Automatic melody extraction algorithm using a convolutional neural network

Jongseol Lee[1,2], Dalwon Jang[1] and Kyoungro Yoon[2]
[1]Communications & Media R&D,
Korea Electronics Technology Institute, Gyeong-gi, Korea
[2]Department of Computer Science and Engineering,
Konkuk University, Seoul, Korea
[e-mail: {leejs, dalwon}@keti.re.kr, yoonk@konkuk.ac.kr}]
*Corresponding author: Kyoungro Yoon

Received October 31, 2016; revised January 18, 2017; revised August 10, 2017; accepted September 15, 2017; published December 31, 2017

Abstract

In this study, we propose an automatic melody extraction algorithm using deep learning. In this algorithm, feature images, generated using the energy of frequency band, are extracted from polyphonic audio files and a deep learning technique, a convolutional neural network (CNN), is applied on the feature images. In the training data, a short frame of polyphonic music is labeled as a musical note and a classifier based on CNN is learned in order to determine a pitch value of a short frame of audio signal. We want to build a novel structure of melody extraction, thus the proposed algorithm has a simple structure and instead of using various signal processing techniques for melody extraction, we use only a CNN to find a melody from a polyphonic audio. Despite of simple structure, the promising results are obtained in the experiments. Compared with state-of-the-art algorithms, the proposed algorithm did not give the best result, but comparable results were obtained and we believe they could be improved with the appropriate training data. In this paper, melody extraction and the proposed algorithm are introduced first, and the proposed algorithm is then further explained in detail. Finally, we present our experiment and the comparison of results follows.

Keywords: melody extraction, convolutional neural network, train–test framework

This research is supported by the Ministry of Culture, Sports, and Tourism (MCST) and the Korea Creative Content Agency (KOCCA) in the Culture Technology (CT) Research & Development Program 2016.

https://doi.org/10.3837/tiis.2017.12.019 **ISSN : 1976-727**

1. Introduction

As a result of the recent proliferation of digital music, the field of music information retrieval has received increasing attention [1,2]. In this area of research, challenging tasks exist. These include query-by-singing/humming [3, 4], tempo estimation [4, 5], cover-song identification [6], music genre classification [7-9], music mood classification [10,11], audio fingerprinting [12,13], downbeat estimation [14], audio tagging [15], automatic melody extraction [16-25], and others. Among the various areas of research in music information retrieval, automatic melody extraction extracts pitch or chroma information from music clips, generally polyphonic music recordings. The output of an automatic melody extraction system is a frequency-based representation of the melody (i.e., frequency values of the frames), and from this melody extraction output, symbolic notations can be generated. The process of creating symbolic notations, together with extracting frequency-based representations of melodies, is called melody transcription: in this study, we address only melody extraction. The melody is the basic information used to analyze the music signal; thus, a melody extraction system can be applied to other music information retrieval systems such as query-by-singing/humming, transcription, music classification, and others [18]. We assume that the input of an automatic melody extraction system is polyphonic music, and the system should extract raw pitch or chroma information about a main melody. By main melody here, we refer to the melody of a lead vocal or lead instrument, and it is commonly the most powerful signal in polyphonic music. Sometimes, an automatic melody extraction system can be used to extract the melody of a specific musical instrument. In such a case, the signal of the musical instrument is first separated, and the melody is extracted from the separated signal. The general structure and existing algorithms of automatic melody extraction can be found in [18].

We attempt to build a novel method of melody extraction using a convolutional neural network (CNN). Melody extraction can be realized using just a signal processing technique. The object of melody extraction is simply to find the frequency of the main melody from a frame of music data, and it can, to some degree, be realized using spectral transform and finding the maximum. To enhance the accuracy, methods based on the connectivity of a melody, a probabilistic melody model, or processing based on a filter are applied. By combining such modules, an entire melody extraction algorithm can be realized. Information on state-of-the-art algorithms capable of melody extraction and their performance can be found in the Music Information Retrieval Evaluation eXchange (MIREX) [27]. However, enhancements to methods to achieving greater accuracy have reached their limit. Figure 7 of [18] shows the best overall accuracy results in MIREX. These have not improved considerably since 2010 because the basic structure of the algorithms has not changed. To achieve groundbreaking improvements in melody extraction accuracy, we believe that a novel approach is necessary. Thus, we propose a new algorithm based on a CNN for the identification of a single dominant pitch. From the music signal, the feature image, which is an input signal of the CNN, is extracted using the band energy, and the image is classified by the CNN. The output of the learning method is note information. Without the extra post-processing parts, it gives an output with a classifier. Minimizing heuristic design of feature, the algorithm depends on the power of classifier. This is a big difference from the existing algorithms, and we expect this difference will leads to a groundbreaking improvements with additional upgrading. In our experiments, results were obtained using three different datasets. Using a general evaluation framework of melody extraction, the

results of the proposed algorithm were compared with the results of state-of-the-art algorithms, and performance was similar.

This article is organized as follows. In section 2, we present previous works related to the proposed algorithm. In Section 3, we present the proposed melody extraction algorithm. In Section 3, the experimental results are shown. Section 4 concludes this study.

2. Previous works

2.1 melody extraction

Melody extraction algorithms can be simplified to five steps: pre-processing; spectral transform and processing; multi-pitch representation (salience function); tracking; and voicing [18]. Among these five steps, spectral transform and pitch tracking are the two most important. The melody implies frequency information, and the algorithms should search for the appropriate frequency value constituting the melody. Thus, spectral transformation is essential. In a common music signal, the melody has the property of continuity, and a pitch doubling or halving effect can easily occur. To avoid the pitch doubling or halving effect using continuity, pitch tracking is necessary. For example, if a musical note “A4” (440 Hz) is found in a frame, a melody extraction algorithm can easily misidentify this frame as that of musical note “A3” (220 Hz) or that of “A5” (880 Hz). By using the outputs of previous and subsequent frames, melody extraction algorithms can increase the possibility of identifying the frame as “A4”. For this, various methods like HMM, dynamic programming, rule-based method, Viterbi smoothing, and so on were used [18]. Rule-based methods that utilize the continuity of frames are found in most early melody extraction research. However, at least one study has shown that training-based methods such as the hidden Markov model (HMM) can enhance accuracy [24]. In [18], various melody extraction algorithms are summarized with the structure of 5 steps.

2.2 Train-test framework

In general, melody extraction algorithms are mostly based on various signal processing techniques. Some melody extraction algorithms are based partly on the train–test framework [17,25]. In a train–test framework, information is derived from a model that is learned from a training dataset. Some music information retrieval tasks use a train–test framework. Music genre/mood classification is such a task. Generally, it is based on a machine learning algorithm such as a support vector machine, a Gaussian mixture model, a linear discriminant analysis, or a *k*-nearest neighbor search. As written in [26], train-test frameworks based on neural network are recently used in order to get the better feature or the better classification performance [9,14,15,25]. In [9], CNN extracts musical pattern in MFCC for musical genre classification. In [14] and [15], CNN is used in downbeat tracking and audio tagging, respectively. For melody extraction, [17] and [25] were previously proposed with train-test framework. But, these two can be also summarized with 5 steps explained above. In [17], Ellis and Poliner previously attempted classification-based melody extraction with support vector machine. In [25], deep neural network is used to make a melody contour. In both [17] and [25], post-processing based on HMM was used as pitch tracking, and the extra process of finding voicing frame with simple thresholding method exists. But, in our algorithm, there are all processed in a single train-test framework of CNN.

3. Melody extraction algorithm

3.1 Structure

The structure of the proposed algorithm is shown in **Fig. 1**. As shown, the proposed algorithm has two parts: feature image extraction and a CNN. The algorithm does not possess any multi-pitch representation, tracking, or voicing parts, which are three of the five main parts presented in [18]. The feature image extraction process can be interpreted as a pre-processing and a spectral transform. Other parts are performed using a CNN. For easy comparison with other algorithms, the music input of the algorithm is designed using a 44100 Hz sampling rate, 16-bit quantization, and mono channel. In addition, a musical note is generated every 10 ms [33]. Details are given in the following subsections.

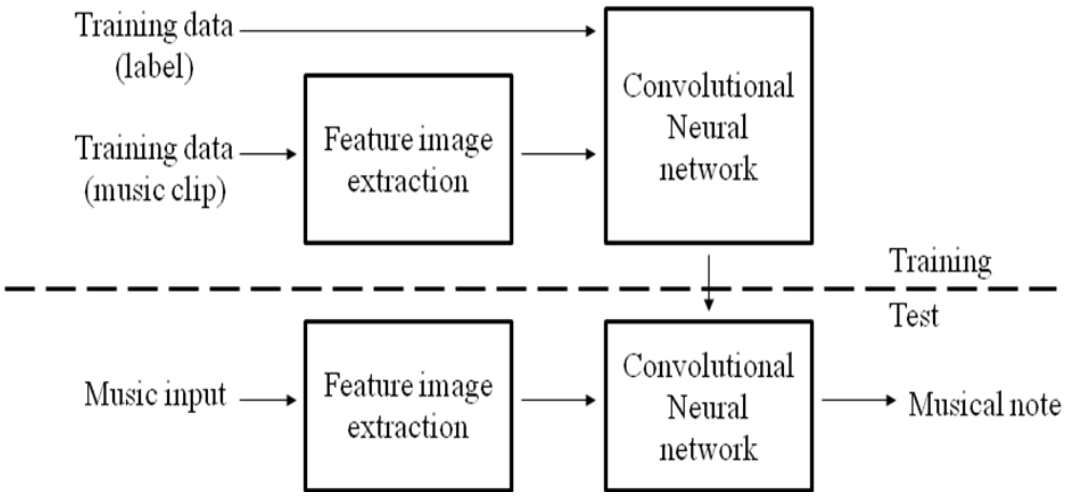

Fig. 1. Structure of a melody extraction algorithm

3.2 Convolutional neural network

A CNN is a type of feed-forward artificial neural network and is generally used with image signal processing, such as face recognition, handwritten character classification, and image classification [29-32]. With a CNN, features can be extracted from an image and images can be classified or an object (i.e., car, tree, man, dog) can be detected from images through the features. Using a CNN, researchers in image classification need not consider the design of feature extraction. However, because a melody extraction system does not deal with image signals, but rather audio signals, we must consider a method that generates image signals, which can be used as the input of a CNN. Various means can be employed to generate an input image from an audio signal. In the proposed algorithm, the simplest method is employed, which is based on the energy of consecutive frames.

Some signal processing techniques, the interest of which lies in music or speech signals, can identify vowel sounds or track melody lines using a spectrogram. Analyzing music signals using re-formatted image signals is common. Thus, from this approach we have derived the idea of using a CNN, which is generally used for image signal processing. We believe that with a CNN, we need not consider tracking, multi-pitch representation, or a method that identifies voicing frames. We expect these to be solved during the training stage.

In **Fig. 2**, the architecture of the CNN used in the proposed algorithm is shown. The size of input is 48 × 24, and in both layers, convolution with 5 × 5 filters and 2 × 2 subsampling are performed. After convolution and subsampling are performed in Layer 1, feature is converted to 22 × 10, and the number of maps is set to 20. After convolution and subsampling are performed in Layer 2, a feature is converted to 9 × 3, and the number of maps is set to 40. Various architectural forms are tested. Our experiments determined that the architecture presented in **Fig. 2** yields the best results.

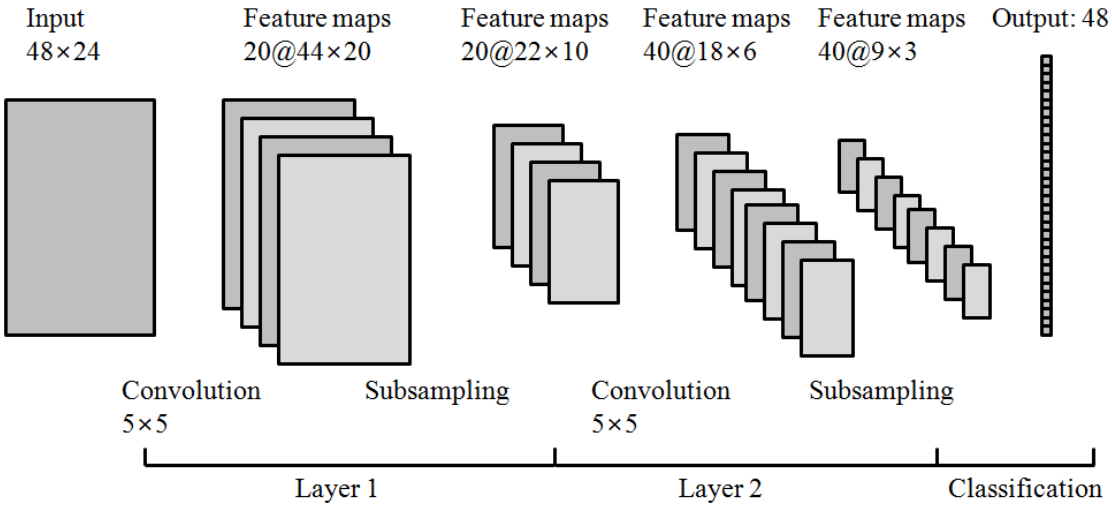

Fig. 2. Architecture of the CNN used in the proposed algorithm

3.3 Output label

In our system, 48 output labels (from 0 to 47) for each frame are set. Label 0 is assigned to the frame that contains no melody. Labels 1 to 47 are assigned to frames with notes starting from "F#2" up to "E6," with increments of half a tone. This means that the output of the CNN is one of 48 labels. The labels as well as the frequency bands related to the labels are shown in **Table 1**.

Table 1. The 48 output labels and their frequency bands.

Label	Note	Frequency	Band range
0	This frame contains no melody		
1	F#2	92.50	<95.21
2	G2	98.00	95.21–100.87
3	G#2	103.83	100.87–106.87
4	A2	110.00	106.87–113.22
5	A#2	116.54	113.22–119.96
6	B2	123.47	119.96–127.09
7	C3	130.81	127.09–134.65
8	C#3	138.59	134.65–143.66
9	D3	146.83	143.66–151.13
10	D#3	155.56	151.13–160.12
11	E3	164.81	160.12–169.64

12	F3	174.61	169.64–179.73
13	F#3	185.00	179.73–190.42
14	G3	196.00	190.42–201.74
15	G#3	207.65	201.74–213.74
16	A3	220.00	213.74–226.45
17	A#3	233.08	226.45–239.91
18	B3	246.94	239.91–254.18
19	C4	261.63	254.18–269.29
20	C#4	277.18	269.29–285.30
21	D4	293.66	285.30–302.30
22	D#4	311.13	302.30–320.24
23	E4	329.63	320.24–339.29
24	F4	349.23	339.29–359.46
25	F#4	369.99	359.46–380.84
26	G4	392.00	380.84–403.48
27	G#4	415.30	403.48–427.47
28	A4	440.00	427.47–452.89
29	A#4	466.16	452.89–479.82
30	B4	493.88	479.82–508.36
31	C5	523.25	508.36–538.58
32	C#5	554.37	538.58–570.61
33	D5	587.33	570.61–604.54
34	D#5	622.25	604.54–640.49
35	E5	659.25	640.49–678.57
36	F5	698.46	678.57–718.92
37	F#5	739.99	718.92–761.67
38	G5	783.99	761.67–806.96
39	G#5	830.61	806.96–854.95
40	A5	880.00	854.95–905.79
41	A#5	932.33	905.79–959.65
42	B5	987.77	959.65–1016.7
43	C6	1046.50	1016.7–1177.2
44	C#6	1108.73	1177.2–1141.2
45	D6	1174.66	1141.2–1209.1
46	D#6	1244.51	1209.1–1281.0
47	E6	1318.51	>1281.0

3.4 Data labeling

To create and then verify the proposed melody extraction system, training and test datasets are necessary. Some datasets developed to verify melody extraction algorithms exist, and they possess manually annotated frequency data with a 10 ms time grid. To use existing datasets in the proposed algorithm as training or test sets, they should be quantized and labeled. An example of quantization is given in **Fig. 3**. As shown, the frequency value of a real melody is not fixed, and these values are quantized to a fixed value.

Because we build a train–test framework-based melody extraction, the proposed system can only output frequency values of the 48 labels presented in **Table 1**. However, the range of notes included in an actual piece of music is much wider. Thus, any note that is lower than "F#2" is considered simply as "F#2" in our algorithm; similarly, any note that is higher than "E6" is considered simply as "E6". This is a structural flaw of a train–test framework. However, we considered it unusual for a melody to have any notes lower than "F#2" or higher than "E6".

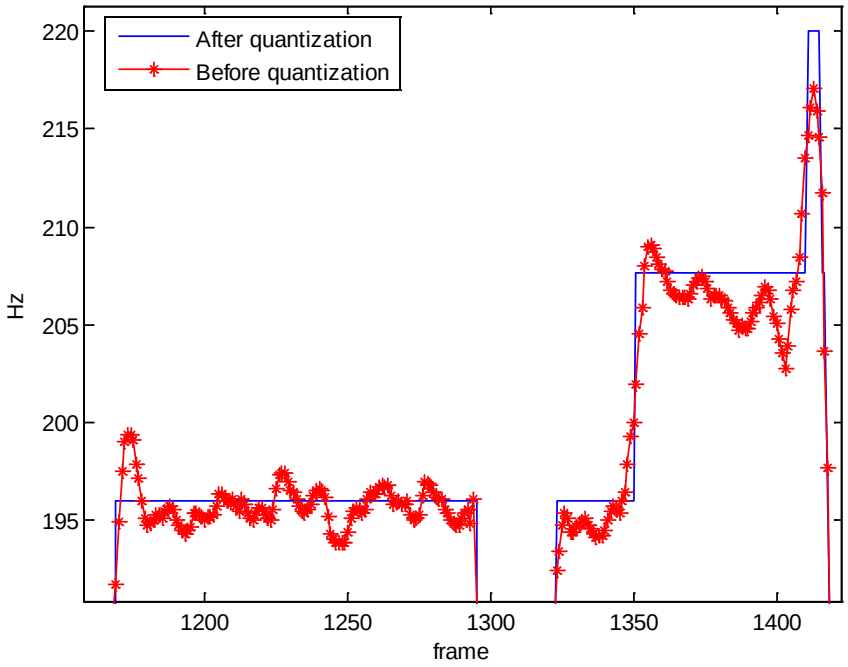

Fig. 3. Example result of quantization

3.5 Feature image extraction

Fig. 4 shows the process of extracting a feature image. An input music clip is first divided into a frame of 4096 sampling points. Because the input music clip has a sampling frequency of 44100 Hz, each frame is 92.9 ms. From a frame, fast Fourier transformation (FFT) is performed, and the energies of the 48 bands are computed. Thus, a frame is expressed as a 48-dimensional feature vector. The proposed algorithm is designed to output every 10 ms, thus a frame shift is set to 10 ms. The band separation is shown in **Table 1**. The feature vectors are collected in consecutive frames. Because we designed the CNN with 48 × 24 input images, one feature image is obtained by collecting 24 consecutive frames. To determine a melodic note of a frame, both the previous and the following frames are required. Thus, our algorithm is designed to obtain a melody note from the 12 previous frames, the present frame, and 11 following frames. **Fig. 5** shows an example of a feature image.

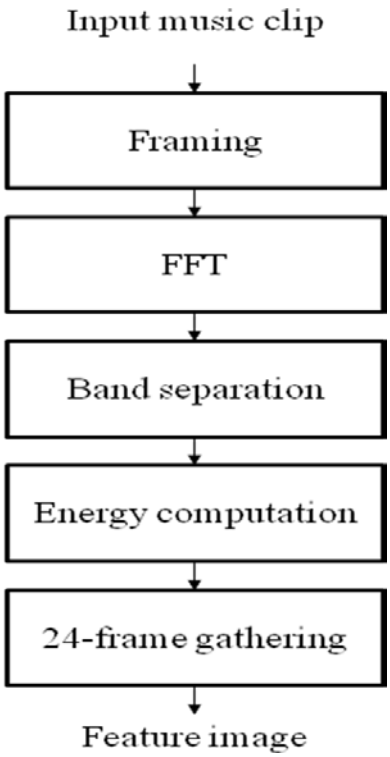

Fig. 4. Process of extracting a feature image

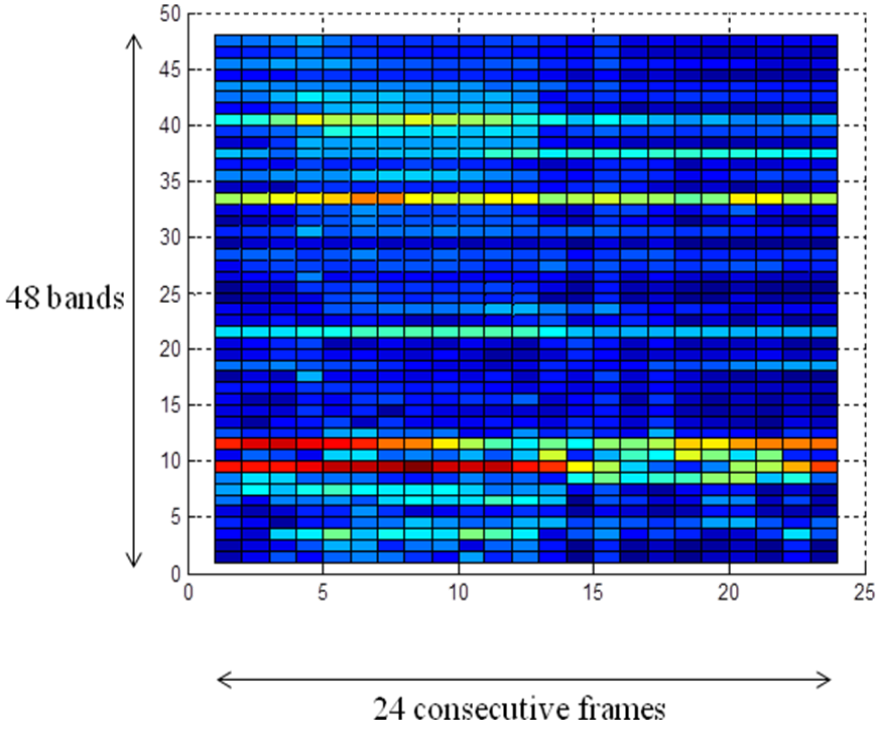

Fig. 5. Example of a feature image

4. Experiments

4.1 Experimental setup

We used the ADC2004, MIREX05, and MIREX09 datasets. The datasets contain single-channel pulse code modulation (PCM) data with a 44100 Hz sampling rate and 16 bit quantization. The datasets have a manually annotated reference frequency of every 10 ms. Specifications for the three datasets are listed in **Table 2**. The table shows that the MIREX09 dataset has three levels of noise mixed with the source. However, in our experiments, only the dataset with a signal-to-accompaniment ratio of 0 dB was used.

Because we used three datasets, six experiments were conducted: two experiments using the ADC2004 dataset data when the MIREX05 or MIREX09 datasets were used for training; two experiments using the MIREX05 dataset when the ADC2004 or MIREX09 datasets were used for training; and finally two experiments using the MIREX09 dataset when the ADC2004 or MIREX05 datasets were used for training.

To implement the CNN, the DeepLearnToolbox was used in our experiments [28]. It is a MATLAB/Octave toolbox for deep learning.

Table 2. The three datasets

Dataset	Number of files	Total length	Genre	Properties
ADC2004	20	6:01	Pop, Opera, Saxophone, Piano	12 with vocals, 8 without vocals
MIREX05	25	11:15	Rock, R&B, Pop, Jazz, Solo classical piano	16 with vocals, 9 without vocals
MIREX09	374	2:44:09	Chinese song	Karaoke recordings Mixed at three different levels of signal-to-accompaniment ratio: 0 dB, –5 dB, and 5 dB All songs have vocals

4.2 Experimental results

In this subsection, the experimental results are presented. To show the melody extraction performance, the evaluation procedures used in MIREX were also used [33]. The evaluations are divided into two parts: voicing detection (i.e., determining whether a particular time frame

contains a "melody pitch") and pitch detection (determining the most likely melody pitch for each time frame). For voicing detection, voicing recall rate and voicing false-alarm rate were computed. Voicing recall is the probability that a frame truly voiced is labeled as voiced, and voicing false-alarm rate is the probability that a frame not actually voiced is falsely labeled as voiced. For pitch detection, raw pitch accuracy and raw chroma accuracy were computed. Raw pitch accuracy is the probability of a correct pitch value (to within ±1/4 tone) being computed, given that the frame is indeed pitched, and raw chroma accuracy is the probability that the chroma (i.e., the note name) is correctly computed over the voiced frames. Overall accuracy gives the proportion of frames that were correctly labeled for both pitch and voicing. A detailed explanation can be found in [33].

In **Tables 3**, **4**, and **5**, the melody extraction performance of the proposed algorithm is compared with the performance of the algorithms submitted in MIREX 2015 [34]. All values were determined after comparing the ground truth and melody extraction results, which are represented in a frequency domain.

Table 3. Melody extraction results for the ADC2004 dataset

Algorithm		Overall accuracy	Raw pitch accuracy	Raw chroma accuracy	Voicing recall rate	Voicing false-alarm rate
Proposed (training with MIREX05)		0.6210	0.6508	0.6933	0.9324	0.5478
Proposed (training with MIREX09)		0.5704	0.5325	0.5521	0.6747	0.2911
Comparison: Algorithms submitted in MIREX 2015	BG1	0.6930	0.7793	0.8239	0.8220	0.2907
	FYJ1	0.6007	0.6744	0.7328	0.6938	0.1586
	FYJ2	0.5617	0.6854	0.7427	0.6008	0.1252
	FYJ3	0.5561	0.6612	0.7286	0.5930	0.1245
	FYJ4	0.6169	0.7084	0.7645	0.6914	0.1266
	IIY1	0.5843	0.6550	0.7298	0.8272	0.4615
	IIY2	0.6348	0.7086	0.7379	0.8465	0.4784
	ZCY1	0.6062	0.6814	0.7341	0.9544	0.8032
	ZCY2	0.6024	0.6778	0.7305	0.9542	0.8090

Table 4. Melody extraction results for MIREX05 dataset

Algorithm		Overall accuracy	Raw pitch accuracy	Raw chroma accuracy	Voicing recall rate	Voicing false-alarm rate
Proposed (training with ADC2004)		0.6441	0.6091	0.6512	0.8517	0.2248
Proposed (training with MIREX 09)		0.5135	0.4225	0.4417	0.5740	0.1925
Comparison: Algorithms submitted in MIREX 2015	BG1	0.6274	0.7036	0.7604	0.8521	0.5381
	FYJ1	0.5852	0.6326	0.7035	0.7053	0.1981
	FYJ2	0.5436	0.6425	0.6941	0.6195	0.2138
	FYJ3	0.5441	0.6343	0.6752	0.6040	0.2020
	FYJ4	0.5619	0.6319	0.7108	0.6959	0.2185
	IIY1	0.6074	0.6858	0.7696	0.9186	0.5687
	IIY2	0.6549	0.7305	0.7799	0.9342	0.5671
	ZCY1	0.4563	0.5113	0.6271	0.9075	0.8111
	ZCY2	0.4563	0.5113	0.6273	0.9078	0.8115

Table 5. Melody extraction results for MIREX09 dataset

Algorithm		Overall accuracy	Raw pitch accuracy	Raw chroma accuracy	Voicing recall rate	Voicing false-alarm rate
Proposed (training with ADC2004)		0.4771	0.4805	0.5153	0.8057	0.5218
Proposed (training with MIREX 05)		0.4505	0.5233	0.5739	0.9091	0.6781
Comparison: Algorithms submitted in MIREX 2015	BG1	0.5397	0.6234	0.6990	0.8189	0.5599
	FYJ1	0.7613	0.8253	0.8433	0.7090	0.0496
	FYJ2	0.7467	0.8119	0.8352	0.6832	0.0563
	FYJ3	0.7442	0.8094	0.8327	0.6756	0.0523
	FYJ4	0.7622	0.8120	0.8378	0.7127	0.0586

	IIY1	0. 6627	0.7890	0.8350	0.9510	0.5234
	IIY2	0.6807	0.8153	0.8277	0.9562	0.5357
	ZCY1	0.4623	0.6472	0.7292	0.9454	0.8707
	ZCY2	0.4618	0.6472	0.7293	0.9455	0.8724

4.3 Consideration

As shown in **Tables 3**, **4**, and **5**, the performance of the proposed scheme with ADC2004 when trained with MIREX05 and that with MIREX05 when trained with ADC2004 was similar to that of the state-of-the-art algorithms presented in MIREX 2015. However, the results related to MIREX09 were not similar. We think this is because of the mismatch between datasets. In fact, the MIREX09 dataset had unusual music data. In the dataset, 374 Karaoke recordings of Chinese songs are present, and the number of singers is limited. Thus, training may have been biased towards the voices of those limited number of singers. Moreover, too many frames in the MIREX09 dataset had a "0" label**.** As the singers are not professionals, they may not have been able to sing well; thus, there are a number of unvoiced frames in the dataset. This leads to the test results with many "0" labels when trained with the MIREX09 dataset, which explains the poor performance. In **Fig. 6**, histograms of three datasets are shown. As shown, the label distribution in the MIREX09 dataset is somewhat different from others: too many "0" labels are present, whereas labels of higher notes do not exist. For this reasons, MIREX09 is not adequate for training even though the dataset is extensive.

From the results shown in **Tables 3**, **4**, and **5**, we could confirm that melody extraction performance is dependent on the training dataset. Thus, we should ensure that an adequate training set is used to enhance performance. Applying the existing datasets such as ADC2004, MIREX05, and MIREX09 is extremely limited and can lead to biased outputs. The ADC2004 and MIREX05 datasets are very small, and the MIREX09 dataset is not appropriate for use in training.

In this study, a simple and novel algorithm to extract melody was proposed, and based on the experimental results, we could locate additional modules needed to compensate for the weak points of the algorithm. Raw pitch and chroma accuracy of the proposed algorithm were much lower than those in other algorithms that had similar overall accuracies. This means that some pre-processing modules that emphasize melodic lines are necessary. Voicing detection in the proposed algorithm is extremely dependent on the training set. In **Table 3**, the results of cases in which the engine was trained with the MIREX09 dataset and in which it was trained with the MIREX05 dataset are very different. If we can obtain a proper training set for detecting only voicing, then the development of a cascade structure, which has a module for determining voicing frames and a module for determining pitch, is preferable. This structure can help to enhance overall performance.

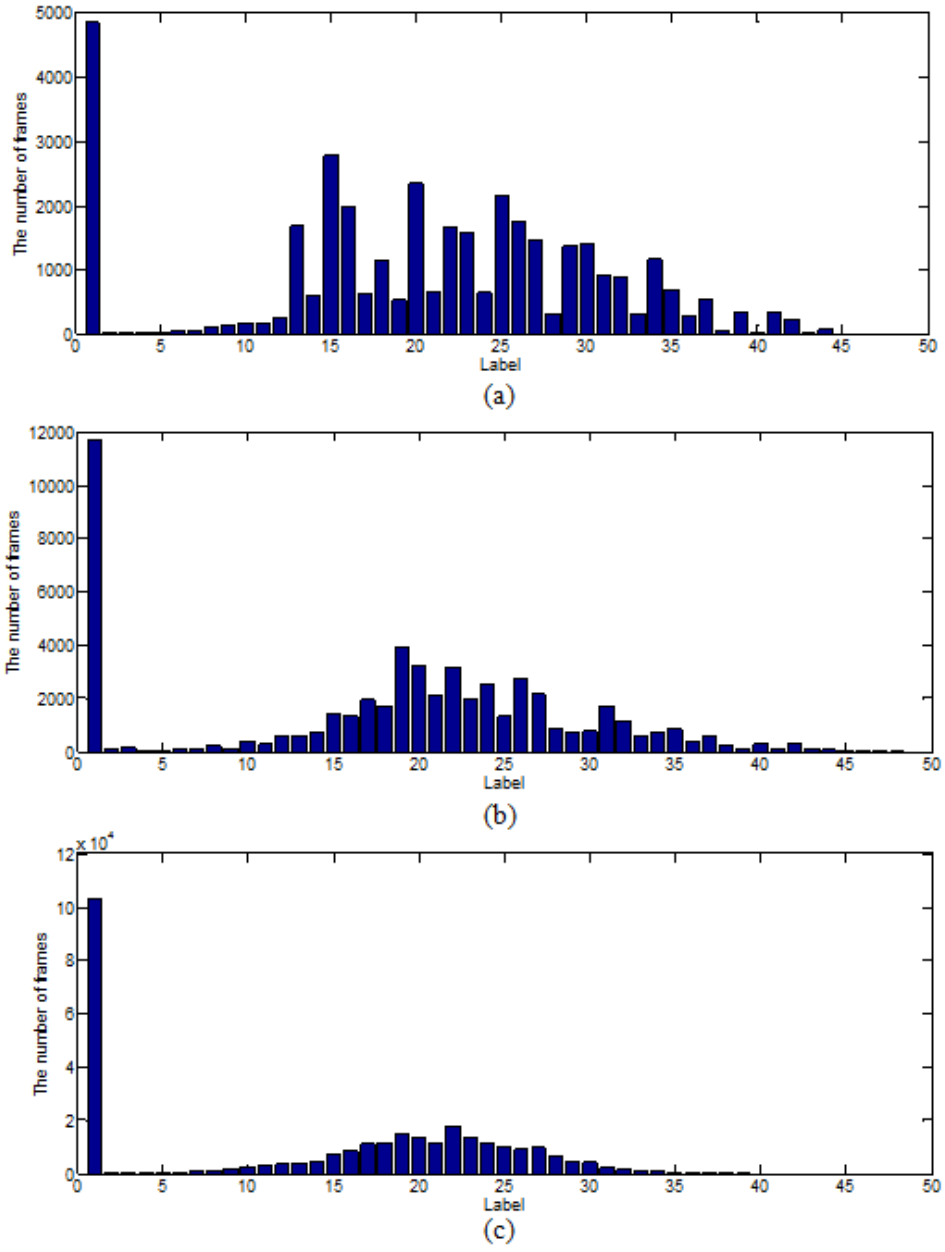

Fig. 6. Histograms of three datasets: (a) ADC2004, (b) MIREX05, and (c)MIREX09

5. Conclusion

In this study, we developed a melody extraction algorithm based on CNN. With the assumption of large training data, we built a melody extraction algorithm based on a train–test framework. The process of the proposed melody extraction algorithm was simplified as two steps: extraction of energy-based feature images and the use of a CNN to extract melodies. A

feature image has the form of a 48 × 24 matrix derived from 24 consecutive frames. The proposed algorithm has a very simple structure with a CNN. This simple structure causes outputs to be biased to the training dataset and this was verified in our experiments. Promising results were obtained with the train-test framework using a CNN. The algorithm proposed in this paper is just a basic structure using a CNN, and there are many ways to enhance this structure. The most important point is the use of a proper training set. Additional parts, such as pre-processing modules emphasizing melodic lines and the use of a cascade structure can be helpful. As the band energy is very simple, it is also a good candidate for enhancement to try features from other melody extraction algorithms. The feature extraction method can also be replaced with others: Hermes' sub-harmonic summation [35], a method based on a log spectrum [36], multi-resolution FFT [37], and so on. Applications of these algorithms for further improvements remain as future work.

References

[1] J. S. Downie, "Music information retrieval," *Annual Review of Information Science and Technology*, 37:295-340, 2003. Article (CrossRef Link)

[2] R. Typke, F. Wiering and R. Veltkamp, "A survey of music information retrieval systems," in *Proc. of ISMIR*, pp. 153-160, 2005. Article (CrossRef Link)

[3] D. Jang, C.-J. Song, S. Shin, S.-J. Park, S.-J. Jang and S.-P. Lee, "Implementation of a matching engine for a practical query-by-singing/humming system," in *Proc. of ISSPIT*, pp. 258-263, 2011. Article (CrossRef Link)

[4] J. S. R. Jang and H. R. Lee, "A general framework of progressive filtering and its application to query by singing/humming," *IEEE Trans. on Audio, Speech, and language Processing*, vol. 16, no. 2, pp 350-358, Feb., 2008. Article (CrossRef Link)

[5] S. W. Hainsworth and M. D. Macleod, "Particle filtering applied to musical tempo tracking," *EURASIP J. Applied Signal Processing*, vol. 15, pp. 2385-2395, 2004. Article (CrossRef Link)

[6] D. P. W. Ellis and G. E. Poliner, "Identifying cover songs with chroma features and dynamic programming beat tracking," in *Proc. of Int. Conf Acoustic, Speech and Signal Processing*, Honolulu, HI, 2007. Article (CrossRef Link)

[7] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. Speech Audio Process*, vol. 10, no. 5, pp. 293-302, 2002. Article (CrossRef Link)

[8] D. Jang, M. Jin and C. D. Yoo, "Music genre classification using novel features and a weighted voting method," in *Proc. of ICME*, 2008. Article (CrossRef Link)

[9] T. LH. Li, A. B. Chan, and A. HW. Chun, "Automatic Musical Pattern Feature Extraction Using Convolutional Neural Network," in *Proc of IMECS*, 2010. Article (CrossRef Link)

[10] X. Hu and J. S. Downie, "Improving mood classification in music digital libraries by combining lyrics and audio," in *Proc of the 10th annual joint conference on Digital libraries*, pp159-168, 2010. Article (CrossRef Link)

[11] J. H. Kim, S. Lee, S. M. Kim and W. Y. Yoo, "Music mood classification model based on Arousal-Valence values," in *Proc of ICACT*, pp 292-295, 2011. Article (CrossRef Link)

[12] D. Jang, C. D. Yoo, S. Lee, S. Kim and T. Kalker, "Pairwise Boosted Audio Fingerprint," *IEEE Trans. Information Forensics and Security*, vol. 4, no. 4, pp. 995-1004, Dec. 2009. Article (CrossRef Link)

[13] J. Haitsma and T. Kalker, "A highly robust audio fingerprinting system," ISMIR 2002. Article (CrossRef Link)

[14] S. Durand, J. P. Bello, B. David, and G. Richard, "Feature Adapted Convolutional neural Networks for Downbeat Tracking," in *Proc. of ICASSP*, 2016. Article (CrossRef Link)

[15] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," in *Proc of ISMIR*, 2016. Article (CrossRef Link)

[16] S. Jo and C. D. Yoo, "Melody extraction from polyphonic audio based on particle filter," in *Proc of ISMIR*, pp. 357-362, 2010. Article (CrossRef Link)

[17] D. P. W. Ellis and G. E. Poliner, "Classification-based melody transcription," *Machine Learning*, Vol. 65, pp. 439-456, 2006. Article (CrossRef Link)

[18] 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*, 2014 Article (CrossRef Link)

[19] K. Dressler, "An auditory streaming approach for melody extraction from polyphonic music," in *Proc. of 12th ISMIR*, Miami, FL, pp. 19–24, Oct. 2011. Article (CrossRef Link)

[20] V. Rao and P. Rao, "Vocal melody extraction in the presence of pitched accompaniment in polyphonic music," *IEEE Trans. Audio, Speech, Lang. Processing*, vol. 18, no. 8, pp. 2145–2154, Nov. 2010. Article (CrossRef Link)

[21] S. Jo, S. Joo and C. D. Yoo, "Melody pitch estimation based on range estimation and candidate extraction using harmonic structure model," in *Proc. of InterSpeech*, Makuhari, Japan, Sept. 2010, pp. 2902–2905. Article (CrossRef Link)

[22] V. Arora and L. Behera, "On-line melody extraction from polyphonic audio using harmonic cluster tracking," *IEEE Trans. Audio, Speech, Lang. Processing*, vol. 21, no. 3, pp. 520–530, Mar. 2013. Article (CrossRef Link)

[23] C. Hsu and J. S. R. Jang, "Singing pitch extraction by voice vibrato/tremolo estimation and instrument partial deletion," in *Proc. of 11th ISMIR*, Utrecht, The Netherlands, Aug. 2010, pp. 525–530. http://ismir2010.ismir.net/proceedings/ismir2010-89.pdf

[24] T.-C. Yeh, M.-J. Wu, J.-S. Jang, W.-L. Chang and I.-B. Liao, "A hybrid approach to singing pitch extraction based on trend estimation and hidden Markov models," in *Proc. of IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Kyoto, Japan, pp. 457–460, Mar. 2012. Article (CrossRef Link)

[25] S. Kum, C. Oh, and J. Nam, "Melody Extraction on Vocal Segments using Multi-Column Deep Neural Networks," in *Proc. of ISMIR*, 2016 Article (CrossRef Link)

[26] E. J. Humphrey, J. P. Bello, and Y. LeCun, "Moving Beyond Feature Design: Deep Architectures and Automatic Feature Learning in Music Informatics," in *Proc of ISMIR*, 2012. Article (CrossRef Link)

[27] Music Information Retrieval Evaluation eXchange [Online], Available: http://www.music-ir.org/mirex/wiki/MIREX_HOME

[28] R. B. Palm, "Prediction as a candidate for learning deep hierarchical models of data", Technical University of Denmark, 2012. Article (CrossRef Link)

[29] A. Krizhevsky, I Sutckever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. of NIPS*, 2012. Article (CrossRef Link)

[30] D.C. Ciresan, U Meier and L. M. Gambardella, " Convolutional neural network committees for handwritten character classification," in *Proc. of International Conference on Document Analysis and Recognition*, pp. 1250–1254, 2011Article (CrossRef Link)

[31] C. Zhang and Z. Zhang. "Improving multiview face detection with multi-task deep convolutional neural networks," in *Proc. of Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, pp. 1036–1041, 2014. Article (CrossRef Link)

[32] J. Zbontar and Y LeCun, "Computing the stereo matching cost with a convolutional neural network, " Proceeding of CVPR 2015. Article (CrossRef Link)

[33] 2016: Audio Melody Extraction [online] Available: http://www.music-ir.org/mirex/wiki/2016:Audio_Melody_Extraction

[34] 2015: MIREX2015 Results [online] Available: http://www.music-ir.org/mirex/wiki/2015:MIREX2015_Results

[35] D. Hermes, "Measurement of pitch by subharmonic summation," *Journal of Acoustic of Society of America*, vol.83, pp.257-264,1988. Article (CrossRef Link)

[36] V. Arora and L. Behera, "On-line melody extraction from polyphonic audio using harmonic cluster tracking," *IEEE Trans. on Audio Speech and Language Processing*, vol. 21, no. 3, pp. 520 –530, Mar. 2013.Article (CrossRef Link)

[37] T.-C. Yeh, M.-J. Wu, J.-S. Jang, W.-L. Chang, and I.-B. Liao, "A hybrid approach to singing pitch extraction based on trend estimation and hidden Markov models," in *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, pp. 457–460, Mar. 2012. Article (CrossRef Link)

Jongseol Lee received the B.S. and M.S. degrees in Information & Communication engineering from Chungbuk National University, Korea, in 1996 and 2001. He is currently pursuing a Ph.D. degree in Computer Science and Engineering at Konkuk University. Since 2011, he has worked as a senior research engineer at the Smart Media Research Center of Korea Electronics Technology Institute (KETI), Seoul, Korea. His research interests include smart media, audio processing, music analysis, recommendation and retrieval.

Dalwon Jang received the B.S., M.S., and Ph.D degrees from Korea Advanced Institute of Science and Technology, in 2002, 2003, and 2010, respectively, all in electrical engineering. Since 2010, he has worked at the Smart Media Research Center of Korea Electronics Technology Institute (KETI), Seoul, Korea. His research interests include content identification, music information retrieval, multimedia analysis, and machine learning.

Kyoungro Yoon received a B.S. degree in Computer and Electronic Engineering from Yonsei University, Seoul, Korea in 1987, an M.S.E. degree in Electrical Engineering/Systems from the University of Michigan, Ann Arbor in 1989, and a Ph.D. in Computer and information Science from Syracuse University in 1999. He was a principal researcher and a group leader in the Mobile Multimedia Research Lab, LG Electronics Institute of Technology from 1999 to 2003. He joined the school of Computer Science and Engineering in 2003 as an assistant professor and became a full professor in 2012. He is with the department of Smart ICT Convergence, since 2017. He served as a co-chair of Ad Hoc Group on User Preferences and the chair of Ad Hoc Group on MPEG Query Format and Ad Hoc Group on MPEG-V of ISO/IEC JTC1 SC29 WG11 (a.k.a. MPEG). He also served as the chair of the Metadata Subgroup and JPSearch Ad Hoc Group of ISO/IEC JTC1 SC29 WG1 (a.k.a. JPEG). He is an editor of various international standards such as ISO IS 15938-12, 23005-2, 23005-5, 23005-6, 24800-3, 24800-5, and 24800-6. His main research interests include smart media system, image processing, multimedia information and metadata processing.