DATA-DRIVEN HARMONIC FILTERS FOR AUDIO REPRESENTATION LEARNING

Minz Won[⋆,†] *Sanghyuk Chun*[§] *Oriol Nieto*[†] *Xavier Serra*[⋆]

[⋆] Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
[†] Pandora Media Inc., Oakland, CA, United States of America
[§] Clova AI Research, NAVER Corp., Seongnam, Republic of Korea

ABSTRACT

We introduce a trainable front-end module for audio representation learning that exploits the inherent harmonic structure of audio signals. The proposed architecture, composed of a set of filters, compels the subsequent network to capture harmonic relations while preserving spectro-temporal locality. Since the harmonic structure is known to have a key role in human auditory perception, one can expect these harmonic filters to yield more efficient audio representations. Experimental results show that a simple convolutional neural network back-end with the proposed front-end outperforms state-of-the-art baseline methods in automatic music tagging, keyword spotting, and sound event tagging tasks.

Index Terms— Harmonic filters, audio representation learning, deep learning

1. INTRODUCTION

With the emergence of deep learning, end-to-end data-driven approaches have become prevalent in audio representation learning [1]. Domain knowledge is often de-emphasized in modern deep architectures and is minimally used in preprocessing steps (e.g., Mel spectrograms). Recent works, with no domain knowledge in their architecture design and preprocessing, reported remarkable results in automatic music tagging [2], voice search [3], and environmental sound detection [4], by using raw audio waveforms directly as their inputs.

Nevertheless, we believe that domain knowledge may facilitate more efficient representation learning, especially when the amount of data is limited [5]. Given that harmonic structure plays a key role in human auditory perception [6], we present a model with a front-end module that can learn compelling representations in a data-driven fashion while forcing the network to employ such harmonic structures. This front-end module, which we call Harmonic filters, is a trainable filter bank [7, 8, 9, 10, 11] that preserves spectro-temporal locality with harmonic structures [12]. Thus, these Harmonic filters aim to bridge the modern assumption-free approaches with the traditional hand-crafted techniques, with the goal to reach a “best of both worlds” scenario.

Contribution. Our contribution is three-fold: (*i*) we propose a versatile front-end module for audio representation learning with a set of data-driven harmonic filters, (*ii*) we show that the proposed method achieves state-of-the-art performance in three different audio tasks, and (*iii*) we present analyses on the parameters of our model that depict the importance of harmonics in audio representation learning.

Organization. The paper is organized as follows: We introduce the Harmonic filters and their architecture design in Section 2. Section 3 describes the tasks and datasets used to assess the Harmonic filters. Section 4 reports experimental results and analyses. Finally, we draw conclusions and discuss future work in Section5.

2. ARCHITECTURE

2.1. Previous Harmonic Representations

The harmonic constant-Q transform (HCQT) [12] is a 3-dimensional representation whose dimensions are *harmonic* (H), *frequency* (F), and *time* (T). By stacking standard constant-Q transform (CQT) representations, one harmonic at a time, the output representation (i.e., HCQT) can preserve the harmonic structure while having spectro-temporal locality. A fully convolutional neural network (CNN) with HCQT inputs could achieve state-of-the-art performance in multi-f0 and melody extraction tasks using several datasets [12].

In our previous work [13], we used two learnable sinc functions (i.e., $\sin(x)/x$) to form each band-pass filter of the first convolutional layer [11], such that the set of harmonics can be learned. By aligning the convolution band-pass filters in each *harmonic*, the first layer outputs an $H \times F \times T$ tensor. When the first harmonic center frequencies are initialized with a MIDI scale, this can be interpreted as an extended, more flexible version of HCQT.

However, the convolution band-pass filter approach to get harmonic spectro-temporal representations requires many convolutions ($H \times F$), including redundant ones (e.g., a 440Hz filter is equivalent to the second harmonic filter of 220Hz). To overcome these efficiency limitations, in this work we replace the convolution band-pass filters of our previous work with an STFT module followed by learnable

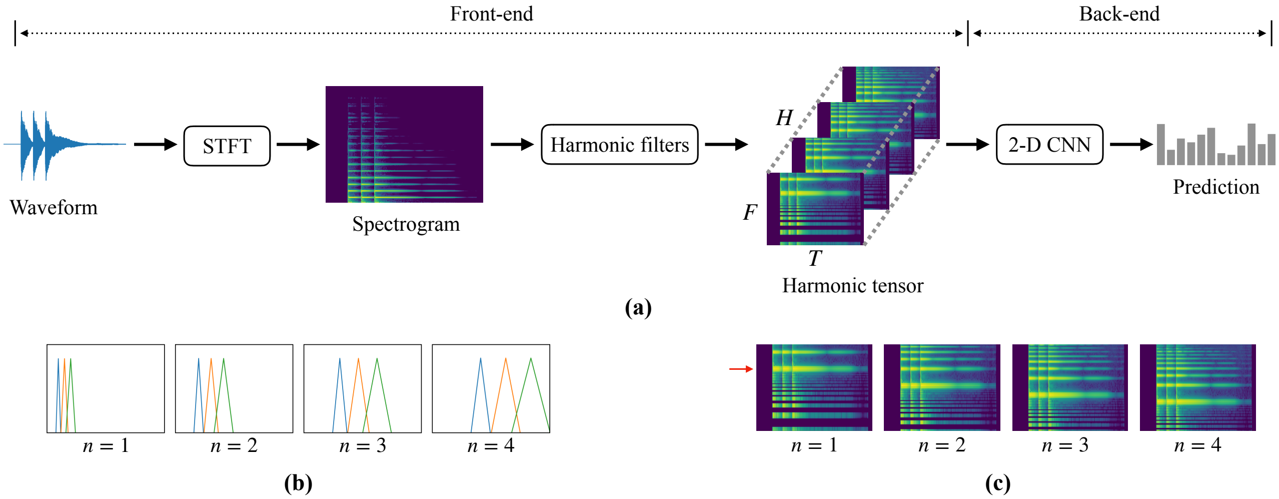

Fig. 1: (a) The proposed architecture using Harmonic filters. The proposed front-end outputs the Harmonic tensor and the back-end processes it depending on the task. The Harmonic filters and the 2-D CNN are data-driven modules that learn parameters during training. (b) Harmonic filters at each harmonic. (c) An unfolded Harmonic tensor. The red arrow indicates the fundamental frequency.

triangular filters, the so-called Harmonic filters.

2.2. Harmonic Filters

First, we formulate a triangular band-pass filter Λ as a function of a center frequency f_c and a bandwidth BW as follows:

$$\Lambda(f; f_c, BW) = \left[1 - \frac{2|f - fc|}{BW}\right]_+, \qquad (1)$$

where $[\cdot]_+$ is a rectified linear function, and f is the frequency bin. Note that when there are multiple triangular band-pass filters with Mel scaled center frequencies, the filter bank performs similarly to the Mel filter bank.[1]

Empirically, the bandwidth BW can be approximated as an affine transform of f_c: $BW \simeq 0.1079 f_c + 24.7$ (equivalent rectangular bandwidth (ERB) [14]). For flexibility's sake, we let the data decide the affine transform with parameters α, β, and Q: $BW = (\alpha f_c + \beta)/Q$.

Now, we define a Harmonic filter Λ_n as follows:

$$\Lambda_n(f; f_c, \alpha, \beta, Q) = \left[1 - \frac{2|f - n \cdot f_c|}{(n \cdot \alpha f_c + \beta)/Q}\right]_+. \qquad (2)$$

The Harmonic filter Λ_n is a triangular band-pass filter of the n-th harmonic of center frequency f_c. Then, our proposed filter bank is defined as a set of Harmonic filters as follows:

$$\{\Lambda_n(f; f_c) \mid n = 1, \ldots, H, f_c \in \{f_c^{(1)}, \ldots, f_c^{(F)}\}\}, \qquad (3)$$

where $f_c^{(i)}$ denotes the i-th center frequency in the first harmonic. Figure 1-(b) shows Harmonic filters with $H = 4$ and $F = 3$. Bandwidths go wider as center frequencies go higher.

Note that, for a given input spectrogram, when $H = 1$ and f_c are initialized with a Mel scale, the filter bank will return an output analogous to the Mel spectrogram. When $H > 1$, we stack the outputs aligned with *harmonic* so that we can have a tensor of dimensionality $H \times F \times T$ as shown in Figure 1-(a). We call this 3-dimensional tensor a Harmonic tensor. Exploiting locality in *time*, *frequency*, and *harmonic* by using this type of representation is advantageous, as discussed in [12]. Furthermore, this Harmonic tensor is flexible since the center frequencies f_c and the bandwidth parameters α, β, Q are all learnable in a data-driven fashion.

2.3. Back-end

Deep networks for audio representation learning can be divided into front-end and back-end: a feature extractor and a classifier, respectively [5]. Figure 1-(a) shows the overview of the proposed architecture. We use an STFT module followed by Harmonic filters as our front-end. For the back-end, a simple conventional 2-D CNN is used since our main goal is to emphasize the advantages of using learnable Harmonic tensors. Harmonics are treated as channels to be fed into the 2-D CNN, thus capturing the harmonic structure through each of its channels. This design choice enforces the convolutional filters to embed harmonic information with locality in time and frequency.

Figure 1-(c) shows an unfolded Harmonic tensor of a 440Hz piano sound. We indicate the fundamental frequency with a red arrow. From left to right, we can see the intensity of the first, second, third, and fourth harmonics at once.

[1] It is not equivalent because Mel filters have asymmetrical triangle shapes.

2.4. Implementation details

First, harmonic center frequencies f_c of the Harmonic tensor are initialized to have a quarter tone interval: $f_c(k) = f_{min} \cdot 2^{k/24}$, where k is the filter index and $f_{min} = 32.7$Hz (C1) is the lowest frequency. The maximum frequency of the first harmonic f_{max} is defined as: $f_{max} = f_s/2H$, where f_s is the sampling rate. After the parameter study, we set the number of harmonics H to 6 for inputs with a 16kHz sampling rate. This results in 128 frequency bins ($F = 128$), with a total of 768 Harmonic filters.

The back-end CNN consists of seven convolutional layers and one fully connected layer to predict the outputs. Each layer includes batch normalization [15] and ReLU nonlinearity. The final activation function is a sigmoid or a softmax, depending on the task. Models are trained for 200 epochs and we choose the best model based on the evaluation metric in the validation set. Scheduled ADAM [16] and stochastic gradient descent (SGD) were used for stable convergence as proposed in [17]. More implementation details and reproducible code are available online.[2]

3. TASKS AND DATASETS

To show the versatility and effectiveness of the Harmonic filters, we experiment with three different tasks: automatic music tagging, keyword spotting, and sound event tagging.

Automatic music tagging. This is a multi-label classification task that aims to predict tags for a given music excerpt. A subset of the MagnaTagATune (MTAT) dataset [21], which consists of ≈26k audio clips, is a widely used set for music tagging. We follow the same data cleaning and split of previous works [2, 20, 17]. This yields ≈21k audio clips with top-50 tags. Area Under Receiver Operating Characteristic Curve (ROC-AUC) and Area Under Precision-Recall Curve (PR-AUC) are used as evaluation metrics following previous literature [5, 20, 17]. Many music tags such as genre, instrumentation, and moods are highly related to the timbre of audio, and harmonic characteristics are crucial for the timbre perception. Hence, one can expect improvements in music tagging by adopting the Harmonic filters in the front-end.

Keyword spotting. MFCC have long been used as input to many speech recognition models because harmonic structure is known to be important for the speech recognition. We believe the Harmonic filters will bring faster convergence and performance improvement than conventional 2-dimensional representations (e.g., CQT, Mel spectrogram). The Speech Commands dataset [22] consists of ≈106k audio samples with 35 command classes (e.g., "yes," "no," "left," "right") for limited-vocabulary speech recognition. Trained models are trivially evaluated with the classification accuracy of choosing one of the 35 classes.

Sound event tagging. The DCASE 2017 challenge [23] used a subset of the AudioSet [24] for the task 4: "large-scale weakly supervised sound event detection for smart cars." It consists of ≈53k audio excerpts with 17 sound event classes, e.g., train horn, car alarm, and ambulance siren. Acoustic events are non-music and non-verbal audio signals, which are expected to have more "inharmonic" characteristics. We are particularly interested in exploring the performance of the proposed model on such audio signals, and thus this task is an ideal candidate for our research. This is also a multi-label classification task and we evaluate it using the average of instance-level F1-scores.

4. EXPERIMENTAL RESULTS

4.1. Comparison with the state of the art

We compare the Harmonic tensor based 2D CNN with the state-of-the-art models of each task. All the experimental results are averaged after three runs. As shown in Table 1, our model outperforms previous results in every task.

In music tagging, we reproduced Musicnn [5] with the same data cleaning and split strategy from others [2, 20] for a fair comparison. As a result, the Mel spectrogram based approach [5] and the raw audio based approach [20] yield comparable results on the MTAT dataset. Our proposed model shows improvements from previous approaches in terms of ROC-AUC and PR-AUC.

As we expected, the keyword spotting accuracy of the proposed model is superior to previous works. Moreover, this showed remarkably fast convergence: the best model according to the validation loss was around 10 epochs while other tasks needed over 100 epochs.

The Harmonic filters were also effective when operating on relatively inharmonic audio signals. We report two different metrics for the DCASE 2017 dataset. F1 (0.1) indicates the F1-score when the threshold of prediction is 0.1, and F1 (opt) is the post threshold optimization score. Note that our model is superior to the state-of-the-art without data balancing or ensembles.

4.2. Parameter study

In this subsection, we provide further understanding of the Harmonic filters by a parameter study and a qualitative analysis on the trained models.

We conduct the parameter study using the MTAT dataset to investigate how the number of harmonics H impacts performance. Table 2 summarizes the results. When $H = 1$, the Harmonic tensor is a 2-dimensional representation like a Mel spectrogram or a CQT, but with frequency bins and bandwidth parameters that are automatically learned and initialized as described in Section 2.4. For 3-dimensional Harmonic tensors ($H > 1$), performance improves as the model uses more

[2] https://github.com/minzwon/data-driven-harmonic-filters

Methods	Music Tagging		Keyword Spotting	Sound Event Tagging	
	MTAT		Speech Commands	DCASE 2017	
	ROC-AUC	PR-AUC	Accuracy	F1 (0.1)	F1 (opt)
Musicnn [5]	0.9089*	0.4503*	-	-	-
Attention RNN [18]	-	-	0.9390	-	-
Surrey-cvssp [19]	-	-	-	-	0.5560
Sample-level [2]	0.9054	0.4422	0.9253	0.4213	-
+ SE [20]	0.9083	0.4500	0.9395	0.4582	-
+ Res +SE [20]	0.9075	0.4473	0.9482	0.4607	-
Proposed	**0.9141**	**0.4646**	**0.9639**	**0.5468**	**0.5824**

Table 1: Performance comparison with state-of-the-art. The numbers are averaged across 3 runs. '*' denotes reproduced result with our data split. F1 (0.1) and F1 (opt) denote F1-score measured by threshold value of 0.1 and optimized one, respectively.

H	1	2	3	4	5	6	7*
ROC-AUC	0.9132	0.9115	0.9118	0.9118	0.9129	0.9141	**0.9146**
PR-AUC	0.4599	0.4541	0.4550	0.4555	0.4562	**0.4646**	0.4617

Table 2: The effect of number of Harmonics (H) on MTAT. '*' has a different size of max pooling due to the smaller F.

Options	512 FFT	256 FFT	Quarter tone	Semi tone
Q(MTAT)	2.1386	1.9537	2.1386	1.8447
Q(Speech Commands)	1.9032	1.9983	1.9032	1.8451
Q(DCASE 2017)	1.9040	1.8762	1.9040	1.8460

Table 3: Trained bandwidth parameter Q in different settings.

harmonics. Note that, as we described in Section 2.4, the frequency range in the first harmonic becomes narrower as the number of harmonics H increases ($f_{max} = f_s/2H$). We hypothesize that this is the reason why there is a slight performance drop between $H = 1$ and $H = 2$. However, much larger H might yield worse results. If $H = 10$ for example, the maximum frequency of the first harmonic becomes 800Hz, which means the Harmonic tensor cannot include the harmonic information of higher pitches, i.e., fundamental frequencies higher than 800Hz.

We also tried to determine the role of learnable center frequencies f_c but we could not find significant differences between learnable and fixed center frequencies. Their performance gaps in three different tasks are all in the range of performance variance. In our experimental setup using quarter tone MIDI scale, there is no observable benefit of using learnable center frequencies f_c.

Finally, we show the role of the bandwidth parameter Q. In this experiment, we used fixed values of α and β with the empirical values [14] and only let Q to be trained. As we mentioned in Section 2.2, the Harmonic tensor is more flexible than HCQT since this parameter does not need to be heuristically set. In Table 3, the bandwidth parameter Q changes based on task, FFT size, and center frequency interval. This proves that the optimal parameter Q is task- and settings-dependent, thus showing the importance of automatically learning it in a data-driven manner.

5. CONCLUSION

In this paper, we introduced data-driven Harmonic filters to form a versatile front-end for audio representation learning. Experimental results report state-of-the-art performance in automatic music tagging, keyword spotting, and sound event tagging tasks. The output of the proposed front-end keeps locality in time, frequency, and harmonic so that the subsequent back-end can explicitly capture harmonic structures. The proposed front-end is flexible since it learns bandwidth parameters in a data-driven fashion. To further scrutinize the representation ability of the proposed model, other complex tasks beyond binary classification should be considered. Analyzing how well this model scales with larger datasets would also be key to better understand the potential of the proposed architecture. Finally, interpretability studies and additional investigation on the learnable parameters of the model may yield valuable insights in terms of how to more optimally apply these Harmonic filters.

6. ACKNOWLEDGEMENT

This work was funded by the predoctoral grant MDM-2015-0502-17-2 from the Spanish Ministry of Economy and Competitiveness linked to the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502).

7. REFERENCES

[1] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al.,

"Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[2] Jongpil Lee, Jiyoung Park, Keunhyoung Luke Kim, and Juhan Nam, "Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms," *the 14th Sound & Music Computing Conference*, 2017.

[3] Tara N Sainath, Ron J Weiss, Andrew Senior, Kevin W Wilson, and Oriol Vinyals, "Learning the speech front-end with raw waveform cldnns," in *the 16th Annual Conference of the International Speech Communication Association*, 2015.

[4] Wei Dai, Chia Dai, Shuhui Qu, Juncheng Li, and Samarjit Das, "Very deep convolutional neural networks for raw waveforms," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[5] Jordi Pons, Oriol Nieto, Matthew Prockup, Erik Schmidt, Andreas Ehmann, and Xavier Serra, "End-to-end learning for music audio tagging at scale," *International Society for Music Information Retrieval Conference*, 2018.

[6] William A Sethares, *Tuning, timbre, spectrum, scale*, Springer Science & Business Media, 2005.

[7] Shrikant Venkataramani, Jonah Casebeer, and Paris Smaragdis, "Adaptive front-ends for end-to-end source separation," in *the 31st Conference on Neural Information Processing Systems*, 2017.

[8] Hiroshi Seki, Kazumasa Yamamoto, and Seiichi Nakagawa, "A deep neural network integrated with filterbank learning for speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[9] Monika Dörfler, Thomas Grill, Roswitha Bammer, and Arthur Flexer, "Basic filters for convolutional neural networks applied to music: Training or design?," *Neural Computing and Applications*, 2018.

[10] Daiki Takeuchi, Kohei Yatabe, Yuma Koizumi, Yasuhiro Oikawa, and Noboru Harada, "Data-driven design of perfect reconstruction filterbank for dnn-based sound source enhancement," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

[11] Mirco Ravanelli and Yoshua Bengio, "Speaker recognition from raw waveform with sincnet," in *IEEE Spoken Language Technology Workshop*, 2018.

[12] Rachel M Bittner, Brian McFee, Justin Salamon, Peter Li, and Juan Pablo Bello, "Deep salience representations for f0 estimation in polyphonic music.," in *International Society for Music Information Retrieval Conference*, 2017.

[13] Minz Won, Sanghyuk Chun, Oriol Nieto, and Xavier Serra, "Automatic music tagging with harmonic cnn," in *Late Breaking Demo in the International Society for Music Information Retrieval Conference*, 2019.

[14] Brian R Glasberg and Brian CJ Moore, "Derivation of auditory filter shapes from notched-noise data," *Hearing research*, vol. 47, no. 1-2, pp. 103–138, 1990.

[15] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *the 32nd International Conference on Machine Learning*, 2015.

[16] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 2015.

[17] Minz Won, Sanghyuk Chun, and Xavier Serra, "Toward interpretable music tagging with self-attention," *arXiv preprint arXiv:1906.04972*, 2019.

[18] Douglas Coimbra de Andrade, Sabato Leo, Martin Loesener Da Silva Viana, and Christoph Bernkopf, "A neural attention model for speech command recognition," *arXiv preprint arXiv:1808.08929*, 2018.

[19] Yong Xu, Qiuqiang Kong, Wenwu Wang, and Mark D Plumbley, "Surrey-cvssp system for dcase2017 challenge task4," *Detection and Classification of Acoustic Scenes and Events*, 2017.

[20] T. Kim, J. Lee, and J. Nam, "Comparison and analysis of samplecnn architectures for audio classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, May 2019.

[21] Edith Law, Kris West, Michael I Mandel, Mert Bay, and J Stephen Downie, "Evaluation of algorithms using games: The case of music tagging.," in *International Society for Music Information Retrieval Conference*, 2009.

[22] Pete Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[23] Annamaria Mesaros, Toni Heittola, Aleksandr Diment, Benjamin Elizalde, Ankit Shah, Emmanuel Vincent, Bhiksha Raj, and Tuomas Virtanen, "Dcase 2017 challenge setup: Tasks, datasets and baseline system," in *Detection and Classification of Acoustic Scenes and Events*, 2017.

[24] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.