MUSIC ARTIST CLASSIFICATION WITH WAVENET CLASSIFIER FOR RAW WAVEFORM AUDIO DATA

A PREPRINT

Xulong Zhang*

School of Computer Science and Technology Fudan University Shanghai, 201203 China xlzhang14@fudan.edu.cn

Yongwei Gao

School of Computer Science and Technology Fudan University Shanghai, 201203 China ywgao16@fudan.edu.cn

Yi Yu

Digital Content and Media Sciences Research Division National Institute of Informatics Tokyo, 101-8430 Japan yiyu@nii.ac.jp

Wei Li

School of Computer Science and Technology
 Shanghai Key Laboratory of Intelligent Information Processing

 Fudan University
 Shanghai, 200433 China
 weili-fudan@fudan.edu.cn

April 10, 2020

ABSTRACT

Models for music artist classification usually were operated in the frequency domain, in which the input audio samples are processed by the spectral transformation. The WaveNet architecture, originally designed for speech and music generation. In this paper, we propose an end-to-end architecture in the time domain for this task. A WaveNet classifier was introduced which directly models the features from a raw audio waveform. The WaveNet takes the waveform as the input and several downsampling layers are subsequent to discriminate which artist the input belongs to. In addition, the proposed method is applied to singer identification. The model achieving the best performance obtains an average F1 score of 0.854 on benchmark dataset of Artist20, which is a significant improvement over the related works. In order to show the effectiveness of feature learning of the proposed method, the bottleneck layer of the model is visualized.

Keywords Artist classification · Music information retrieval · WaveNet classifier · Raw audio

1 Introduction

With the dramatically increasing number of music data in the music world, how to quickly retrieve the desired song is very difficult. Therefore, the content-based automatic analysis of the song becomes important for music information retrieval (MIR). The artist's name, as important song information, is one of the direct factors for discriminating a song. All karaoke systems and music stores usually categorize a large database by using the names of artists. Artist classification can also be used for song recommendation based on similar artists. Although most of the audio files

^{*}https://orcid.org/0000-0001-7005-992X

in the standard music collection contain artist tag information within metadata. However, in many cases, audio files and metadata cannot be directly obtained, such as listening to songs and music recognition, extracting audio from film and television shows, and recording live concerts. Hence, automatic artist classification is a very meaningful and valuable task in MIR. However, there are two main factors lead to this task very challenging, one is that there are many artists, and the songs of each artist are uneven in number. The other is that copyright protection causes the problem that numerous songs are difficult to obtain.

Existing artist classification methods [1, 2, 3] are based on traditional machine learning. The raw audio data are divided into short frames, and audio features in the frequency domain at each frame are calculated as input data of the model. The input data is used to train a classifier such as the Gaussian Mixture Model (GMM) [4], KNN[5], and MLP[6], etc. Since the input data is calculated only from one frame, the context information of the music is ignored. To avoid this problem, song-level audio feature is introduced to artist identification [7]. But the song-level feature is directly statistic features over on the successive frame. Previous works [8] focus more on feature engineering, but the design of features is complicated.

In recent years, deep learning based methods have been developed rapidly in music information retrieval. In [9],Mel-Frequency Cepstrum Coefficients (MFCCs) are calculated as the input and a stacked Convolutional and Recurrent Neural Network (CRNN) model is built to learn the mapping between MFCCs and its corresponding artist name. While the model is able to learn the context information from the time-frequency features, the transformation from raw audio data with 640 float data to the MFCCs with just 13 coefficients leads to information loss. And the temporal structure of the MFCCs are based on the spectrum with frame level structure, in which the time granularity is more rougher than the raw audio data sample point [10].

Motivated by the WaveNet used as classification in voice activity detection[11]. In this paper, we study an end-to-end artist classification method in the time-domain, which directly models the features of artists from raw audio waveform. Compared with ordinary Convolutional Neural Networks (CNN) or feedforward networks, the WaveNet architecture is better able to handle the long-term temporal dependencies that exists in audio signal [11]. Therefore, we use the WaveNet architecture for feature extraction. Subsequently, several convolutional layers and maxpooling layers are used to identify the artist name. Experiments show that the proposed method achieves superior results compared to state-of-the-art algorithms[7].

The rest of this paper is organized as follows. Section 2 introduces the related works, i.e., audio feature representation, artist classification baseline, etc. Section 3 presents the proposed artist classification method. Experiment results are shown and discussed in Section 4. Section 5 concludes the paper and points out the future works.

2 Related works

2.1 Audio feature representation

There are many common audio features in music information retrieval, such as MFCCs[12], Chroma[4], Linear Prediction Cepstrum Coefficients (LPCCs)[13], spectral flux, spectral centroid, spectral rolloff, etc. Generally, the calculation of these features is statistics on the spectrum. However, these feature calculation from the high-dimensional vector of raw waveform data to low-dimensional feature representations usually miss lots of information in the original spectrum or waveform. In recent years, more and more researchers attempt to build models to directly process raw audio data for various tasks in MIR. In [14], raw audio waveforms are used as the input for speech recognition. In [15], the authors proposed an acoustic model that takes raw multi-channel waveforms as the input. Through directly operating in the time domain, feature extraction is not required. The network can take advantage of the temporal structure of signals which is lost by conducting the fixed time-frequency transformation.

2.2 Deep learning for MIR tasks

In recent years, deep learning methods have significantly improved the performance of MIR tasks. For example, Sharama *et al.* used Wave-U-Net [16] for singer identification task and obtained the state-of-the-art results. Zhang *et al.*[17] built a CNN model taking several concatenated audio features as the input for singing voice detection task and achieved best performance in term of F1 measure. Choi *et al.* [18] applied transfer learning method into music classification and regression tasks, which outperforms the MFCCs-based baseline method by adopting the convnet features. Jansson *et al.* [19] applied the U-Net architecture, designed for image processing, into the task of source separation, and achieved the state-of-the-art performance.

In the MIR, auto tagging and some other classification tasks such as acoustic scene calssification is similar to artist classification. Eghbal-zadeh *et al.* [20] combine the i-vectors with CNN for acoustic scene classification (ASC).

Fonseca et al. [21, 22] do a fusion of shallow and deep learning for ASC. Xu et al. [23] applied CNN with statistical features for general audio tagging.

Lee *et al.*[24] proposed sample-level deep couvolutional neural networks (DCNN) with using raw waveform for music auto-tagging. The sample-level DCNN learn representations from sampling point of waveform beyond typical frame-level input representations. By use the DCNN on the raw waveform the accuracy in music auto-tagging is improved and comparable to previous state-of-the-art performances on public datasets.

For the artist classification task, Keunwoo *et al.* [18] used artist label train convolutional network and combined the middle layers output as a pre-trained convnet feature. With artist label trained convenet feature outperform the MFCC feature.

2.3 Artist classification baseline

For the artist classification task, Ellis *et al.* [4] investigated beat-synchronous Chroma feature, which is designed to reflect melodic and harmonic content and be invariant to instrumentation. Finally, the frame-level features MFCCs and Chroma are combined with Gaussian as for the classifier, the accuracy of the combined feature is 0.57.

In [7], Eghbal-zadeh *et al.* proposed a song-level descriptor, i-vectors, which is calculated by using the frame-level timbre features MFCCs. The i-vector provides a low-dimensional and fixed-length representation for each song and can be used in a supervised and unsupervised manner. With the application of i-vector in artist classification, which is the current state of the art on the Artist20 dataset [4].

2.4 WaveNet

WaveNet[14], firstly proposed for Text to Speech (TTS) system with timbre features along to speaker, has been used as encoder to learn the representation vector from raw wave data.In [25, 15], the WaveNet architecture is proposed for speech / music generation in the time domain, where the predicted distribution of each audio sample is conditional on its previous audio samples.

In [14], the WaveNet architecture is used as a discriminant model which achieves considerable results for the phoneme recognition task. In recent works, the WaveNet is popularly used in speech tasks such as voice activity detection [11] and speech augmentation [26].

3 Methodology

3.1 Architecture

The architecture of the proposed network is shown in Fig. 1, in which the network on the left side is novel to this paper. Firstly, it takes the fix-length segment as the input and feeds into the WaveNet encoder. Subsequently, several convolutional neural network layers are built to discriminate which artist the input belongs to. Below we described them in detail.

The WaveNet is an auto-regressive network that directly estimates a raw waveform from the sample point in the temporal domain. Given the waveform sequence $x = x_1, ..., x_N$ as input, the model estimates the joint probability of the signal as follows:

$$p(x) = \prod_{n=1}^{N} p(x_n x_{n-R-1}, x_{n-R}, \dots, x_{n-1}, \Delta)$$
 (1)

where Δ represents model parameters and R represents the receptive field length. The architecture of WaveNet consists of several stacks of residual blocks, including gated activation, dilated causal convolution, and convolutions. Within a residual block, the gated activation function is defined as:

$$z = \tanh W_{f,l} * x \odot \delta(W_{g,l} * x) \tag{2}$$

where \odot represents the element-wise product operator, * is a causal convolution operator, l denotes the layer index, g and f represent a gate and a filter, respectively, and l denotes a trainable convolution filter. The number of filters and the kernel size, number of residual blocks in the WaveNet are adapted directly from prior work[27]. The skip connections are taken the sum from residual blocks. The input size and the number of filters of convolution operations

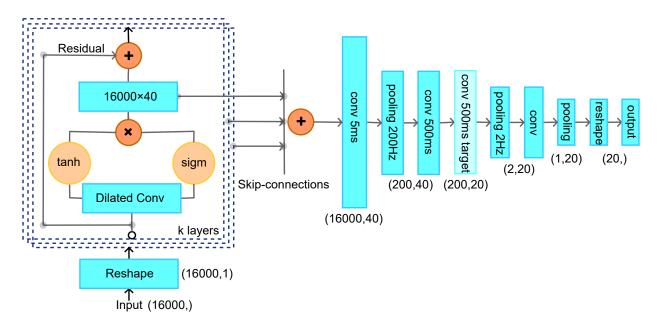


Figure 1: Overview of the WaveNet based deep model architecture. The left part is wavenet for encoder and the right part is CNN for the final classification.

in residual blocks are fixed by experiment with grid search in a range. After that, the output of size (16000,40) are extracted to feed into the CNN for classification. The layers number of CNN classifier and the number of filters of convolution operations are modified from the setting in the prior work[24]. The output size are same with the number of artist for classification.

The architecture of WaveNet model used in this paper is shown in Figure 1. The encoder WaveNet consists of several stacked residual blocks with dilated convolutions, having large receptive field without increasing the computational cost. A dilated convolution is a convolution where the filter is used in a region larger than its size by skipping input values with a predetermined dilation factor. Skipping the input values enables the inflated convolution to have a larger receiving field than the standard convolution. Several dilated convolutions are then stacked to further increase the receptive field.

Once the embedding of the audio signal is obtained by the WaveNet-based network, the representation is then fed to the classification layer. Classification layers are built by stacking several convolutional and pooling layers, as shown in the right of Fig. 1. Each pooling layer acts as a process downsampling raw waveform data.

3.2 Settings and Training

Waveform of each track is divided into fixed-length blocks in the time domain. According to ground truth of each track, all blocks are labelled.

The WaveNet encoder adopted in this paper contains a residual block, which is formed by stacking 9 layers of dilated convolutions with a certain exponentially increasing dilated factor (from 1 to 512 in this block).

We empirically set the channel of each dilated convolutional layer with a fixed size of 40. The outputs of all residual blocks are then added up and fed into a one-dimensional regular convolution with filter number of 16000 and kernel size of 40. Therefore, the dimension of the obtained embedded feature space is (16000, 40). Subsequently, an adaptive one-dimensional average pooling layer is finally used which operates in the time domain to further aggregate the activations of all residual blocks. The number of artists we consider in this paper is 20. A one-dimensional filter with size of 2 is used for all regular convolutions and dilated convolutions throughout the network in the WaveNet. And the ReLU nonlinearity is adopted as an activation function for each convolution layer.

In the training stage, batch size and learning rate are set into 32 and 0.001, respectively. The Adam optimizer [28] is adopted because it has shown strong performance in convolution-based classification tasks with limited hyper-parameter [9]. The early stop is added with the patience of 10 to avoid over-fitting. The weights of the best model can be saved

according to accuracy calculated on validation dataset during the training. The codes of this work are available at https://gitlab.com/exp_codes/wavnet_clf_4_sid.git.

4 Experimental results and discussion

4.1 Dataset

Totally three datasets are used in the experiments for evaluation, including two public datasets (Artist20 and MIR1k) and a self-made dataset(Singer107).

The Singer107 dataset were collected from the online music service, it contains 107 singers and each singer have 3 albums totally 3262 tracks.

The MIR1K dataset² containing 19 singers was used, which contains 1000 clips of singing recordings. Each track in the MIR1K dataset is two-channel, where the left channel is music accompaniment component and the right channel is vocal component of a single singer. In our experiments, only vocal components in the right channel are used. This datasets of 1000 cips was randomly divided into three subsets, 712 clips for training, 124 clips for validation, and 164 clips for testing.

The dataset Artist20 [4] was published for the evaluation of the performance of artist classification, consisting of 1413 MP3 tracks from 20 artists. Each track is monophonic with a sample rate of 16KHz. To our knowledge, the Artist20 dataset is the only public dataset for artist classification. There are different music genre in the Artist20 such as Pop, Rock and Folk. For the dataset split, there are two different split methods have been applied on Artist20 in the previous studies. The one is based on albums, where four albums are used for training, one album for validation, and one album for testing. The other one is based on song split, which is according to the ratio of 8:1:1 to divide 1413 music tracks into a train set, a validate set, and a testing set. The details of two split methods are shown in Table. 1.

Table 1: The a	vailable	tracks i	n differe	ent split of the a	rtist20 dataset
	Split	Subset	Tracks	Total duration	•

Split	Subset	Tracks	Total duration
album	train	918	63h 57min
album	test	249	16h 22min
album	valid	246	16h 26min
song	train	1140	77h 22min
song	test	131	9h 11min
song	valid	142	10h 11min

The dataset splitting method by song has more data for model training, and the song splitting method is used in our experiments. Since the baseline method [7] only used the training and testing subset, the validate subset is not used in our re-implementation. This ensures that the training and testing data we use is completely consistent with that used in the baseline method.

4.2 Evaluation Metrics

In order to give a comprehensive view of the results, we compare model predictions with the ground truth labels to obtain true positives(TP), false positives (FP) true negatives (TN) and false negatives (FN) over all artists in the test set. To summarize results, the metric of accuracy, precision, recall and f1 are calculated for each artist and the mathematical definition are shown in Equation (3).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 * precision * recall}{precision + recall}$$
(3)

²https://sites.google.com/site/unvoicedsoundseparation/mir-1k

Artist classification is a multi-class classification problem, and the macro F1 is used in evaluation for the performance. The multi-class evaluation is divided into multiple two-class evaluations, the F1 value of each two-class classification is calculated, and then the average of the F1 scores of all two categories is taken as macro F1. Besides, the metrics accuracy, precision, and recall are provided. During training, the metrics accuracy and loss on the validation are used to adjust parameters in the model. The mathematical definition of the macro metrics of f1, accuracy, precision and recall are shown in Equation (4).

$$Macro([accuracy, precision, recall, f1]) = \frac{\sum_{n=1}^{N} [accuracy, precision, recall, f1]}{N} \tag{4}$$

Where the n represent the n^{th} artist, and the N is the total number of the artists.

Each metric is calculated by using two evaluation methods:

- 1) Evaluation in units of each song: The final predicted label of each song was voted by its all segments' prediction.
- 2) Evaluation in units of each segment: All predicted segments were used to evaluation.

4.3 input size selection

The input size is an important factor for the performance of the proposed method. In this section, we varied the input size from 0.5 seconds to 2 seconds with a step of 0.5 seconds. Due to the limitation of computing resources, the inputs of more than 2 seconds were not evaluated. For each input size, we ran our method on the validation dataset of Artist20 by using the above two evaluation methods.

As shown in Tab. 2, when the input size was settled as 1 second, each metric reaches their maxima. Therefore, we chose the 1s as the input size in our experiments.

We can also observe that the trained model at song level significantly improves the performance at each metric. Since the model training is input in unit of segment, a full song contains 180 segments or more. At the beginning and end of the song and some areas in the middle that do not contain obvious features, classification errors are prone to occur. Voting by all the segments corresponding to the song can reduce the impact of error-prone segments, and the results also show that the classification precision and recall of the song level evaluation have been improved.

	segment			song				
segment size (s)	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
0.5	0.387	0.417	0.385	0.392	0.756	0.815	0.756	0.757
1.0	0.549	0.566	0.546	0.549	0.854	0.881	0.851	0.854
1.5	0.423	0.432	0.431	0.424	0.632	0.634	0.651	0.612
2.0	0.418	0.421	0.488	0.428	0.595	0.591	0.732	0.616

Table 2: The performance with different segment size as input on Artist20.

On the test dataset, the error prediction cases are listed in Tab. 3. The most error prone are artist *madonna* and *tori_amos*, with listening to the error predict tracks the music have a similar style. Furthermore, the confusion matrices calculated on testing set at segment level and that at the song level are displayed in Fig. 2 and Fig. 3, respectively, for visual comparison. In confusion matrix, the diagonal lines means that the points generated based on the predicted results and the ground truth. The vertical axis of the confusion matrix is the actual artist label, and the horizontal axis is the predicted artist label. Therefore, the clearer the diagonals in the confusion matrix, the better the classification effect. Obviously, we can observe that classification results at song level are better.

4.4 Comparison results with baseline

In this section, we compare our method with baseline methods, i.e., i-vector-AC [7]. We re-implemented it and trained it on the datasets we used for fair comparisons. The comparison results are shown in Table. 4, where the proposed method is named as WaveNet-CNN-AC. It can be seen that our method outperformed the baseline method at each metrics. Compared with the best performance of existing method, it obtained a gain of 3.3 % at Accuracy, 7.6 % at Precision, 3.2 % at Recall, and 4.2 % at F1-measure.

Table 3: Model	predict error	cases according	full song	length
Table 5. Model	DICUICL CITO	Cases according	run song	ICHYUI

Ground Truth	Predict	Tracks
radiohead	dave_matthews_band	1
radiohead	madonna	1
depeche_mode	fleetwood_mac	1
depeche_mode	beatles	1
aerosmith	cure	1
u2	madonna	1
u2	tori_amos	1
u2	prince	1
suzanne_vega	tori_amos	1
steely_dan	fleetwood_mac	1
cure	u2	1
cure	metallica	1
prince	fleetwood_mac	2
prince	suzanne_vega	1
madonna	tori_amos	3
roxette	madonna	1

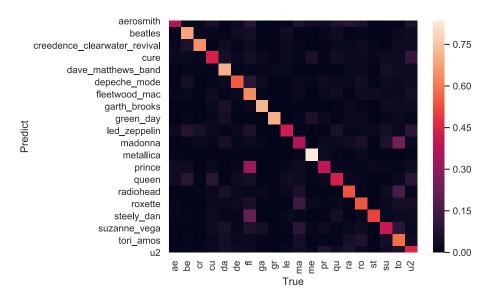


Figure 2: The confusion matrix of the segment level on Artist20. (The vertical axis is the true artist label with abbreviation, and the horizontal axis is the predict artist label by the model.)

Table 4: Comparison with baseline method on benchmark dataset of Artist20

Method	Split	Accuracy	Precision	Recall	F1
	Song		0.805	0.819	
WaveNet-CNN-AC	Song	0.854	0.881	0.851	0.854

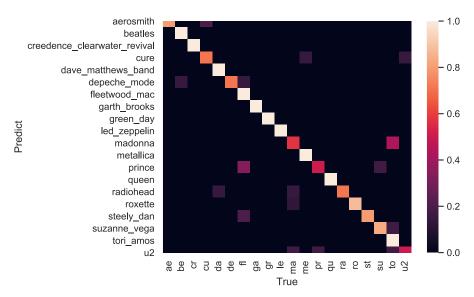


Figure 3: The confusion matrix of the song level evaluation on Artist20.

The size the 20 artists differs greatly from the number of artists in practical applications. We constructed and expanded the number of artists in the self-built Singer107 dataset, in which the artist category was expanded from 20 to 107 for comparison experiments. The results are shown in Table 5.

Table 5: Comparison with baseline method under the same conditions on Singer107 dataset

Method	Split	Accuracy	Precision	Recall	F1
i-vector-AC	Song	0.701	0.723	0.704	0.714
WaveNet-CNN-AC	Song	0.796	0.804	0.823	0.816

From the experimental results on the dataset of 107 singers, the proposed method can still maintain high performance. The contrast baseline method decreases the recognition accuracy more significantly as the number of artists increases, while the proposed method can achieve an F1 value of 0.816.

To intuitively verify the effectiveness of the proposed model, visualization of the bottleneck layer were conducted.

After the bottleneck layer of the network, each input segment has been converted to a vector. For visualization, here we used a t-SNE method [29] to reduce the 20-dimension vector outputted by the bottleneck layer to a 2-dimension vector, shown in Fig. 4. There are 20 artists with difference colors. We can see that the tracks belonging to the same artist are almost gathered together and the tracks belonging to different artists are almost separated, which demonstrates that the features learned by the proposed model have a good ability to distinguish artists. There also show overlaps on different clusters, the similar artist is prone to error classification on the domain of two-dimension. The rock bands such as Beatles, Queen, Aerosmith and Led zeppelin are clustered in the nearby section in the figure. while pop artists such as Madonna and Roxette are cluster in another corner in the figure.

4.5 Transfering into singer identification task

Singer identification task and artist classification task are similar in MIR. The main difference is that singer identification is to distinguish a single musician, while artist classification is to recognize a music band containing multiple musicians. Most artists in the dataset Artist20 are music bands. For example, there are four musicians in the rock band U2. In this section, we conducted an experiment to investigate the effectiveness of the proposed algorithm on the singer identification task.

The evaluation results are listed in Table 6. We can see that the proposed method obtained 95.7 % at accuracy, which means that error rate is 4.3%. On the MIR1K dataset the proposed method outperform the state-of-the-art method by

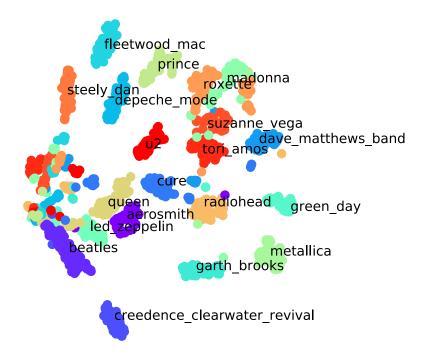


Figure 4: Visualization the vectors outputted by bottleneck layer.

Table 6: Evaluation for singer identification on MIR1K dataset

Method	Accuracy	Precision	Recall	F1
i-vector-AC	0.906	0.899	0.899	0.070
WaveNet-CNN-AC	0.957	0.960	0.942	0.937

4%. In Tab. 4, the error rate on the Artist20 test set is 14.5%. It can be seen that the result of singer identification on the MIR1K dataset are better than the artist classification results on Artist20.

The confusion matrix with song level and segment level of singer identification on the MIR1K dataset are also shown in Figure 5 and 6 separately. From the confusion metric, it can be seen that the most error-prone are located between singers with ID 6 and 7. The error cases were list in Table 7

Table 7: Singer identification error cases on MIR1K dataset

Ground Truth	Predict	Tracks
yifen	annar	1
stool	abjones	1
leon	jmzen	1
bug	bobon	3
jmzen	amy	1

Overall, the proposed method also performs well in the singer identification task. Through listening the error-prone clips with singer name *bug* and the singer name *bobon* in Table 7, we observe that these two singer are both male and similar. For this case, the proposed method worked poorly.



Figure 5: The confusion matrix of the segment level evaluation with 1s as input length on MIR1K dataset.

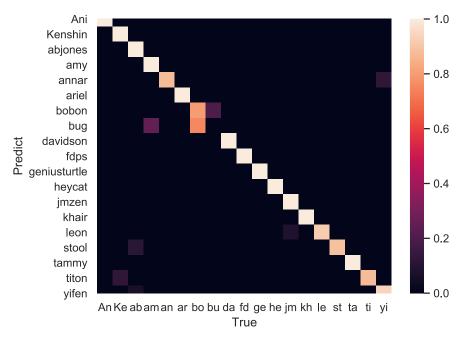


Figure 6: The confusion matrix of the song level with 1s input length on MIR1K dataset.

4.6 Discussion

The task of artist classification is still a challenging task in MIR. There are many difficulties to solve. Firstly, although the characteristics of different voices, e.g., timbre, are helpful to this task. However, singers with similar voices are difficult to recognize. Secondly, a song sung by a music band usually contains multiple singing voices, which undoubtedly increases the difficulty.

Besides, data imbalance is also a crucial problem for the data-driven based artist classification method. The number of albums and corresponding songs of the 20 artists in the Artist20 dataset is basically balanced. However, in fact, the number of artists online are huge, and the number of their albums and songs is uneven, which will be a great challenge for model training. With the continuous inclusion of new artists, the model needs to continuously add new training samples for continuous update and iteration.

5 Conclusion

In this paper, a classification method based on the WaveNet model is proposed for the artist classification task, which takes the raw audio waveform in the time domain as the input. The experimental results on the dataset Artist20 show that the proposed method processing in the time domain can learn effective features to distinguish artists.

Compared with the state-of-the-art method under the same experiment conditions, the results show that the proposed method perform better, achieving an F1 value of 0.854 with the song level evaluation on Artist20. Lastly, the proposed method is transplanted into the task of singer identification task, which also works well.

The effective combination of frequency domain and time domain is of great significance to music information retrieval. In the future, we plan to combine the raw waveform and spectrum of the audio signal to jointly construct a hybrid model for artist classification task.

Acknolegement

This work was supported by National Key R&D Program of China (2019YFC1711800), NSFC (61671156).

References

- [1] Christophe Charbuillet, Damien Tardieu, Geoffroy Peeters, et al. Gmm supervector for content based music similarity. In *International Conference on Digital Audio Effects, Paris, France*, pages 425–428, 2011.
- [2] Taufiq Hasan and John HL Hansen. A study on universal background model training in speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):1890–1899, 2011.
- [3] Xulong Zhang, Yiliang Jiang, Jin Deng, Juanjuan Li, Mi Tian, and Wei Li. A novel singer identification method using gmm-ubm. In *Proceedings of the 6th Conference on Sound and Music Technology (CSMT)*, pages 3–14. Springer, 2019.
- [4] Daniel PW Ellis. Classifying music audio with timbral and chroma features. In *Proceedings of the 8th International Conference on Music Information Retrieval*, 2007.
- [5] Tushar Ratanpara and Narendra Patel. Singer identification using perceptual features and cepstral coefficients of an audio signal from indian video songs. EURASIP Journal on Audio, Speech, and Music Processing, 2015(1):16, 2015.
- [6] Ying Hu and Guizhong Liu. Singer identification based on computational auditory scene analysis and missing feature methods. *Journal of Intelligent Information Systems*, 42(3):333–352, 2014.
- [7] Hamid Eghbal-Zadeh, Bernhard Lehner, Markus Schedl, and Gerhard Widmer. I-vectors for timbre-based music similarity and music artist classification. In *ISMIR*, pages 554–560, 2015.
- [8] Jongpil Lee, Jiyoung Park, and Juhan Nam. Representation learning of music using artist, album, and track information. *arXiv preprint arXiv:1906.11783*, 2019.
- [9] Zain Nasrullah and Yue Zhao. Music artist classification with convolutional recurrent neural networks. In 2019 International Joint Conference on Neural Networks (IJCNN), pages 1–8. IEEE, 2019.
- [10] Tara N Sainath, Ron J Weiss, Andrew Senior, Kevin W Wilson, and Oriol Vinyals. Learning the speech frontend with raw waveform cldnns. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

- [11] Ido Ariav and Israel Cohen. An end-to-end multimodal voice activity detection using wavenet encoder and residual networks. *IEEE Journal of Selected Topics in Signal Processing*, 13(2):265–274, 2019.
- [12] Todor Ganchev, Nikos Fakotakis, and George Kokkinakis. Comparative evaluation of various mfcc implementations on the speaker verification task. In *Proceedings of the SPECOM*, volume 1, pages 191–194, 2005.
- [13] Ooi Chia Ai, M Hariharan, Sazali Yaacob, and Lim Sin Chee. Classification of speech dysfluencies with mfcc and lpcc features. *Expert Systems with Applications*, 39(2):2157–2165, 2012.
- [14] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. arXiv preprint arXiv:1609.03499, 2016.
- [15] Yedid Hoshen, Ron J Weiss, and Kevin W Wilson. Speech acoustic modeling from raw multichannel waveforms. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4624–4628. IEEE, 2015.
- [16] Bidisha Sharma, Rohan Kumar Das, and Haizhou Li. On the Importance of Audio-Source Separation for Singer Identification in Polyphonic Music. In *Proc. Interspeech 2019*, pages 2020–2024, 2019.
- [17] Xulong Zhang, Shengchen Li, Zijin Li, Shizhe Chen, Yongwei Gao, and Wei Li. Singing voice detection using multi-feature deep fusion with cnn. In *Proceedings of the 7th Conference on Sound and Music Technology (CSMT)*, pages 41–52. Springer, 2020.
- [18] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. Transfer learning for music classification and regression tasks. *arXiv preprint arXiv:1703.09179*, 2017.
- [19] Andreas Jansson, Eric J. Humphrey, Nicola Montecchio, Rachel M. Bittner, Aparna Kumar, and Tillman Weyde. Singing voice separation with deep u-net convolutional networks. In *ISMIR*, 2017.
- [20] Hamid Eghbal-zadeh, Bernhard Lehner, Matthias Dorfer, and Gerhard Widmer. A hybrid approach with multichannel i-vectors and convolutional neural networks for acoustic scene classification. In 2017 25th European Signal Processing Conference (EUSIPCO), pages 2749–2753. IEEE, 2017.
- [21] Eduardo Fonseca, Rong Gong, and Xavier Serra. A simple fusion of deep and shallow learning for acoustic scene classification. *arXiv preprint arXiv:1806.07506*, 2018.
- [22] Rong Gong, Eduardo Fonseca, Dmitry Bogdanov, Olga Slizovskaia, Emilia Gomez, and Xavier Serra. Acoustic scene classification by fusing lightgbm and vgg-net multichannel predictions. In *Proc. IEEE AASP Challenge Detection Classification Acoust. Scenes Events*, pages 1–4, 2017.
- [23] Kele Xu, Boqing Zhu, Qiuqiang Kong, Haibo Mi, Bo Ding, Dezhi Wang, and Huaimin Wang. General audio tagging with ensembling convolutional neural networks and statistical features. *The Journal of the Acoustical Society of America*, 145(6):EL521–EL527, 2019.
- [24] Jongpil Lee, Jiyoung Park, Keunhyoung Luke Kim, and Juhan Nam. Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. *arXiv preprint arXiv:1703.01789*, 2017.
- [25] Dario Rethage, Jordi Pons, and Xavier Serra. A wavenet for speech denoising. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5069–5073. IEEE, 2018.
- [26] Jisung Wang, Sangki Kim, and Yeha Lee. Speech augmentation using wavenet in speech recognition. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6770–6774. IEEE, 2019.
- [27] Aaron van den Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George van den Driessche, Edward Lockhart, Luis C Cobo, Florian Stimberg, et al. Parallel wavenet: Fast high-fidelity speech synthesis. *arXiv preprint arXiv:1711.10433*, 2017.
- [28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.
- [29] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.