 doi:10.20944/preprints201711.0027.v1

*Article*

# Classification-Based Singing Melody Extraction Using Deep Convolutional Neural Networks

**Sangeun Kum[1] and Juhan Nam[1,*]**

[1] Music and Audio Computing Lab, Korea Advanced Institute of Science and Technology, 291 Daehak-ro Yuseong-gu Daejeon, Republic of Korea; keums@kaist.ac.kr

* Correspondence: juhannam@kaist.ac.kr; Tel.: +82-42-350-2926

**Abstract:** Singing melody extraction is the task that identifies the melody pitch contour of singing voice from polyphonic music. Most of the traditional melody extraction algorithms are based on calculating salient pitch candidates or separating the melody source from the mixture. Recently, classification-based approach based on deep learning has drawn much attentions. In this paper, we present a classification-based singing melody extraction model using deep convolutional neural networks. The proposed model consists of a singing pitch extractor (SPE) and a singing voice activity detector (SVAD). The SPE is trained to predict a high-resolution pitch label of singing voice from a short segment of spectrogram. This allows the model to predict highly continuous curves. The melody contour is smoothed further by post-processing the output of the melody extractor. The SVAD is trained to determine if a long segment of mel-spectrogram contains a singing voice. This often produces voice false alarm errors around the boundary of singing segments. We reduced them by exploiting the output of the SPE. Finally, we evaluate the proposed melody extraction model on several public datasets. The results show that the proposed model is comparable to state-of-the-art algorithms.

**Keywords:** convolution neural networks; melody extraction; singing voice activity detection; voice false alarm detection

## 1. Introduction

Melody extraction is the task of estimating the fundamental frequency that corresponds to the melodic line of a polyphonic music. Since melody is the essence of music from which listeners can identify the piece, melody extraction has been applied to various music information retrieval tasks such as query-by-humming [1] and cover song identification [2]. In popular music, melody is usually performed by singers and so the melody extraction task is often recast into detecting the presence of singing voices and estimating the dominant voice pitch when background music is accompanied. The expressive nature of the singing melody has been utilized for explaining characteristics of different music genres [3] or singers [4] as well. Furthermore, the continuous pitch curves have been incorporated in source separation algorithms to take vocal and background music apart [5].

A number of melody extraction algorithms, where some of them are particularly for singing voices, have been proposed so far. They can be broadly classified into three categories according to the approach type: salience-based, source separation-based, and classification-based ones [6]. While the majority of previous work are associated with the first two approaches, the data-driven approach based on classification, which predicts a finite set of pitch labels from audio features, have been rarely explored. An early work used a support vector machine classifier to predict a pitch label from spectrogram [7]. Since then, it had been no attempt until Bittner et al. proposed a random forest classifier that predicts pitch contours from pitch salience features [8].

The lack of the classification-based approach can be attributed to the following reasons. First, melodic pitch is a physically measurable value as opposed to abstract labels defined in high-level tasks such as genre or mood classification. Thus, it is more intuitive to directly leverage time-frequency

 doi:10.20944/preprints201711.0027.v1



representations where the patterns for pitch estimation are observable, as in the saliency-based or source separation-based approaches. Second, in the classification-based approach, the melodic pitch is supposed to be quantized to a certain resolution (e.g. semitone in [7]). While this discrete pitch may be useful for some applications that require a MIDI-level pitch notation, it loses detailed information about singing styles such as vibrato or note-to-note transition patterns. Third, the classification-based approach typically requires a sufficient amount of labeled data to achieve good performance. Manual extraction of melodic pitch in a frame-level is a highly tedious labor, particularly for mixed tracks. This has hindered the availability of labeled datasets.

In the recent past, however, there have been important changes that have encouraged the classification-based approach. First, multi-track audio recording data including singing voice as a separate track have been more available [9–11]. With the multi-track datasets, the melody labels can be obtained more easily by applying a monophonic pitch detector to the isolated vocal track. Second, deep learning, the powerful data-driven learning algorithm based on neural networks, has emerged and tremendously advanced, achieving a remarkable series of state-of-the-art results in numerous tasks. An indispensable element in the success of deep learning is the availability of large-scale labeled datasets.

Leveraging the datasets and recent advances in deep learning, several classification-based methods using neural networks have been recently attempted. Rigaud and Radenen proposed to use two types of neural networks. One is for detecting singing voice activity built with 3-Bidirectional Long Short-Term Memory (BLSTM) layers following [12]. The other is for extracting singing pitch composed of 2-hidden fully connected layers and a softmax layer that discriminates up to an eighth of semi-tone [13]. Comparing the state-of-the-art salience-based system, *Melodia* [14], they show significantly improved results. Kum and Nam proposed multi-column deep neural networks (MC-DNN) where each column network is trained to predict a pitch label with a different pitch resolution and the outputs are combined [15]. The results showed that the ensemble method achieves better performance than a single model and also it returns a high pitch resolution. Park and Yoo presented a LSTM-based melody classification algorithm where they added harmonic sum loss to the objective function to incorporate the harmonic structure in melodic tone [16]. They showed that the harmonic sum loss makes the model more robust to octave mismatch and interference from background music.

In this work, we propose a singing melody extraction model using deep convolutional neural networks (DCNN). While DCNN-based models have been shown to achieve state-of-the-art results in many music information retrieval (MIR) tasks including singing voice detection [17], polyphonic piano transcription [18], chord recognition [19] and music-auto tagging [20], to the best of our knowledge, they have been not applied to singing melody extraction yet. We use two DCNNs, one for singing pitch extractor (SPE) and the other for singing voice activity detector (SVAD). In particular, we investigate the importance of pitch resolution in the SPE. Also, we suggest to use the output of pitch prediction in the SPE as a mean to suppress voice false alarm errors from the result of the SVAD. Using several public datasets, we show that the proposed method significantly outperforms our previous work and the overall results are comparable to state-of-the-arts.

## 2. Proposed Methods

The proposed melody extraction method is illustrated in Figure 1. It is composed by two main parts. The SPE extracts melody features and predicts its pitch from a short segment of spectrograms (11 frames). Then, the output is temporally smoothed by a hidden Markov model (HMM) based post-processing. The SVAD serves to distinguish singing voice frames from a long segment of mel-spectrograms (115 frames) and removes melodic contours of the non-voice segments. In addition, the voice false alarm detector reduces the false positive from the SVAD results by exploiting the output of SPE.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



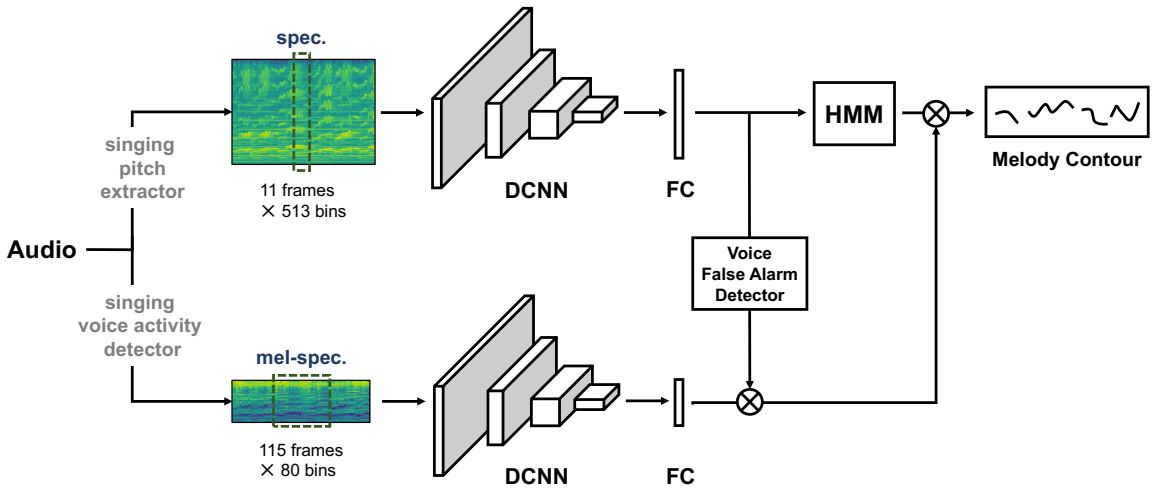


**Figure 1.** The diagram of our architecture for melody extraction including singing voice activity detector.

### *2.1. DCNN Model Configuration for the Singing Pitch Extractor*

The architecture of SPE are summarized in Table 1. The SPE is configured with four convolutional blocks and one fully connected layer. Each block contains two convolutional layers and two pooling layers except the last one. The convolution filters have a filter size of 3×3 and the number of filters in the convolutional blocks gradually increases as 64, 128, 256 and up to 512. Then, average-pooling is applied to the time axis and max-pooling is to the frequency axis. The intuition behind this setting is that singing pitch is typically continuous and so temporal smoothing by the average maintain the pitch information better than max-pooling. Experimentally, we confirmed that this actually worked better than using max-pooling on both axes. We apply batch normalization on each convolutional layer and use the Leaky ReLU as an activation function for the non-linearity. We include dropout to the end of each block and use softmax activation function for the output layer. The pitch labels cover from D2 (73.416 Hz) to B5 (987.77 Hz). We quantized the pitch labels on MIDI scale but with high resolutions. $R_t$ denotes the resolution in $1/t$ semitone unit. For example, $R_1$ indicates pitch resolution in semitone unit. $R_2$, $R_4$, $R_8$, $R_{16}$ and $R_{32}$ indicate progressively higher resolutions than semitone by a factor of 2.

We used spectrogram as input for the SPE. We first resampled audio clips to 8 kHz and merged stereo channels into mono. We then computed spectrogram with Hann window of 1024 samples and hop size of 80 samples. We compressed the magnitude of the spectrogram in a log scale and used 513 bins from 0 Hz to 4000 Hz. As in the previous work [15], we took multiple frames of spectrogram as input to capture contextual information from neighboring frames and use the pitch label at the center position of the context window. We also experimented with different sizes of input frames and obtained the best results at 11 frames in SPE as well. Thus, we fix the input size to 11 frames for all experiments.

### *2.2. DCNN Model Configuration for the Singing Voice Activity Detector*

The architecture of SVAD is summarized in Table 2. The SVAD is configured with four convolutional blocks and 1×1 convolutional layer. Each convolution block contains two 3×3 convolutional layers followed by batch normalization. The number of channels in the blocks gradually increases as 64, 128, 256 and up to 512. 3×3 max-pooling layers with 2×2 stride are used at the end of each convolutional block. At the final stage of the DCNN model, we used 1×1 convolution and

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



**Table 1.** Configuration of DCNN for the singing pitch extractor

| input: SPEC. 11(frames)× 513(bins) | | | | |
|---|---|---|---|---|
| | $R_8$ | $R_{16}$ | $R_{32}$ | output |
| block1 | 64 conv1 (3×3) | | | 64×11×513 |
| | 64 conv2 (3×3) | | | 64×11×513 |
| | average-pool (2×1) | | | 64×5×513 |
| | max-pool (1×3) | | | 64×5×171 |
| block2 | 128 conv3 (3×3) | | | 128×5×171 |
| | 128 conv4 (3×3) | | | 128×5×171 |
| | average-pool (2×1) | | | 128×2×171 |
| | max-pool (1×3) | | | 128×2×57 |
| block3 | 256 conv5 (3×3) | | | 256×2×57 |
| | 256 conv6 (3×3) | | | 256×2×57 |
| | average-pool (2×1) | | | 256×1×57 |
| | max-pool (1×3) | | | 256×1×14 |
| block4 | 512 conv7 (3×3) | | | 512×1×14 |
| | 256 conv8 (3×3) | | | 256×1×14 |
| | max-pool (1×7) | | | 256×1×2 |
| FC | 512 | 1024 | 2048 | - |
| softmax | 361 | 721 | 1441 | - |

global average pooling instead of fully connected layer. The architecture was inspired by the *Network In Network* model [21], which has the advantage of avoiding the problem of overfitting and greatly reducing the amount of computation without degrading performance. After the DCNN predicts the output, we used a median filter of 110ms as the final step to perform temporal smoothing [17].

The SVAD takes 115 frames of mel-spectrogram as input to capture contextual information over a long time span following [17]. We resampled audio signals to 16kHz and merged stereo channels to mono. We extracted mel-spectrogram with 80 triangular filters between 0 and 8 kHz, a frame length of 1024, hop size of 160 samples. We compressed the magnitude by a log scale.

### *2.3. Voice False Alarm Detection for the SVAD*

The SVAD takes a long segment (115 frames or 1.15 seconds) as input. We observed that the setting often produces false positive errors around the boundary of melody contours or short pauses between two melody contours. It is because long input frames taken from the boundary regions contain singing voice in part and so the SVAD is likely to predict the existence even if there is no voice at the center position. Therefore, we need to have additional means to minimize the false positive errors by taking a smaller size of input frames. For this purpose, a method of reducing the errors by detecting sub-semitone fluctuations has been previously attempted [22]. In this work, we propose a novel method that utilizes the output of the SPE.

We empirically found that, when the SPE takes non-voiced frames as input and predicts the pitch, the output was not dominant at a particular class and tends to have a low probability for each class. This is probably because the model was trained only with voice frames and so the model cannot make a prediction with high confidence for the unseen input. By exploiting the observation, we add a voice false alarm detector (VFAD) based on the SPE as follows:

$$S_{VFAD}(n) = \begin{cases} 1 & \text{if } \arg\max_n y_{SPE}(n) > \theta \\ 0 & \text{if } \arg\max_n y_{SPE}(n) < \theta \end{cases}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



**Table 2.** configuration of DCNN for Singing Voice Activity Detector

| input : Mel-Spec 115 (frames) × 80 (bins) | | | |
|---|---|---|---|
| | stride | architecture | output |
| block1 | 1 | 64 conv1 (3×3) | 64×115×80 |
| | 1 | 64 conv2 (3×3) | 64×115×80 |
| | 2 | maxpool (3×3) | 64×57×39 |
| block2 | 1 | 128 conv3 (3×3) | 128×57×39 |
| | 1 | 128 conv4 (3×3) | 128×57×39 |
| | 2 | maxpool (3×3) | 128×28×19 |
| block3 | 1 | 256 conv5 (3×3) | 256×28×19 |
| | 1 | 256 conv6 (3×3) | 256×28×19 |
| | 2 | maxpool (3×3) | 256×13×9 |
| block4 | 1 | 256 conv7 (3×3) | 256×13×9 |
| | 1 | 512 conv8 (3×3) | 512×13×9 |
| | 2 | maxpool (3×3) | 512×6×4 |
| 1×1 conv block | 1 | 128 conv (1×1) | 128×6×4 |
| | 1 | 2 conv (1×1) | 2×6×4 |
| | - | global average-pool | 2 |
| Total #params : 2,983,746 | | | |

where $y_{SPE}(n)$ is the softmax output of SPE at $n$ frame and $\theta$ is a threshold. We obtain the final result of singing voice activity, $S(n)$ by incorporating the VFAD into the SVAD:

$$S(n) = S_{SVAD}(n) \cdot S_{VFAD}(n) \tag{1}$$

where $S_{SVAD}(n)$ is the result of SVAD that returns one for voiced frames or zero for unvoiced frames.

### *2.4. Temporal Smoothing by HMM*

After predicting the output in the SPE, we conduct temporal smoothing for the frame-wise pitch prediction. The procedure was basically borrowed from the Viterbi decoding based on HMM in [7]. The HMM state corresponds to each of the melody pitch values and the prior probabilities and the transition matrix are computed from the ground-truth of the training set. As posterior probabilities, the prediction from the combined output of SPE is used. To generate the prior and transition probabilities, we counted the number of occurrences and all pitch-to-pitch transition per pitch label, respectively. In addition, we normalized the transition matrix by replacing each element with the average of its corresponding diagonal. This alleviates the sparsity problem in the transition matrix obtained from a limited training set by assuming that all adjacent pitch transitions depend only on their interval rather that absolute pitch value.

However, even with the normalization, the diagonal components of the transition matrix are still dominant. Thus, when the pitch difference between consecutive melodies is small, the result of smoothing tends to keep the same pitch. This leads to the loss of detail changes in the pitch contours. To deal with the problem, we add more weights to off-diagonal elements by multiplying a penalty matrix to the transition matrix as follows:

$$P = e^{\lambda} D + I \tag{2}$$

$$\check{T} = PT \tag{3}$$

where $D$ is the off-diagonal matrix of the transition matrix $T$, whose diagonal elements are zeros. $I$ is identity matrix. By increasing the value of the off-diagonal component, it adjusts the sensitivity to small pitch changes during the smoothing process.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



## 3. Dataset

### 3.1. Training Datasets

We used the RWC and MedleyDB datasets to train the SPE. To train the SVAD model, we used the Jamendo dataset in addition to the two. We divided them into training and validation splits to tune the network parameters. To avoid overfitting and select the best performing model, we chose songs such that genre and gender are evenly distributed over the splits and also the songs of the same singer are not divided over the splits.

- RWC [23]: 80 Japanese popular songs and 20 American popular songs with singing voice melody annotations. We divided the dataset into two splits, 85 songs for training and the remaining 15 songs for validation. The total length of the dataset is 407 minutes.
- MedleyDB [10]: 122 songs with a variety of musical genres and 70 of them including vocals with melody annotations. Among them, we chose 60 songs that are dominated by vocal melody. We divided the dataset into two splits, 47 songs for training and the remaining 13 songs for validation. The total length of the dataset is about 200 minutes.
- Jamendo [24]: 93 songs designed for the evaluation of singing voice detection. The training, validation and test set splits are designated as 61, 16 and 16 songs, respectively. The total length of the dataset used for training is about 360 minutes.

We also augmented the three datasets to obtain more generalized models. Pitch shifting has proven to be an effective way to increase data and improve results for singing voice activity detection [17] and melody extraction [15] as well. To this end, instead of resampling that modifies the pitch and length of audio clips at the same time [25], we used a phase vocoder method that conducts pitch-shifting independent of time-stretching [26]. We augmented the training set by applying the pitch-shifting by $\pm 1, 2$ semitones, thereby increasing the data size by five times.

### 3.2. Test Datasets

To evaluate the proposed model, we use publicly available datasets: ADC2004, LabROSA, MIR1k and iKala. Synthesized sounds or instrument sounds (e.g. 'train13MIDI.wav' in LabROSA or 'midi1.wav' in the ADC04 dataset) were excluded from the training data so that both SPE and SVAD focus on singing voice in polyphonic music. Thus, we used only singing voice songs as test data among the whole datasets.

- LabROSA [1]: 13 excerpts that contain Rock, R&B, pop, and jazz songs, as well as audio generated from a MIDI file. We evaluated our algorithm using 9 songs out of a total of 13 songs.
- ADC2004 [1]: 20 excerpts of 20 seconds that contain pop, jazz and opera songs, as well as synthesized singing and audio from MIDI files. Jazz and MIDI songs were excluded from the evaluation.
- iKala [11]: 262 Chinese songs clips of 30 seconds performed by 6 professional singers.
- MIR-1k [9] : 1000 songs clips with the total duration of 133 minutes. 19 amateur singers (11 males and 8 females) participated in the recording.

### 3.3. Evaluation

We evaluated the proposed method in terms of five metrics, including overall accuracy (OA), raw pitch accuracy (RPA), raw chroma accuracy (RCA), voicing detection rate (VR) and voicing false alarm rate (VFA), as detailed in [6]. We compute them using *mir-eval*, a Python library designed for objective

[1] We obtained the LabROSA dataset from the website, http://labrosa.ee.columbia.edu/projects/melody/. This was used for part of the 2005 MIREX melody extraction task. In our previous work [15], we referred to it as MIREX05.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



evaluation in MIR tasks [27]. The evaluation consists of two main parts: voice detection determining whether a voice is included in a particular time frame (VR and VFA) and pitch detection determining the most accurate melody pitch for each time frame (RPA, RCA, and OA). We convert the pitch labels, which were quantized to MIDI scale, back to frequency scale (Hz) to compare them with the ground truth.

$$f = 2^{(m-69)/12} \cdot 440(Hz) \tag{4}$$

where $m$ is the estimated pitch label. The pitch of the frame is considered correct if the difference between the estimated pitch frequency and the ground-truth is within $\pm 50$ cents (0.5 semitone). In addition, we progressively reduced the pitch tolerance to $\pm 25$, $\pm 12.5$ cents. We report the results in order to show the performance under more strict conditions.

## 4. Experiments

Given the SPE and SVAD models and training data, we conducted several experiments to figure out the effect of different settings in the models. In the followings, we describe the experimental setup.

### *4.1. Training Details of DCNNs*

We randomly initialized the network parameters using He uniform initialization [28] and trained them with stochastic gradient descent with Nesterov momentum which was set to 0.9. We iterated it over all the training data up to 100 epochs. The initial learning rate was set to 0.02. To prevent overfitting, we applied a dropout ratio of 0.3 after all max-pooling layers. By means of early-stopping strategy, if the validation accuracy does not increase after 20 iterations, we reset the learning rate to 1/2 of the initial learning rate and repeated the training. We iterate this process five times. For fast computing, we ran the code using Keras [29], a deep learning library in Python, on a computer with two GPUs.

### *4.2. Pitch resolution and ensemble model*

Our first experiment is to figure out the maximum pitch resolution of the SPE. High pitch resolution allows the SPE to predict continuous pitch curves, mitigating the pitch quantization problem that the classification-based approach has intrinsically. In our previous work [15], we progressively increased the pitch resolution and observed that the performance saturates before $R_8$. With the DCNN-based model, we conduct the same experiment and find the pitch resolution that provides the best performance.

We also combine multiple neural networks with different pitch resolutions as we conducted in [15]. We denote SC-SPE$_r$ as a single-column DCNN with a pitch resolution $R_r$ and MC-SPE$_r$ as an ensemble model that combines SC-SPE$_r$, SC-SPE$_{r/2}$ and SC-SPE$_{r/4}$. We evaluted all the models on two test sets (ADC2004, LabROSA) and compared the accuracy.

### *4.3. HMM-based Postprocessing*

We conducted temporal smoothing of the pitch prediction using the Viterbi decoding. The prior probabilities and transition matrix were estimated from the ground-truth of the training set. To increase the value of the off-diagonal components, we set the $\lambda$ according to Equation 2. We used a set of $\lambda$ and empirically found that $\lambda$ of 1 yielded the best results.

### *4.4. Singing voice activity detector with VFAD*

As mentioned in Chapter 2.3, we use the VFAD to reduce false positive frames after the SVAD. If the maximum softmax output of SPE does not exceed a specific threshold $\theta$, it is assumed that the frame is not a singing voice. The threshold $\theta$ was set to a value between 0 and 0.05 to find the proper threshold. We used the songs from the ADC04 and LabROSA datasets. We evaluated the performance in terms of VR, precision, F1 score and VFA. We also compared the performance of the SVAD with

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



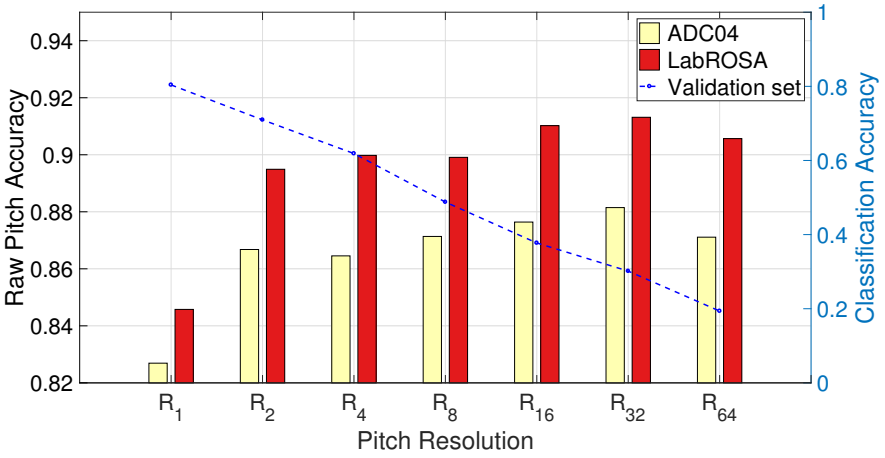


**Figure 2.** Raw pitch accuracies of test datasets (ADC04 and LabROSA) and classification accuracies of validation dataset according to the pitch resolution.

those from state-of-the-art algorithms. We reported the results on the Jamendo dataset as unseen test data. To evaluate the performance of the SVAD, we compute three common evaluation metrics: VR, precision, F1 score [30].

## 5. Results

### *5.1. DCNN Models of Melody Extraction*

Figure 2 shows the RCA on the two test datasets and the classification accuracy of the validation set with varying pitch resolution. We conducted the experiment by increasing the pitch resolution until the accuracy becomes saturated. The result shows that the higher the pitch resolution, the lower the classification accuracy. This is because it is not easy to predict the exact class corresponding to the reference pitch as the melody extractor has more classes to predict. On the other hand, the RPA on the test datasets tend to increase as the pitch resolution is higher. Compared to the DNN model which was saturated at $R_4$ [15], the DCNN model has the best results at $R_{32}$. This indicates that the DCNN is more capable of handling high pitch resolutions. However, we should note that higher resolutions require more network parameters.

Figure 3 shows that the MC-SPE models perform better than the SC-SPE models in general, validating that combining multiple models with different pitch resolutions is more effective [15]. However, the effect of using the multi-column models becomes less significant as the pitch resolution increases. This is clearly indicated by $\Delta$ RPA, the difference of RPA between the two models. For $R_{32}$ on the LabROSA, the SC-SPE model is even better the MC-SPE model, achieving the best accuracy. Thus, considering the ensemble model requires as many parameters as the model size, SC-SPE is seen to be a more practical choice in the DCNN-based approach.

### *5.2. HMM-based Post-processing*

Figure 4 show that both RPA and RCA increase by more than 1% on both datasets after the temporal smoothing. Comparing the difference between RPA and RCA, we can observe that the octave error decreases significantly. This indicates that the abrupt rise and fall of pitch contours are suppressed.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



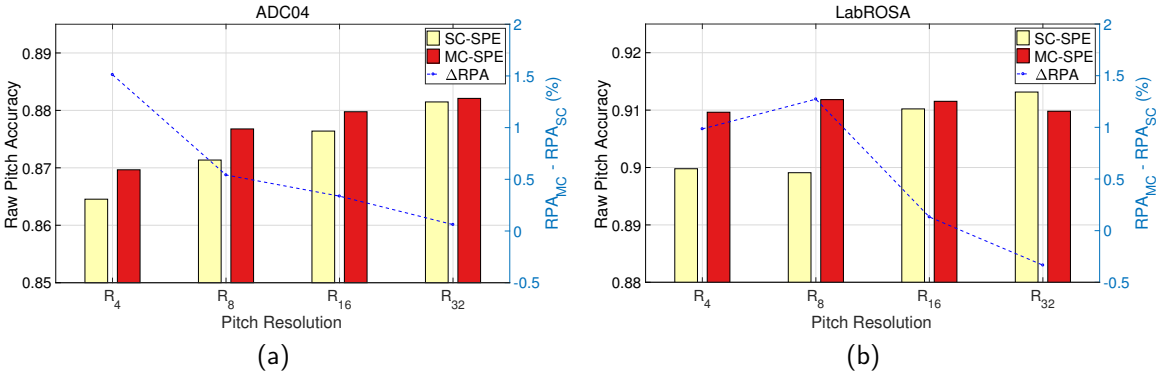


**Figure 3.** Comparison of raw pitch accuracy between SC-SPE and MC-SPE when the pitch resolution is increasing. $\text{RPA}_{SC}$ and $\text{RPA}_{MC}$ correspond to RPA of SC-SPE and MC-SPE, respectively. MC-$\text{SPE}_R$ is a network where a model that combines SC-$\text{SPE}_R$, SC-$\text{SPE}_{R/2}$ and SC-$\text{SPE}_{R/4}$.

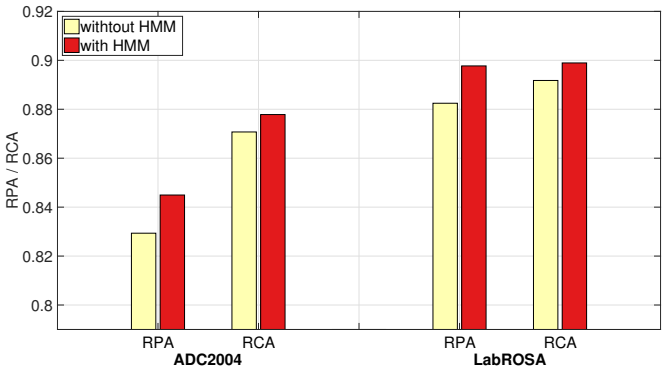


**Figure 4.** Performance increment by HMM-based pitch smoothing on $\text{SC-SPE}_{32}$.

### *5.3. Singing Voice Activity Detector for Melody Extraction*

We compared the performance of the SVAD with VFAD on ADC04 and LabROSA. Table 3 shows the evaluation matrix when $\theta$ is 0, 0.03, and 0,05. The larger the value of $\theta$, the smaller the VFA. If the threshold is set too high, the F1 score is lowered. Using VFAD does not significantly reduce VFA because the number of frame is relatively small to be removed by VFAD among the voice frames detected by the SVAD. However, this process makes it possible to provide more natural melody contours. As shown in Figure 5, the effect of VFAD can be seen by comparing the blue line (obtained from SVAD) with the yellow line (obtained by further using VFAD). Based on the results from Table 3, we set 0.03 as a trade-off.

**Table 3.** Comparison of the proposed SVAD performance with the VFAD according to theta value

| | **ADC04** | | | **LabROSA** | | |
|---|---|---|---|---|---|---|
| $\theta$ | 0 | 0.03 | 0.05 | 0 | 0.03 | 0.05 |
| VR | **0.861** | 0.858 | 0.846 | **0.891** | 0.891 | 0.887 |
| Precision | 0.976 | 0.976 | **0.977** | 0.933 | 0.939 | **0.945** |
| F1 score | **0.915** | 0.914 | 0.907 | 0.912 | 0.914 | **0.915** |
| VFA | 0.113 | 0.112 | **0.106** | 0.121 | 0.109 | **0.097** |

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



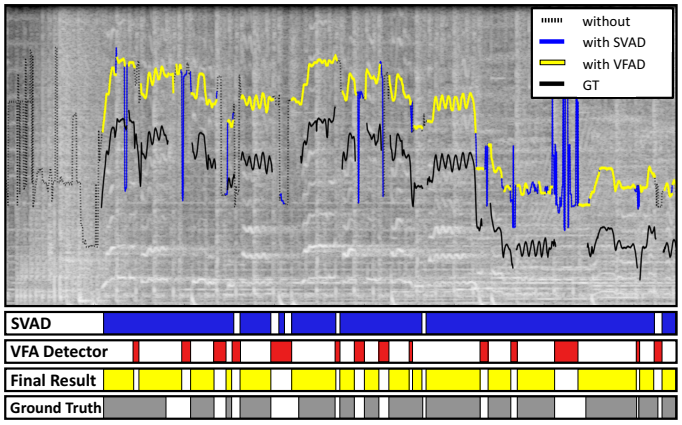


**Figure 5.** An example of singing voice activity detection with VFAD: (1) The SPE predicts the pitch over all frames. (2) The SVAD determines the singing voice frames (blue box in the bottom) and removes non-vocal melody contours (dotted black). However, some melody lines are misidentified as singing voice (blue line). (3) To reduce the false alarm errors, the VFAD determines non-singing voice frames (red box in the bottom). (4) Finally, we obtain more elaborate melody contours (yellow line). (5) The ground truth (black line) is plotted 100Hz below the prediction for visual comparison.

**Table 4.** Comparison of SVAD results on the Jamendo test dataset

| | VR | Precision | F1 score |
|---|---|---|---|
| Lehner [31] | 0.906 | 0.898 | 0.902 |
| Leglaives [12] | **0.926** | 0.895 | 0.910 |
| **Proposed** | 0.893 | **0.933** | **0.913** |

Table 4 compares the proposed method to two state-of-the-arts algorithms on 16 songs in the Jamendo test dataset. The Lehner algorithm is based on LSTM-RNN and the Fluctogram feature to reduce false positives [31] and the Leglaive algorithm is on BLSTM-RNN[12]. We did not show other state-of-the-arts algorithm using DCNN due to different evaluation metrics, for example, [17]. The proposed method has higher precision and lower voice recall than the two. This conservative result in detecting the voice activity is attributed to the VFAD. However, when we evaluate them according to the F1 score, the proposed method slightly outperformed the two compared algorithms.

## 6. Comparison to State-of-the-Art Melody Extraction Methods

### *6.1. Evaluation Metrics*

Table 5 compares the proposed method with state-of-the-art algorithms on the four test datasets. The model used for the final test is SC-SPE$_{32}$. The melody extraction results of the compared algorithms were obtained from MIREX[2]. For ADC04 and MIREX05, we should note that our results are not exactly comparable to them because we used only songs with vocal and also the LabROSA dataset is a subset of MIREX05. Also, we did not list recently reported results based on deep learning [13,16] because they have different test settings.

The proposed method significantly outperforms our previous multi-column DNN model [15] for three datasets. The overall accuracy of the proposed model is above 80% for all datasets except MIR-1K. This might be because the audio files in MIR-1K have poor recording quality. Compared to

[2] http://www.music-ir.org/mirex/wiki/MIREX_HOME

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



**Table 5.** Comparison of Melody Extraction Results

(a) ADC04

| Method | OA | RPA | RCA | VR | VFA |
|---|---|---|---|---|---|
| Dressler [32] | **0.853** | **0.883** | **0.889** | **0.901** | 0.158 |
| Arora et al. [33] | 0.690 | 0.814 | 0.859 | 0.765 | 0.235 |
| Bosch et al. [34] | 0.697 | 0.767 | 0.799 | 0.776 | 0.202 |
| Salamon et al. [14] | 0.740 | 0.772 | 0.794 | 0.806 | 0.152 |
| Ikemiya et al. [5] | 0.719 | 0.814 | 0.846 | 0.849 | 0.435 |
| Kum et al. [15] | 0.731 | 0.758 | 0.783 | 0.889 | 0.412 |
| **Proposed** | 0.811 | 0.798 | 0.802 | 0.859 | **0.112** |

(b) MIREX05

| Method | OA | RPA | RCA | VR | VFA |
|---|---|---|---|---|---|
| Dressler [32] | 0.715 | 0.770 | 0.806 | 0.831 | 0.300 |
| Arora et al. [33] | 0.634 | 0.692 | 0.765 | 0.810 | 0.344 |
| Bosch et al. [34] | 0.637 | 0.688 | 0.7338 | 0.791 | 0.385 |
| Salamon et al. [14] | 0.676 | 0.698 | 0.769 | 0.776 | 0.239 |
| Ikemiya et al. [5] | 0.674 | 0.764 | 0.815 | 0.945 | 0.557 |
| LabROSA | | | | | |
| Kum et al. [15] | 0.684 | 0.776 | 0.786 | 0.870 | 0.490 |
| **Proposed** | **0.859** | **0.842** | **0.844** | **0.891** | **0.109** |

(c) MIR-1k

| Method | OA | RPA | RCA | VR | VFA |
|---|---|---|---|---|---|
| Kum et al. | 0.613 | **0.726** | **0.770** | **0.934** | 0.658 |
| **Proposed** | **0.741** | 0.718 | 0.749 | 0.817 | **0.196** |

(d) iKala

| Method | OA | RPA | RCA | VR | VFA |
|---|---|---|---|---|---|
| **Proposed** | 0.811 | 0.769 | 0.773 | 0.849 | 0.085 |

the results from MIREX, the proposed method achieved better accuracy except those from Dressler on ADC04 [32]. A notable result is that the proposed method has significantly low voice false alarm. This may be attributed to the proposed SVAD that is supported by the VFAD.

### *6.2. Melodia vs. Proposed Method*

In general, classification-based approach to melody extraction produces discrete pitch contours, losing detailed singing information. However, the proposed method can generate nearly continuous curves by increasing the output resolution up to $R_{32}$. Therefore, they preserve natural singing styles such as vibrato or note-to-note transition patterns. In order to confirm the continuity, we obtained the evaluation results by reducing the pitch tolerance to $\pm 25$, $\pm 12.5$ cents, and compared them with the results from Melodia, a saliency-based algorithm that generates the continuous pitch curves [14]. Figure 6 shows that the proposed method (SC-SPE) achieves 5 to 10% higher than Melodia although the performance becomes worse for smaller tolerance. Figure 7 compares an example of pitch contours, each from Meloida and the proposed method. This illustrates more intuitively that the proposed method produces highly continuous curves that are similar to the ground-truth in Hz.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 November 2017 doi:10.20944/preprints201711.0027.v1



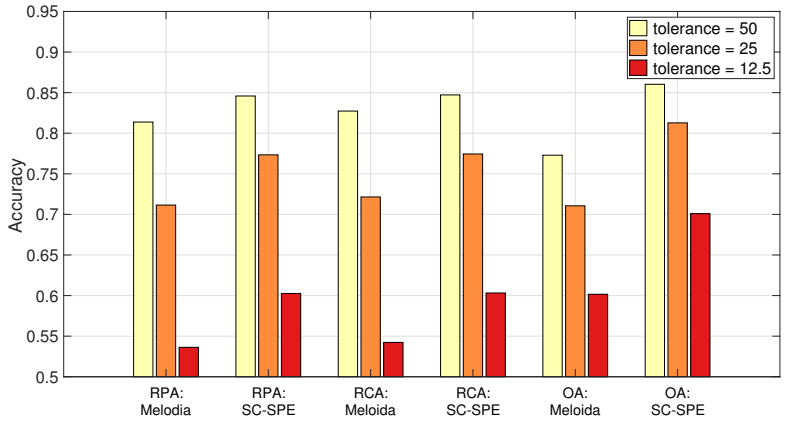


**Figure 6.** Comparison of evaluation matrix of LabROSA according to pitch tolerance. The tolerance value used in the MIREX melody extraction task is 50 cents.

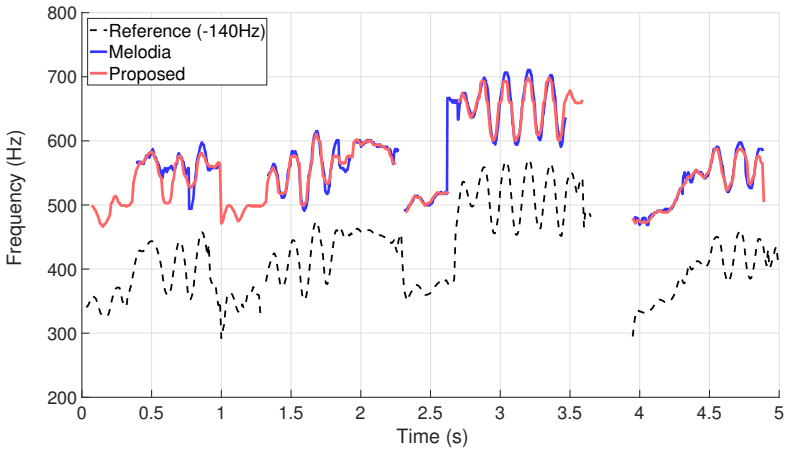


**Figure 7.** Comparison of pitch contours for the 'opera_fem4.wav' of ADC04. The reference pitch was plotted below 140 Hz for visual comparison.

## 7. Conclusions

We proposed a novel melody extraction algorithm composed of singing pitch detector and singing voice activity detector using deep convolution neural networks. We have shown that the SPE can effectively extract melody features and classify pitch classes. Since the pitch can be predicted with a high resolution, the classification-based algorithm can produce nearly continuous curves. The multi-column method for predicting pitches of various resolutions can improve performance in DCNN, but the effect becomes less significant as the pitch resolution is higher. We propose a high performance SVAD with VFAD to minimizes false positive errors. Finally, we compared our melody extraction model to previous state-of-the-arts methods on several public test dataset and showed that the results are comparable to those from the best.

**Supplementary Materials:** The demo of the proposed melody extraction method is available at http://mac-bach.kaist.ac.kr/keums/melodyExtraction.

**Acknowledgments:** This research was supported by Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Science, ICT & Future Planning (2015R1C1A1A02036962)

 doi:10.20944/preprints201711.0027.v1



## References


1. Ghias, A.; Logan, J.; Chamberlin, D.; Smith, B.C. Query by humming: musical information retrieval in an audio database. In Proceedings of the ACM international conference on Multimedia. ACM, 1995, pp. 231–236.
2. Serra, J.; Gómez, E.; Herrera, P. Audio Cover Song Identification and Similarity: Background, Approaches, Evaluation, and Beyond. *Advances in Music Information Retrieval* **2010**, *274*, 307–332.
3. Salamon, J.; Rocha, B.; Gómez, E. Musical genre classification using melody features extracted from polyphonic music signals. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2012, pp. 81–84.
4. Kako, T.; Ohishi, Y.; Kameoka, H.; Kashino, K.; Takeda, K. Automatic Identification for Singing Style based on Sung Melodic Contour Characterized in Phase Plane. In Proceedings of the International Society for Music Information Retrieval (ISMIR), 2009, pp. 393–398.
5. Ikemiya, Y.; Itoyama, K.; Yoshii, K. Singing voice separation and vocal F0 estimation based on mutual combination of robust principal component analysis and subharmonic summation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **2016**, *24*, 2084–2095.
6. Salamon, J.; Gómez, E.; Ellis, D.P.; Richard, G. Melody extraction from polyphonic music signals: Approaches, applications, and challenges. *IEEE Signal Processing Magazine* **2014**, *31*, 118–134.
7. Ellis, D.P.W.; Poliner, G.E. Classification-based melody transcription. *Machine Learning* **2006**, *65*, 439–456.
8. Bittner, R.M.; Salamon, J.; Essid, S.; Bello, J.P. Melody extraction by contour classification. In Proceedings of the International Society for Music Information Retrieval (ISMIR), 2015, pp. 500–506.
9. Hsu, C.L.; Jang, J.S.R. On the Improvement of Singing Voice Separation for Monaural Recordings Using the MIR-1K Dataset. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* **2010**, *18*, 310–319.
10. Bittner, R.M.; Salamon, J.; Tierney, M.; Mauch, M.; Cannam, C.; Bello, J.P. MedleyDB: A Multitrack Dataset for Annotation-Intensive MIR Research. In Proceedings of the International Society for Music Information Retrieval (ISMIR), 2014, Vol. 14, pp. 155–160.
11. Chan, T.S.; Yeh, T.C.; Fan, Z.C.; Chen, H.W.; Su, L.; Yang, Y.H.; Jang, R. Vocal activity informed singing voice separation with the iKala dataset. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015, pp. 718–722.
12. Leglaive, S.; Hennequin, R.; Badeau, R. Singing voice detection with deep recurrent neural networks. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015, pp. 121–125.
13. Rigaud, F.; Radenen, M. Singing Voice Melody Transcription using Deep Neural Networks. In Proceedings of the International Society for Music Information Retrieval (ISMIR), 2016, pp. 737–743.
14. Salamon, J.; Gómez, E. Melody extraction from polyphonic music signals using pitch contour characteristics. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* **2012**, *20*, 1759–1770.
15. Kum, S.; Oh, C.; Nam, J. Melody Extraction on Vocal Segments Using Multi-Column Deep Neural Networks. In Proceedings of the International Society for Music Information Retrieval (ISMIR). ISMIR, 2016, pp. 819–825.
16. Park, H.; Yoo, C.D. Melody extraction and detection through LSTM-RNN with harmonic sum loss. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017, pp. 2766–2770.
17. Schlüter, J.; Grill, T. Exploring Data Augmentation for Improved Singing Voice Detection with Neural Networks. In Proceedings of the International Society for Music Information Retrieval (ISMIR), 2015, pp. 121–126.
18. Kelz, R.; Dorfer, M.; Korzeniowski, F.; Böck, S.; Arzt, A.; Widmer, G. On the Potential of Simple Framewise Approaches to Piano Transcription. In Proceedings of the International Society for Music Information Retrieval (ISMIR), 2016, pp. 475–481.
19. Korzeniowski, F.; Widmer, G. A fully convolutional deep auditory model for musical chord recognition. IEEE International Workshop on Machine Learning for Signal Processing (MLSP). IEEE, 2016, pp. 1–6.
20. Lee, J.; Park, J.; Kim, K.L.; Nam, J. Sample-level Deep Convolutional Neural Networks for Music Auto-tagging Using Raw Waveforms. In Proceedings of the Sound and Music Computing Conference (SMC), 2017, pp. 220–226.

21. Lin, M.; Chen, Q.; Yan, S. Network in network;. *arXiv preprint arXiv:1312.4400* **2013**.
22. Lehner, B.; Widmer, G.; Sonnleitner, R. On the reduction of false positives in singing voice detection. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014, pp. 7480–7484.
23. Goto, M.; Hashiguchi, H.; Nishimura, T.; Oka, R. RWC Music Database: Popular, Classical and Jazz Music Databases. In Proceedings of the International Society for Music Information Retrieval (ISMIR), 2002, Vol. 2, pp. 287–288.
24. Ramona, M.; Richard, G.; David, B. Vocal detection in music with support vector machines. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2008, pp. 1885–1888.
25. Poliner, G.E.; Ellis, D.P.; Ehmann, A.F.; Gómez, E.; Streich, S.; Ong, B. Melody transcription from music audio: Approaches and evaluation. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* **2007**, *15*, 1247–1256.
26. Laroche, J. Time and pitch scale modification of audio signals. In *In Proceedings of the Applications of digital signal processing to audio and acoustics*; Springer, 2002; pp. 279–309.
27. Raffel, C.; McFee, B.; Humphrey, E.J.; Salamon, J.; Nieto, O.; Liang, D.; Ellis, D.P.; Raffel, C.C. mir_eval: A transparent implementation of common MIR metrics. In Proceedings of the International Society for Music Information Retrieval (ISMIR). Citeseer, 2014.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE international conference on computer vision, 2015, pp. 1026–1034.
29. Chollet, F. Keras. https://github.com/fchollet/keras, 2015.
30. Sokolova, M.; Lapalme, G. A systematic analysis of performance measures for classification tasks. *Information Processing & Management* **2009**, *45*, 427–437.
31. Lehner, B.; Widmer, G.; Bock, S. A low-latency, real-time-capable singing voice detection method with LSTM recurrent neural networks. In Proceeding of the Signal Processing Conference (EUSIPCO). IEEE, 2015, pp. 21–25.
32. Dressler, K. Pitch estimation by the pair-wise evaluation of spectral peaks. In Proceedings of the AES Conference on Semantic Audio. Audio Engineering Society, 2011.
33. Arora, V.; Behera, L. On-line melody extraction from polyphonic audio using harmonic cluster tracking. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* **2013**, *21*, 520–530.
34. Bosch, J.J.; Bittner, R.M.; Salamon, J.; Gómez, E. A Comparison of Melody Extraction Methods Based on Source-Filter Modelling. In Proceedings of the International Society for Music Information Retrieval (ISMIR), 2016, pp. 571–577.