectors [11]. Each element of a chroma vector corresponds to a semitone class of western tonal music. Each bin of the CQT spectrogram is assigned to the semitone that it is closest to (considering the logarithmic scaling of semitone frequencies) and an element of a chroma vector is the computed as the sum of all bin magnitudes belonging to the respective semitone class. The obtained chroma vectors are normalized to mean 0 and variance 1.

III. LSTM-BASED TRAINING

In recent years, stacked LSTM networks which consist of multiple hidden LSTM unit, as well as bidirectional LSTM networks [15] into which time series vectors are input in both forward and backward direction, have been used with considerable success. For this paper, we utilized a encoder-decoder LSTM network (ED-LSTM) as well as the stacked/bidirectional LSTM [16] network architecture. The ED-LSTM network can not only deal with data containing different sequence lengths but is also able to directly map sequences to sequences (the chord progression states in our system). Because of their properties, encoder-decoder models are often used in research on automatic language translation.

We compared the following 3 LSTM networks to demonstrate the usability of the bidirectional and encoder-decoder LSTM architectures for the chord recognition task.

1) *Stacked bidirectional LSTM network (SBi-LSTM)*
A neural network consisting of 3 stacked bidirectional LSTM units.
2) *Stacked bidirectional ED-LSTM network (SBiED-LSTM)*
The output of 3 stacked bidirectional LSTM units (see Fig. 4) is used as the input of an encoder layer of an encoder-decoder structure. This network is able to train label estimation from input data, as well as to explicitly learn time series characteristics of the label data.
3) *Conditional stacked bidirectional ED-LSTM network (CSBiED-LSTM)*
We propose a LSTM network architecture in which the output of three bidirectional LSTM units is used as input for both an encoder layer as well as a classification layer. The output of said classification layer is then used as input of another 3 bidirectional LSTM units whose output in turn is used as the input of the decoder layer connected to the previously mentioned encoder layer (see Fig. 5).

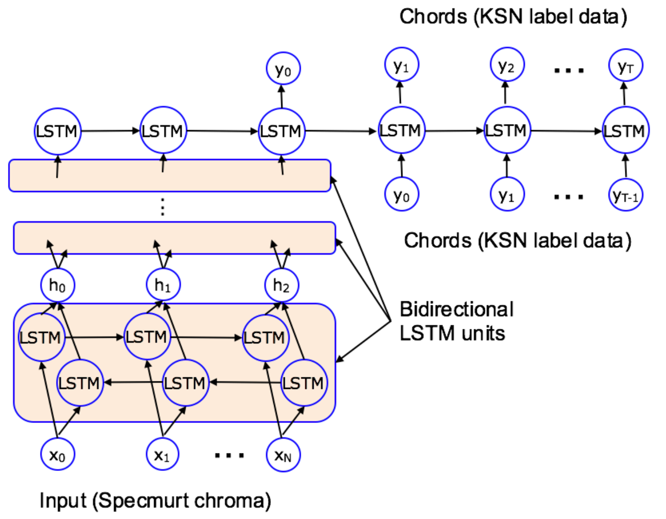

Fig. 4. In the SBiED-LSTM network, the chroma vector input is first processed by three bidirectional LSTM units, whose output is used as the input of an encoder layer of an encoder-decoder structure.

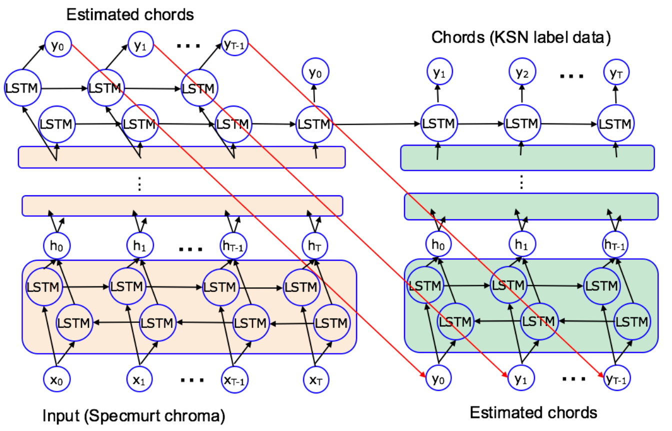

Fig. 5. In the CSBiED-LSTM network, stacked bidirectional LSTM units are used for preprocessing both the input of the encoder layer as well as that of the decoder layer.

IV. EXPERIMENTAL EVALUATION

We trained our system using 8 musical pieces from the classic database of the RWC Music Database and evaluated the chord recognition rate utilizing cross validation (using another musical piece from same database). The chord progression labels were obtained from the KS notation database (KSN) [17]. The label data was created by music university students and experts trained in harmony theory. The notation itself is based on functional harmony theory and therefore especially suited for music information analysis based on chord progressions.

For this paper, we used 25 chord classes: 12 major chords, 12 minor chords, and a "no chord" class. The training data

TABLE II
PARAMETERS OF THE LSTM NETWORK TRAINING PROCESS

Tempo	120bpm
Tonic key	C Major
Number of stacked LSTM layers	3
Number of training epochs	5000
Number of musical pieces	8

TABLE III
EXPERIMENTAL RESULTS (CHORD RECOGNITION)

	(1)SBi	(2)SBiED	(3)CSBiED
Accuracy (%)	76.5	85.8	82.3

was available in MIDI and WAVE format where the MIDI onset timings correspond as closely as possible to the real performance recorded in the WAVE data. In our experiments, we used the MIDI data and transposed every piece to C-Major. In a second step, we converted the MIDI data to audio data after scaling each MIDI file to a tempo of 120 beats per minute. Lastly, the audio data was split into segments each 4 seconds long. The LSTM networks were then trained to recognize the chord progressions of these 4 second segments. Table II shows the parameters of the neural network training process.

Each LSTM network was trained a data set containing around 1000 segments taken from the 8 classical pieces used for training. The chord recognition accuracy was evaluated using approximately 120 segments of the validation piece. The evaluation results are shown in Table III. The stacked bidirectional encoder-decoder LSTM network achieved the highest chord recognition accuracy rate. The result of the conditional stacked bidirectional encoder-decoder LSTM network was was relatively close, and one can see that neural network structures utilizing an encoder-decoder model performed significantly better than a simple stacked bidirectional LSTM network. A possible reason for the worse performance of the CSBiED LSTM network in comparison with its simpler SBiED LSTM counterpart could be the significant increase of neural network weights leading to overfitting due to the relatively small amount of available training data.

V. CONCLUSION

We evaluated 3 types of chord recognition LSTM network architectures which utilize the bidirectional LSTM structure as well as the encoder-decoder model. During preprocessing, our system applied specmurt analysis to suppress overtone frequencies in the audio spectrograms. The experimental results showed that the encoder-decoder model is effective for the recognition of chord progressions.

However, there is still room for improvement: We used the approximation that the magnitudes of overtone frequencies decrease simply inversely with increasing distance from the fundamental frequency. However, the exact overtone frequency distribution is dependent on the musical instrument as well as the recording environment and even slightly differs for different pitches. If one estimates the overtone frequency distribution from data, the performance of the specmurt analysis could increase, resulting in more effective suppression of overtone components [18]. Another possibility for improving preprocessing could be the use of non-negative matrix factorization (NMF), which has been used in recent research on multi pitch analysis and a sound source separation [19].

REFERENCES

[1] T. Kawakami, m. Nakai, H. Shimodaira, and S. Sagayama, "Hidden markov model applied to automatic harmonization of given melodies," *IPSJ SIG Technical Reports*, vol. 2000, no. 19 (1999-MUS-034), pp. 59–66, 2000.

[2] G. Hori, H. Kameoka, and S. Sagayama, "Input-output hmm applied to automatic arrangement for guitars," *Information and Media Technologies*, vol. 8, no. 2, pp. 477–484, 2013.

[3] 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 (TASLP)*, vol. 24, no. 5, pp. 927–939, 2016.

[4] G. Peeters, "Chroma-based estimation of musical key from audio-signal analysis." in *ISMIR*, 2006, pp. 115–120.

[5] M. Mauch and S. Dixon, "Approximate note transcription for the improved identification of difficult chords." in *ISMIR*, 2010, pp. 135–140.

[6] M. Schedl, E. Gómez, J. Urbano *et al.*, "Music information retrieval: Recent developments and applications," *Foundations and Trends® in Information Retrieval*, vol. 8, no. 2-3, pp. 127–261, 2014.

[7] K. Lee and M. Slaney, "Automatic chord recognition from audio using a hmm with supervised learning." in *ISMIR*, 2006, pp. 133–137.

[8] D. Quick and P. Hudak, "Grammar-based automated music composition in haskell," in *Proceedings of the first ACM SIGPLAN workshop on Functional art, music, modeling & design*. ACM, 2013, pp. 59–70.

[9] M. Sundermeyer, R. Schlüter, and H. Ney, "Lstm neural networks for language modeling," in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[10] K. Takahashi, T. Nishimoto, and S. Sagayama, "Multi - pitch analysis using deconvolution of log - frequency spectrum," *IPSJ SIG Technical Reports*, vol. 2003, no. 127 (2003-MUS-053), pp. 61–66, 2003.

[11] T. Fujishima, "Realtime chord recognition of musical sound: a system using common lisp music." in *ICMC*, 1999, pp. 464–467.

[12] M. A. Bartsch and G. H. Wakefield, "To catch a chorus: Using chroma-based representations for audio thumbnailing," in *Applications of Signal Processing to Audio and Acoustics, 2001 IEEE Workshop on the*. IEEE, 2001, pp. 15–18.

[13] J. C. Brown, "Calculation of a constant q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.

[14] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "Rwc music database : Database of copyright-cleared musical pieces and instrument sounds for research purposes," *IPSJ Journal*, vol. 45, no. 3, pp. 728–738, 2004.

[15] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5, pp. 602–610, 2005.

[16] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[17] H. Kaneko, D. Kawakami, and S. Sagayama, "Functional harmony annotation database for statistical music analysis," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR): Late Breaking session*, 2010.

[18] S. Saito, H. Kameoka, K. Takahashi, T. Nishimoto, and S. Sagayama, "Specmurt analysis of polyphonic music signals," *IEEE transactions on audio, speech, and language processing*, vol. 16, no. 3, pp. 639–650, 2008.

[19] M. Nakano, J. Le Roux, H. Kameoka, N. Ono, and S. Sagayama, "Infinite-state spectrum model for music signal analysis," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1972–1975.