

Music-PAW: Learning Music Representations via Hierarchical Part-whole Interaction and Contrast

Anonymous Author(s)*

ABSTRACT

The excellent performance of recent self-supervised learning methods on various downstream tasks has attracted great attention from academia and industry. Some recent research efforts have been devoted to self-supervised music representation learning. Nevertheless, most of them learn to represent equally-sized music clips in the waveform or a spectrogram. Despite being effective in some tasks, learning music representations in such a manner largely neglect the inherent part-whole hierarchies of music. Due to the hierarchical nature of the auditory cortex [24], understanding the bottom-up structure of music, i.e., how different parts constitute the whole at different levels, is essential for music understanding and representation learning. This work pursues hierarchical music representation learning and introduces the Music-PAW framework, which enables feature interactions of cropped music clips with part-whole hierarchies. From a technical perspective, we propose a transformer-based part-whole interaction module to progressively reason the structural relationships between part-whole music clips at adjacent levels. Besides, to create a multi-hierarchy representation space, we devise a hierarchical contrastive learning objective to align part-whole music representations in adjacent hierarchies. The merits of audio representation learning from part-whole hierarchies have been validated on various downstream tasks, including music classification (single-label and multi-label), cover song identification and acoustic scene classification.

CCS CONCEPTS

- Computer systems organization → Embedded systems; Redundancy; Robotics;
- Networks → Network reliability.

KEYWORDS

music representation, pre-training, part-whole hierarchy

ACM Reference Format:

Anonymous Author(s). 2018. Music-PAW: Learning Music Representations via Hierarchical Part-whole Interaction and Contrast. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 11 pages. <https://doi.org/XXXXXXX.XXXXXXX>

1 INTRODUCTION

The availability of massive annotated data has contributed to the tremendous success of deep neural networks in various research

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

<https://doi.org/XXXXXXX.XXXXXXX>

fields. However, such labeled data are scarce and often difficult to acquire in many cases. Recent advances in self-supervised learning methods, such as transformer-based [23, 40, 57, 62, 68, 74] and contrastive learning methods [6, 9, 13, 21, 27, 32, 56, 73], have achieved excellent performance in learning generic representations from unlabeled data, transferable to various language and vision downstream tasks. In the music domain, some early attempts explore transferring the success of self-supervised learning to music representation learning. For instance, CLMR [53] and COLA [49] investigate whether a mature visual contrastive learning method, SimCLR [9], can benefit music self-supervised learning. Inspired by another successful visual representation learning framework BYOL [21], BYOL-A [45] directly pulls the representations of equally-sized clips from the same audio closer in the representation space.

Despite significant progress made by existing work, how to exploit the inherent part-whole structure of musical audio in representation learning remains an open research problem. A part-whole hierarchy refers to the structure of how different *part* music clips constitute a *whole* music clip. For example, when a people listen to a piece of music, the music clips containing prelude, verse, and chorus will affect his cognition and understanding of the whole music audio. Existing work [1, 45, 49, 53, 60] crops audios into fixed-size clips, and learn the semantic correlations of audios at the same hierarchy, largely neglecting the part-whole structures for audio understanding.

Substantial evidence has been discovered from the human visual cortex [24] that humans parse visual scenes into part-whole hierarchies with many different levels [17, 28, 55]. Likewise, hierarchical signal processing also exists in the auditory cortex [51]. Without learning such part-whole hierarchies might make the audio representations less functional in real-world applications. For example, cover song identification [71, 72] requires obtaining the similarity of two songs with different lengths, and multi-label music classification [65] requires the different music clips reflecting different labels. Hence, it is essential to represent music with part-whole hierarchies, which has not been explored before in self-supervised music representation learning.

In this paper, we set the goal of learning music representations with part-whole hierarchies. To achieve this, we propose a novel self-supervised music representation learning framework Music-PAW, i.e., Music PArt-Whole learning. Technically, Music-PAW comprises three key components: the hierarchical audio cropping, the part-whole interaction module, and the hierarchical contrastive learning objective. Our **contributions** are:

- (i) We design a hierarchical audio cropping strategy progressively crops a music audio into multiple sub-clips and the devised part-whole interaction module simultaneously infers the structural relationships between part-whole clips at different levels; At the **input** part of Music-PAW, a musical audio clip is the sub-clip of the last hierarchical level at each hierarchical level, which results in part-whole clips

with a hierarchy. At the **architecture** part of Music-PAW, the model can better capture the structure of how different part music clips constitute the whole music clip at each hierarchy.

- (ii) We propose a hierarchical contrastive learning objective which pulls part-whole representations in adjacent hierarchies closer in the latent representation space; This objective helps Music-PAW create a multi-hierarchy representation space, where we reach a consensus on the part-whole music representations at each hierarchy.
- (iii) We validate the benefits of music representations learned with part-whole hierarchies on various music downstream tasks, including music genre classification [11, 47, 53, 65] (MagnaTagATune [37] and GTZAN [54]) and cover song identification [67, 71, 72] (SHS100K [16] and Covers80 [31]). Besides, to evaluate if our method is general, we conduct experiments on acoustic scene classification [35, 36] (DCASE2016 [42]). We show that the same pretrained model successfully generalizes all our downstream tasks. Our ablation studies demonstrate the effectiveness of the Music-PAW framework and the rationality of our analyses.

2 RELATED WORK

2.1 Self-supervised Learning.

Self-supervised Learning (SSL) targets learning representations from the vast unlabeled data through hand-crafted pretext tasks. For instance, context prediction [13, 14], image jigsaw puzzle [46], predicting the image's rotation [19], and reconstruction [3, 5, 15, 20]. Although these methods have demonstrated their validity, the learned representations lack generalization. The recent contrastive learning methods [7, 9, 12, 21, 22, 26, 27, 29, 32, 43, 56, 73] have made much progress in self-supervised learning. For example, MoCo [26] views contrastive learning as a dictionary look-up to build a dynamic queue, including samples of the current and previous mini-batch. Another method SimCLR [9], is a simple framework for contrastive learning without a memory bank. BYOL [21] proposed a new paradigm for contrastive learning, which only used positive samples while casting away negative samples.

2.2 Music and Audio Representation Learning.

Benefiting from contrastive self-supervised learning, which has been applied to learn visual representations [9, 21, 26, 30, 48, 69] and has achieved outstanding performance, some works [1, 45, 49, 53, 60] with regard to using contrastive learning methods to learn music and audio representations have emerged recently. CLMR [53], CLAR [1], COLA [49] and Multi-Format[60] expand their work in SimCLR [9] contrastive framework to learn audio representations. Another contrastive paradigm, BYOL [21], which casts away the negative pairs during the pre-training stage, has also been employed to obtain auditory representations by BYOL-A [45]. Apart from these works, wave2vec [50] and wave2vec 2.0 [2] have explored how to obtain the representation of speech.

2.3 Part-whole Hierarchies.

Several studies have attempted to capture part-whole hierarchies for visual representation. [58] is the first attempt to parse an image

into a part-whole hierarchy. Recently, Hinton [28] has proposed an imaginary system GLOM, which answers how a neural network can parse an image into a part-whole hierarchy. After that, ViP [55] is proposed to divide visual representations into part-level and whole-level. Likewise, [17] devises a framework capable of providing a representation of part-whole hierarchies from visual cues. Hip [8] introduces a sequence of splitting-processing-merging blocks to process information at different hierarchical levels.

3 METHODOLOGY

Our overall framework presents in Figure 1. We design a *hierarchical audio cropping* strategy to partition a long audio clip. Afterward, part-whole audio clip pairs in different hierarchies interact internally through the multi-head attention mechanism in the transformer encoder [59] within *part-whole transformer block*. Lastly, *hierarchical contrastive learning* is to align part-whole audio representations in adjacent hierarchies.

3.1 Hierarchical Audio Cropping.

In order to model music's hierarchical structure information, which can help the network understand how different music clips comprise the whole music clip, we devise a hierarchical audio cropping strategy. Specifically, we partition the input musical audio into M (we set $M=2$) music segments with a same time span. We then split the above obtained music segments into M music segments with a shorter length similarly. We repeat the process until getting N (we set $N=4$) kinds of the size of music segments. The *hierarchical audio cropping* strategy guarantees the content of long music clips can be distributed in their sub-clips, making their information match each other during the interaction. For a better reading, when mentioning the music clips together with its sub-clips, we uniformly call the longer music clips as long clips.

3.2 Part-whole Transformer Block

It is hard for the network to learn part-whole hierarchies of music directly since the long clip and its sub-clips are a simple physical contains relationship. It is challenging for the network to understand how the representations of sub-clips constitute the representation of long clip, *i.e.*, the part-whole relationship. Therefore, we interact the long clips with their sub-clips through the transformer encoders. Specifically, $S^n = \{s_m^n | n \in [0, N-1], m \in [0, M^n]\}$ represents the musical segment set with various size from a music file. The s_m^n denotes the m th music segment in the S^n set. A larger superscript of the s_m^n represents shorter music clips. We use the encoder to get $f(s_m^n)$ to project them into a shared latent space. For simplicity, we denote $f(s_m^n)$ as s_m^n as following.

Interaction Unit. To improve understanding, we use a (whole, part) pair as an instance to describe this module, as shown in Figure 2. Specifically, we select an music clip s_m^n from S^n and its sub-clips $\tilde{s}^{n+1} = \{s_m^{n+1} | m \in [0, M]\}$ from S^{n+1} . After encoding by encoder $f(\cdot)$, We denote them in the form of matrix as $s_m^n \in \mathbb{R}^{1 \times D_e}$ and $\tilde{s}^{n+1} \in \mathbb{R}^{M \times D_e}$, respectively. D_e is the feature dimension of the output of the encoder. We first obtain $Q^n, K^n, V^n \in \mathbb{R}^{1 \times D_t}$ by apply the linear transformation $W_Q^n, W_K^n, W_V^n \in \mathbb{R}^{D_e \times D_t}$ on s_m^n . Likewise, we employ the linear projection $W_Q^{n+1}, W_K^{n+1}, W_V^{n+1} \in \mathbb{R}^{D_e \times D_t}$

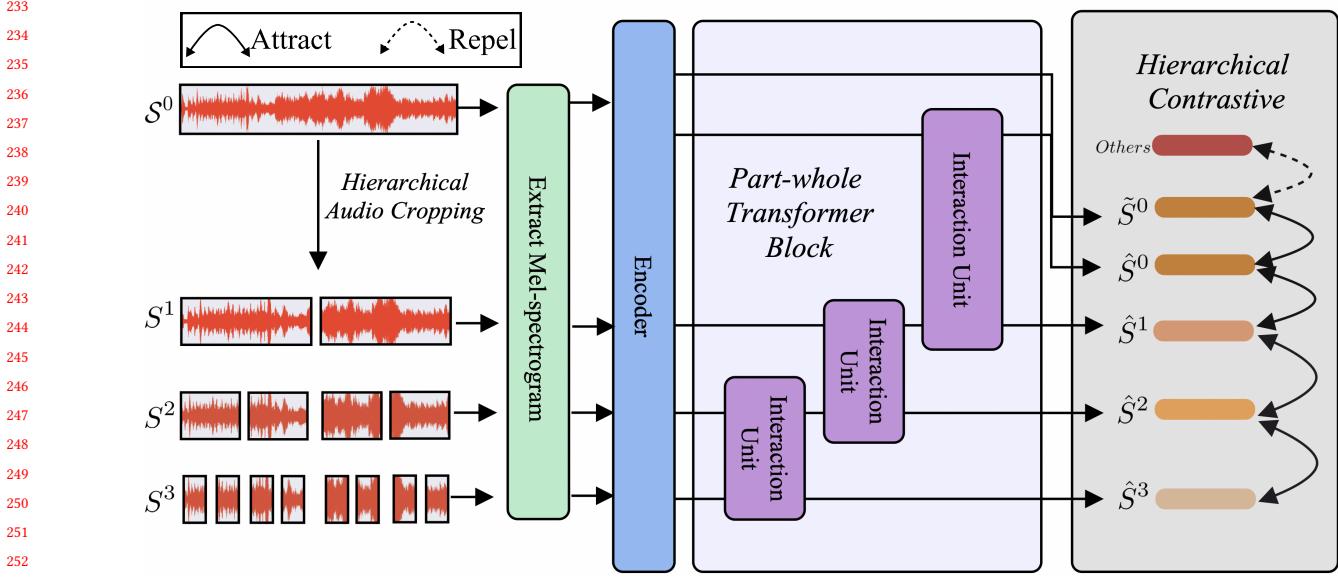


Figure 1: The over architecture of our proposed method. Our method mainly includes hierarchical audio cropping, part-whole transformer block, and hierarchical contrastive learning.

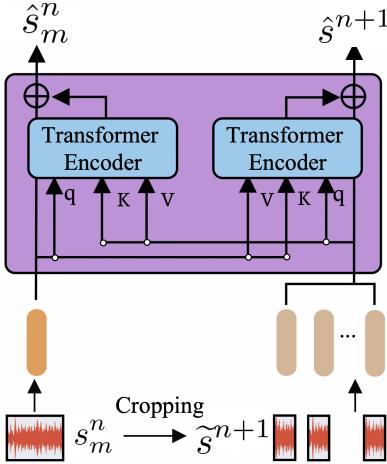


Figure 2: Internal structure of the interaction unit.

on \tilde{s}^{n+1} to get $Q^{n+1}, K^{n+1}, V^{n+1} \in \mathbb{R}^{M \times D_t}$. We then input the Q^n, K^{n+1}, V^{n+1} as *Query*, *key*, *Value* into the transformer encoder. The output, i.e., interacted information, of the transformer encoder will be injected into the original s_m^n ,

$$\hat{s}_m^n = s_m^n + \lambda^n \text{Transformer}(Q^n, K^{n+1}, V^{n+1}) \quad (1)$$

where λ^n is a hyper-parameter to control the importance of the interacted result, $\text{Transformer}(\cdot)$ denotes the network of transformer encoder, which consists of the multi-head attention sub-layer and position-wise feed-forward sub-layer. In this situation, the long clip is viewed as a query vector so that the whole-level information of the sub-clips could be encoded into the long clip through the attention mechanism. In the same way, we embed the interacted

information into s^{n+1} ,

$$s^{n+1} = \tilde{s}^{n+1} + \lambda^{n+1} \text{Transformer}(Q^{n+1}, K^n, V^n) \quad (2)$$

where λ^{n+1} is a hyper-parameter to trade off the importance of the interacted result. In this case, the sub-clips are viewed as query vectors so that the part-level information of the long clips could be distributed to the sub-clips. Other music clips of the set S^n can all interact with its corresponding sub-clips. We note that the music clips with the middle size, such as S^1 and S^2 in Figure 1, will interact two times: the first time's output will be the second time's input.

3.3 Hierarchical Contrastive Learning

The outputs from *part-whole transformer block* will contrast with each other in a hierarchical contrastive learning space. Following the previous methods [9, 21], we use an MLP to project all related vectors into a contrastive learning latent space. For better readability, the symbol of the outputs from the encoder and transformer block represents the vectors after projection. To align part-whole audio representations in adjacent levels, we pair a variety of hierarchical part-whole pairs (*whole*, *part*) which can be denote as $P = \{(\hat{S}^0, \hat{S}^1), (\hat{S}^1, \hat{S}^2), \dots, (\hat{S}^n, \hat{S}^{n+1}), \dots\}$ and $\hat{S}^n = \{\hat{s}_m^n | m \in [0, M^{n-1}]\}$. The letter with a hat above represents the output of *part-whole transformer block*. For each audio file in dataset, we can obtain $N - 1$ pairs. We first compute the contrastive loss of different part-whole pairs of an music file,

$$\mathcal{L}_b^{pw} = \sum_{(\hat{S}^i, \hat{S}^{i+1})}^P \lambda^i \left(\sum_{\hat{s}_b^i}^{\hat{S}^i} \sum_{\hat{s}_b^{i+1}}^{\hat{U}_i} \exp(sim(\hat{s}_b^i, \hat{s}_b^{i+1})) / \tau \right) \quad (3)$$

$$\lambda_i = \text{len}(\hat{s}_b^{i+1}) / \text{len}(\hat{s}_b^i) \quad (4)$$

where i represents the i th level in part-whole hierarchies. The \hat{U}_i is a set contains sub-clips which is cropped from \hat{s}_b^i , and simultaneously all exist in \hat{S}^{i+1} . λ^i is a weight to trade off the importance of the corresponding part-whole pair and $len(\cdot)$ is a function to get the time span of the corresponding music segments, *i.e.*, the more the two music clips differ in length, the less their loss term effect. $sim(\cdot)$ is to compute the similarity of two target vectors. τ denotes a temperature parameter. Subscript b of \mathcal{L}_b^{pw} is the index of this audio in current batch. Note that our strategy is different from [30] in essence. We sample the music clips with various lengths, and the short music clip is a part of its corresponding long clip. The two sampled audio clips from the paper [30] are near in temporal and are of the same length.

We then integrate the \mathcal{L}_b^{pw} into the common used contrastive loss NT-Xent loss[9, 52, 66] to constitute our final hierarchical contrastive learning objective \mathcal{L}_b^{hc} for regularizing the part representations and whole representations in every hierarchy,

$$\mathcal{L}_b^{hc} = -\log \frac{\mathcal{L}_b^{pw} + \exp(sim(s_b^0, \tilde{s}_b^0)/\tau)}{\mathcal{L}_b^{pw} + \mathcal{L}_{neg}} \quad (5)$$

$$\mathcal{L}_{neg} = \sum_{u=1}^B \exp(sim(s_b^0, \tilde{s}_u^0)/\tau) + \sum_{u=1}^B \mathbb{1}_{u \neq b} \exp(sim(s_b^0, \tilde{s}_u^0)/\tau) \quad (6)$$

where $\mathbb{1}_{u \neq b} \in \{0, 1\}$ is an indicator function evaluating to 1 if $u \neq b$ and B is batch size. Since the set S_b^n only has one element if n equals zero, we denotes $S_b^0 = \{s_b^0\}$. $sim(\cdot)$ is to compute the similarity of two target vectors. We only use another augmented view of the long clip at the top hierarchy as an additional positive sample. Negative samples consist of the interacted view \tilde{s}_u^0 and the augmented view \tilde{s}_u^0 , which get from other \mathcal{S}^0 within the current batch.

4 EXPERIMENTS

4.1 Experimental Settings

Datasets. We evaluate the pre-trained models on music classification (MTAT [37] and GTZAN [54]), on cover song identification (SHS100K [16] and Covers80 [31]) and on acoustic scene classification (DCASE2016 [42]). **Note** that We use the subset of SHS100K–SHS100K-SUB. We crawl raw audios through youtube-dl¹ using the provided URLs from Github². However, many audio files can not be acquired due to the invalid copyright, resulting in many differences between the previous and our downloaded versions. We randomly select 1,088 songs and all cover versions to construct SHS100K-SUB, containing 12K audio files.

Setup. Following the procedure of [9, 21, 26], we first train a linear classifier with the training data from MTAT on the top of the frozen encoder, which is pre-trained with the same dataset. Afterward, following the semi-supervised learning setup of [4, 9, 21], we randomly select 1% and 10% labeled training data of MTAT. We then fine-tune the whole base network on the selected subsets. Moreover,

¹<https://github.com/ytdd-org/youtube-dl>

²<https://github.com/NovaFrost/SHS100K2>

we transfer our learned audio representations to other classification dataset, namely GTZAN [54], and to other audio-related tasks, namely acoustic scene classification and cover song identification. In order to avoid the randomness of experiments, all results reported in our paper are average of 5 separate runs.

Metrics. We use Accuracy, ROC-AUC, and PR-AUC as metrics for GTZAN evaluation. Unlike GTZAN, a single-label dataset, MTAT is a multi-label dataset resulting in little significance of Accuracy in evaluating it. Hence, like [53, 65], only ROC-AUC and PR-AUC are applied as its metrics. Following the previous work [36, 71, 72], we use MAP, Precision@10, and MR1 as the evaluation indicators of SHS100K-SUB and Covers80 and use Accuracy as DCASE2016's metric.

Implementation Details Before inputting music data into the encoder, We utilize *torchaudio*³ tool to extract the mel-spectrogram of audio data. Following the setup in COLA [49], we transform the mel-spectrogram to the logarithmic scale. Note that all music clips of various sizes are compressed to the identical-size log-mel spectrogram through various hop lengths. Therefore, we can use only one encoder to project the log-compressed mel-spectrogram into a latent space with uniform dimensionality. We use the same group of data augmentations as in CMLR [53]. The group includes polarity inversion, noise, gain, filter, delay, and pitch shift. Each augmentation is randomly selected accordingly to its setting probability. We use FCN-4 [10] and Short-chunk CNN [65] (FCN-7) for our encoder, respectively. Their distinction is the number of the neural network. More detail can be found in their articles. The encoder outputs a 512-dimension feature as a representation. The transformer encoders [59] we used both have 3 layers, and their multi-head attention sub-layer has 3 heads. An *multi-layer perceptron*(MLP) as projection head is used to map the representation to a contrastive space. This MLP consists of a linear layer with output size 512 followed by rectified linear units (ReLU) [44], and a final linear layer with output dimension 256. We use the Adam optimizer [34]. The learning rate is 0.0003, and weight decay is 1.0×10^{-6} during the pre-training stage. Others are default. We employ an early stopping mechanism for the linear evaluation and fine-tuning stages when the validation scores do not improve for 10 epochs. We set the batch size to 48 and train for 300 epochs, taking about 100 hours on 2 RTX3090 GPUs. At the spectrogram extracting, the hop size is 128 during time-frequency transformation. STFT is performed using 256-point FFT while the number of mel-bands is 128.

4.2 Experimental Evaluation

Linear evaluation of multi-label classification on MTAT. MTAT is a benchmark of music tagging [10, 38, 47, 65] which is a multi-label binary classification task that aims to predict relevant tags for a given song. We first evaluate Music-PAW's representations by training a linear classifier based on the frozen representations, following the procedure described in [9, 21, 26]. We report the ROC-AUC and PR-AUC in % on the test set of MTAT in table 1. We pre-train the unsupervised methods with two backbones, FCN-4 and FCN-7. Besides, we show several supervised methods for music

³<https://github.com/pytorch/audio>

Table 1: Comparison between Music-PAW and various baselines evaluated on a multi-label music classification dataset MTAT. Self-supervised models are with the linear evaluation protocol. The best results of self-supervised methods are in bold.

Method	Backbone	Param(M)	GFLOPs	ROC-AUC	PR-AUC
<i>Supervised:</i>					
CRNN [11]	FCN-4 + GRU	0.39	242.34	86.1	28.8
FCN-5 [10]	-	0.45	48.65	89.1	35.6
FCN-4 [10]	-	0.37	8.93	89.7	37.1
Musicnn [47]	-	0.78	19.40	90.2	37.3
SampleCNN [39]	-	1.85	27.97	90.0	36.8
SampleCNN+SE [33]	SE-SampleCNN	6.93	78.52	90.1	37.3
Self-attention [64]	SampleCNN + Transformer	10.5	214.72	90.4	37.6
Harmonic CNN [63]	-	3.60	68.19	90.9	39.0
Short-chunk CNN + Res [65]	ResNet	12.1	44.56	90.5	38.4
Short-chunk CNN [65]	FCN-7	3.67	32.44	90.6	38.9
<i>Self-supervised:</i>					
CLMR [63]	SampleCNN	1.85	27.97	89.3	36.0
COLA [49]	FCN-4	0.37	8.93	88.7	34.8
Multi-Format [61]	FCN-4	0.37	8.93	88.9	35.4
BYOL-A [45]	FCN-4	0.37	8.93	89.1	35.8
Music-PAW	FCN-4	0.37	8.93	89.4	36.4
Multi-Format [61]	FCN-7	3.67	32.44	87.0	31.3
COLA [49]	FCN-7	3.67	32.44	89.1	36.0
BYOL-A [45]	FCN-7	3.67	32.44	89.2	36.3
Music-PAW	FCN-7	3.67	32.44	89.5	36.7

Table 2: Semi-supervised training using 1% and 10% MTAT labeled data. Results for the supervised baseline are from [10] and [65]. Best performance are in bold.

Method	Backbone	ROC-AUC		PR-AUC	
		1%	10%	1%	10%
Supervised	FCN-4	75.4	85.6	20.1	30.5
	FCN-7	70.6	86.4	14.3	30.7
CLMR [63]	SampleCNN	78.5	86.8	25.0	32.6
BYOL-A [45]	FCN-4	79.0	87.0	24.9	32.0
Multi-Format [61]	FCN-4	79.5	86.8	25.5	31.8
COLA [49]	FCN-4	79.7	87.2	25.1	32.3
Multi-Format [61]	FCN-7	78.7	87.2	23.5	32.3
BYOL-A [45]	FCN-7	79.7	87.6	26.3	32.9
COLA [49]	FCN-7	80.5	87.4	26.6	32.7
Music-PAW	FCN-4	80.6	87.6	25.9	32.8
Music-PAW	FCN-7	81.2	88.0	27.0	34.1

tagging according to [65]. With a standard FCN-4, Music-PAW obtains 89.4% ROC-AUC (36.4% PR-AUC), which is a 0.3%(respective 0.6%) improvement over the previous self-supervised state-of-the-art. With deeper architectures with more parameters, Music-PAW achieves more advanced results and consistently outperforms the

previous self-supervised methods. Although they are still significantly below the supervised models, Music-PAW tightens the gap with respect to the supervised methods of [65] and [10].

Semi-supervised training on MTAT. We are interested in whether music representations with part-whole hierarchies can benefit downstream training with a small amount of labeled data. Towards this end, we conduct self-supervised training on the unlabeled training data of MTAT and fine-tune the backbone model with 1% and 10% labeled training data [4, 9, 21], *i.e.*, semi-supervised training. Besides various self-supervised learning baselines, we also incorporate some supervised baselines, *i.e.*, FCN-4 and FCN-7 with kaiming initialization [25] for comparison and train them using the same labeled subset. We report the results of models on the MTAT test set, as shown in Table 2. Compared with supervised baseline, self-supervised methods substantially surpass the supervised baseline *w.r.t.* either 1% or 10% labeled data, which demonstrates the merits of self-supervised music representation learning. Compared with existing self-supervised learning baselines, Music-PAW further brings performance improvement. Specifically, *w.r.t* base classification architecture FCN-4 and FCN-7 with 1% labeled audio data, Music-PAW obtain respectively 0.9% and 0.7% improvement on ROC-AUC over the state-of-the-art self-supervised baselines. When fine-tuning FCN-4 and FCN-7 with 10% labeled music data, Music-PAW obtain respectively 0.5% and 1.2% improvement on PR-AUC.

581 **Table 3: Out-of-distribution evaluation on music genre classification dataset GTZAN. Self-supervised fine-tuning methods,**
 582 **including Music-PAW, are pre-trained on unlabeled data of MTAT and fine-tuned on the labeled data of GTZAN. Best results**
 583 **are in bold, the second best results are underlined.**

585 Method	586 Backbone	587 Accuracy	588 ROC-AUC	589 PR-AUC
<i>590 Supervised:</i>				
591 CRNN [11]	592 FCN-4 + GRU	593 54.8	594 86.1	595 52.7
596 FCN-5 [10]	597 -	598 64.8	599 93.3	600 71.3
601 FCN-4 [10]	602 -	603 57.9	604 91.5	605 62.5
606 Musicnn [47]	607 -	608 59.0	609 91.8	610 66.3
611 SampleCNN [39]	612 -	613 56.6	614 90.7	615 63.8
616 SampleCNN+SE [33]	617 SE-SampleCNN	618 57.9	619 90.1	620 61.5
621 Self-attention [64]	622 SampleCNN + Transformer	623 54.5	624 86.7	625 54.2
626 Harmonic CNN [63]	627 -	628 60.3	629 93.5	630 70.9
631 Short-chunk CNN + Res [65]	632 ResNet	633 55.2	634 88.1	635 59.7
636 Short-chunk CNN [65]	637 -	638 64.1	639 94.1	640 71.0
<i>641 Supervised fine-tuning:</i>				
642 Supervised-MTAT	643 FCN-4	644 59.3	645 92.1	646 65.5
647 Supervised-MTAT	648 FCN-7	649 64.1	650 92.9	651 69.5
<i>652 Self-supervised fine-tuning:</i>				
653 BYOL-A [45]	654 FCN-4	655 57.6	656 92.6	657 65.6
658 Multi-Format [61]	659 FCN-4	660 59.7	661 92.3	662 65.5
663 COLA [49]	664 FCN-4	665 56.2	666 92.7	667 65.4
668 Music-PAW	669 FCN-4	670 63.8	671 93.2	672 66.8
673 Multi-Format [61]	674 FCN-7	675 62.8	676 93.2	677 71.2
678 BYOL-A [45]	679 FCN-7	680 63.4	681 <u>94.6</u>	682 <u>74.8</u>
685 COLA [49]	686 FCN-7	687 <u>65.5</u>	688 94.2	689 72.6
693 Music-PAW	694 FCN-7	695 71.7	696 94.7	697 77.2

615 **Out-of-distribution Classification Evaluation.** Note that in the
 616 above evaluation, the downstream training set and the pre-training
 617 set are from the same dataset, *i.e.*, MTAT, and share similar data
 618 distributions. We are interested in whether music representations
 619 with part-whole hierarchies could boost downstream training on
 620 out-of-distribution data samples. Towards this end, we adopt an-
 621 other music classification dataset GTZAN [18, 75], where both
 622 the features and the labels are different from MTAT. We conduct
 623 self-supervised learning on the whole training set of MTAT with un-
 624 labeled data, fine-tune the backbones on the training set of GTZAN
 625 with labeled data, and evaluate them on GTZAN’s test set. We de-
 626 note these models as *self-supervised fine-tuning*, which include the
 627 proposed Music-PAW. We consider two kinds of backbone archi-
 628 tectures, *i.e.*, FCN-4 and FCN-7. Similar to the above evaluation,
 629 we incorporate various *supervised* baselines, which are trained on
 630 the labeled data of GTZAN. In addition, we consider a *supervised*
 631 *fine-tuning* baseline Supervised-MTAT, which is firstly trained on
 632 the labeled training set of MTAT and then fine-tuned on the labeled
 633 training set of GTZAN. The performance of Music-PAW and various
 634 baselines are shown in table 3. When the backbone is FCN-4, Music-
 635 PAW is superior to other self-supervised methods with identical
 636 architecture but can not get better results compared with the most
 637 advanced supervised methods. After we deepen the architecture
 638 with FCN-7 and fine-tune it on GTZAN’s training set, our method
 639 can obtain 71.7% accuracy on GTZAN’s test set, representing a
 640 substantially improvement over the state-of-the-art supervised and
 641 self-supervised pre-training method.

642 **Cover Song Identification & Acoustic Scene classification.** We
 643 are interested in whether music representations with part-whole hi-
 644 erarchy could benefit many other music downstream tasks besides
 645 music classification, even can be general to other audio scene. To-
 646 wards this end, we consider cover song identification and acoustic
 647 scene classification for evaluation. For brevity, we use R.FCN and
 648 S.FCN to denote the supervised FCN-4 [10] baselines with random
 649 initialization and with supervised training on the MTAT dataset,
 650 respectively. M-F refers to a state-of-the-art supervised baseline
 651 Multi-Format [61].

652 **Cover song identification** [31, 70, 72], aims to identify an alterna-
 653 tive version of a previous musical compositions. A cover song or
 654 cover version is a new performance or recording by a musician
 655 other than the original performer or composer of the song. To
 656 compare more clearly, we time the actual values of MAP and Preci-
 657 sion@10 with 100. We add FCN-4 on the top of CQTNet to get our
 658 backbone. Random init means the backbone’s FCN-4 is randomly
 659 initialized. We also compare the performance of Music-PAW with
 660 the state-of-the-art supervised baseline Multi-Format [61].

Table 4: Cover song identification evaluation results. Best results are in bold.

Method	SHS100K-SUB			Covers80		
	MAP ↑	Precision@10 ↑	MR1 ↓	MAP ↑	Precision@10 ↑	MR1 ↓
Ki-Net [67]	11.2	15.6	68.33	36.8	5.2	32.10
TPP-Net [71]	26.7	21.7	35.75	50.0	6.8	17.08
FCN-4 [10]	28.9	23.0	34.86	52.9	7.3	12.50
CQT-Net [72]	44.6	32.3	18.09	66.6	7.7	12.20
Random init	43.3	31.7	21.13	62.4	7.9	14.43
Supervised-MTAT	47.1	33.8	21.51	71.6	8.2	8.73
BYOL-A [45]	46.2	33.4	19.78	73.0	8.4	6.67
Multi-Format [61]	47.7	33.9	19.74	71.8	8.3	8.83
COLA [49]	47.2	34.0	20.15	74.6	8.5	7.86
Music-PAW	52.2	35.8	14.53	74.8	8.6	7.53

Table 5: Acoustic scene classification results. Following cross-validation setup of DCASE2016, we report folder-wise accuracy. The best results are in bold, the second best results are underlined.

Method	DNN [35]	R. FCN [10]	S. FCN [10]	M-F [1]	COLA [49]	BYOL-A	Music-PAW
folder1	80.0.	76.2	80.7	80.3	84.1	<u>83.4</u>	84.8
folder2	70.7	76.2	79.7	79.7	80.0	78.3	<u>79.7</u>
folder3	74.8	73.2	79.9	75.8	77.2	78.5	<u>78.5</u>
folder4	80.1	69.2	76.0	77.4	77.1	80.1	<u>79.8</u>
average	76.4	73.7	79.1	78.3	79.6	<u>80.1</u>	80.7

initialized, and Supervised-MTAT's is pre-trained on MTAT. Like Supervised-MTAT, ours and other self-supervised methods add their pre-trained encoder on the top of CQTNet. We fine-tune the combined network using SHS100K-SUB's training set and evaluate on SHS100K-SUB's test set and Covers80. All experimental results are shown in table 4. Our learned audio representations significantly improve the identification performance on SHS100K-SUB and obtain slight improvement on Covers80.

*Acoustic scene classification*⁴ aims to detect the scene where there is a test recording, and further identify the environment such as "park" and "home". Following [35], we adopt the DCASE2016 dataset, which is split into a development file and an evaluation file. The evaluation protocol in [35] concerns two classification tasks. Firstly, we follow its cross-validation setup, which uses four folders containing different training and testing files from the development file. The results are reported in table 5. All self-supervised methods have the same backbone FCN-4. We also compare with R. FCN-4, whose parameters are randomly initialized, and with S. FCN-4, whose parameters are pre-trained on MTAT in a supervised manner. Our method performs best on folder1. On the other three folders, we get the second best performance. We can obtain the best average accuracy of the four folders. Afterward, we use the audio

data from the whole development file to fine-tune our model and evaluate it on the audio data from DCASE2016's evaluation file. The evaluation results are reported in table 6. Music-PAW obtain the second best detection accuracy. It is not the best result compared with the SOTA supervised method RN2, but the gap is very tiny. Moreover, Music-PAW can perform much better than all previous self-supervised, which demonstrates the strong transferability of our audio representations.

4.3 Model Analysis

We take an in-depth analysis of the model architecture and the multi-hierarchy representation space through ablation study and representation visualization, respectively.

Ablation Study. We are interested in if critical components all contribute to the effectiveness of Music-PAW. In this regard, we construct a variant without the part-whole transformer block as Hc, a variant without hierarchical contrastive learning objective as Pw block, and a variant without both as Baseline. The results of Music-PAW and all variants across different metrics and two datasets are shown in Figure 3. In a nutshell, removing any component will lead to a performance drop. In particular, removing both components, which means a complete loss of part-whole music learning, leads to a significantly performance drop compared to removing one

⁴<https://dcase.community/>

Table 6: Acoustic scene classification performance on evaluation set of DCASE2016.

Method	DNN [35]	R. FCN [10]	S. FCN [10]	ResNet [36]	VGG [36]	Densenet [36]	BYOL-A [45]
Accuracy	81.0	81.0	83.6	83.2	83.0	83.7	82.6
Method	M-F [1]	COLA [49]	DN1 [36]	RN1 [36]	RN3 [36]	RN2 [36]	Music-PAW
Accuracy	82.6	83.8	86.1	86.0	86.5	87.1	<u>86.8</u>

of them. These results show the necessity of part-whole feature interaction and multi-hierarchy representation of music.

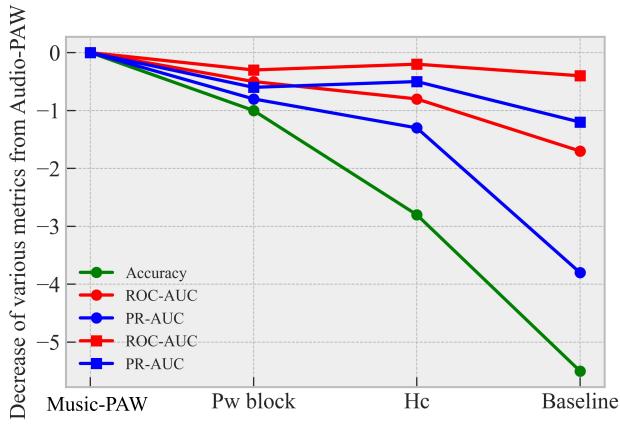
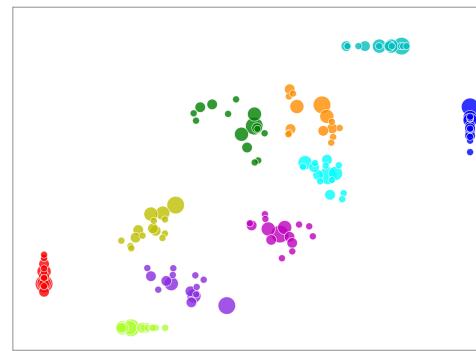


Figure 3: Ablation study of Music-PAW architecture. Square points and circle points correspond to performance on MTAT and GTZAN, respectively.

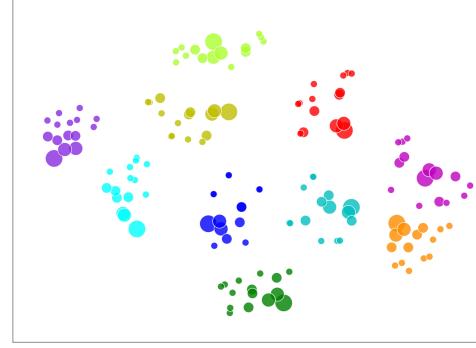
Visualization. We are interested in if the our framework forms a multi-hierarchy representation space of music. We extract music representations in different hierarchies using encoders pre-trained with Music-PAW and Baseline on the MTAT dataset. We then conduct t-SNE [41] transformation, and visualize the transformed representations in figure 4. The size of points represents the hierarchies of music clips, where a larger size indicates a longer music clip. According to the results, in many cases, the baseline fails to distinguish music of different hierarchies, which are also irregularly distributed. In contrast, the representations of different hierarchies from the same music in Music-PAW are more distinguishable and exhibit more noticeable intra-hierarchy clusters. These results demonstrate the merits of Music-PAW in learning music representations with part-whole hierarchies.

5 CONCLUSION

This work studies how to learn music representations with part-whole hierarchies from unlabeled data. We introduce Music-PAW, a self-supervised learning framework that explicitly infers the structural relationships of part-whole music components at different hierarchies through the devised part-whole transformer block. Moreover, the hierarchical contrastive learning objective in Music-PAW



(a) Baseline



(b) Music-PAW

Figure 4: Visualization of music representations in various hierarchies. Points in the same color with various sizes are hierarchical representations of the same music.

bridges the connections of part-whole audio representations in adjacent hierarchies and distinguishes them from part-whole audio representations from other audios, forming a multi-hierarchy audio representation space. We demonstrate the effectiveness of audio representations with part-whole hierarchies across various downstream audio tasks *w.r.t.* multiple evaluation settings, including in-domain evaluation, semi-supervised evaluation, and out-of-distribution evaluation. Although the datasets used in our work are medium-scale, we have planned to evaluate our methods on larger datasets.

REFERENCES

- [1] Haider Al-Tahan and Yalda Mohsenzadeh. 2021. CLAR: Contrastive Learning of Auditory Representations. *AISTATS 2021* (2021).

- 929 [2] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020.
930 wav2vec 2.0: A framework for self-supervised learning of speech representations.
931 *Advances in Neural Information Processing Systems* 2020-December (2020), 1–12.
932 [3] Pierre Baldi. 2012. Autoencoders, Unsupervised Learning, and Deep Architectures.
933 *ICML Unsupervised and Transfer Learning* (2012), 37–50. <https://doi.org/10.1561/2200000006>
934 [4] Lucas Beyer, Xiaohua Zhai, Avital Oliver, and Alexander Kolesnikov. 2019. S4L:
935 Self-supervised semi-supervised learning. *Proceedings of the IEEE International Conference on Computer Vision 2019-Octob* (2019), 1476–1485. <https://doi.org/10.1109/ICCV.2019.00156>
936 [5] Andrew Brock, Jeff Donahue, and Karen Simonyan. 2019. Large scale GaN
937 training for high fidelity natural image synthesis. *7th International Conference on Learning Representations, ICLR 2019* (2019), 1–35.
938 [6] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018.
939 Deep clustering for unsupervised learning of visual features. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11218 LNCS (2018), 139–156. https://doi.org/10.1007/978-3-030-01264-9_9
940 [7] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and
941 Armand Joulin. 2020. Unsupervised Learning of Visual Features by Contrasting
942 Cluster Assignments. *NeurIPS* (2020), 1–13. <http://arxiv.org/abs/2006.09882>
943 [8] Joao Carreira, Skanda Koppula, Daniel Zoran, Adria Recasens, Catalin Ionescu,
944 Olivier Henaff, Evan Shelhamer, Relja Arandjelovic, Matt Botvinick, Oriol
945 Vinyals, Karen Simonyan, Andrew Zisserman, and Andrew Jaegle. 2022. Hierar-
946 chical Perceiver. *CVPR 2022* (2022). <http://arxiv.org/abs/2202.10890>
947 [9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020.
948 A simple framework for contrastive learning of visual representations. *37th
949 International Conference on Machine Learning, ICML 2020* Part F16814, Figure 1
950 (2020), 1575–1585.
951 [10] Keunwoo Choi, György Fazekas, and Mark Sandler. 2016. Automatic tagging
952 using deep convolutional neural networks. *Proceedings of the 17th International
953 Society for Music Information Retrieval Conference,ISMIR 2016* (2016), 805–811.
954 [11] Keunwoo Choi, Gyorgy Fazekas, Mark Sandler, and Kyunghyun Cho. 2017. Con-
955 volutional recurrent neural networks for music classification. *ICASSP, IEEE
956 International Conference on Acoustics, Speech and Signal Processing - Proceedings*
957 (2017), 2392–2396. <https://doi.org/10.1109/ICASSP.2017.7952585>
958 [12] Ching-Yao Chuang, Joshua Robinson, Liu Yen-Chen Antonio Torralba, and
959 Stefanie Jegelka. 2020. (NIPS2020)Debiased Contrastive Learning. *NeurIPS* (2020),
960 1–20.
961 [13] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT:
962 Pre-training of deep bidirectional transformers for language understanding.
963 *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association
964 for Computational Linguistics: Human Language Technologies - Proceedings
965 of the Conference 1* (2019), 4171–4186.
966 [14] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. 2015. Unsupervised visual
967 representation learning by context prediction. *Proceedings of the IEEE Interna-
968 tional Conference on Computer Vision 2015 International Conference on Computer
969 Vision, ICCV 2015* (2015), 1422–1430. <https://doi.org/10.1109/ICCV.2015.167>
970 [15] Jeff Donahue and Karen Simonyan. 2019. Large scale adversarial representation
971 learning. *Advances in Neural Information Processing Systems* 32, NeurIPS (2019),
972 1–32.
973 [16] Xingjian Du, Zhesong Yu, Bilei Zhu, Xiaoou Chen, and Zejun Ma. 2021. Bytecover:
974 Cover Song Identification Via Multi-Loss Training. *ICASSP 2021* (2021), 551–555.
975 <https://doi.org/10.1109/icassp39728.2021.9414128>
976 [17] Nicola Garau, Niccolò Bisagno, Zeno Sambugaro, and Nicola Conci. 2022. Inter-
977 pretative part-whole hierarchies and conceptual-semantic relationships in neural
978 networks. *CVPR 2022* (2022). <http://arxiv.org/abs/2203.03282>
979 [18] Michal Genussov and Israel Cohen. 2010. Musical genre classification of audio
980 signals using geometric methods. *European Signal Processing Conference* 10, 5
981 (2010), 497–501.
982 [19] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. 2018. Unsupervised rep-
983 resentation learning by predicting image rotations. *6th International Conference on
984 Learning Representations, ICLR 2018 - Conference Track Proceedings* (2018),
985 1–16.
986 [20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley,
987 Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial
988 networks. *Commun. ACM* 63, 11 (2020), 139–144. <https://doi.org/10.1145/3422622>
989 [21] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H.
990 Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhao-
991 han Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu,
992 Rémi Munos, and Michal Valko. 2020. Bootstrap your own latent: A new ap-
993 proach to self-supervised Learning. *NeurIPS 2020* 200 (2020). [http://arxiv.org/abs/2006.07733](http://arxiv.org/abs/abs/2006.07733)
994 [22] Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction
995 by learning an invariant mapping. *Proceedings of the IEEE Computer Society
996 Conference on Computer Vision and Pattern Recognition 2* (2006), 1735–1742.
997 <https://doi.org/10.1109/CVPR.2006.100>
- [23] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. 2021. Transformer in Transformer. *NeurIPS* (2021), 1–12. <http://arxiv.org/abs/2103.00112>
[24] Jeff Hawkins. 2021. A thousand brains: A new theory of intelligence. (2021).
[25] Kaiming He. 2015. Delving Deep into Rectifiers : Surpassing Human-Level
Performance on ImageNet Classification. *CVPR 2015* (2015).
[26] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Mo-
mentum Contrast for Unsupervised Visual Representation Learning. In *Proceedings of
the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.
9726–9735. <https://doi.org/10.1109/CVPR42600.2020.00975>
[27] Olivier J. Hénaff, Ali Razavi, Carl Doersch, S. M. Ali Eslami, and Aaron Van Den
Oord. 2019. Data-efficient image recognition with contrastive predictive coding.
CVPR 2020 2018 (2019).
[28] Geoffrey Hinton. 2021. How to represent part-whole hierarchies in a neural
network. (2021), 1–44. <http://arxiv.org/abs/2102.12627>
[29] Qianjiang Hu, Xiao Wang, Wei Hu, and Guo Jun Qi. 2021. AdCO: Ad-
versarial Contrast for Efficient Learning of Unsupervised Representations from
Self-Trained Negative Adversaries. *Proceedings of the IEEE Computer Society
Conference on Computer Vision and Pattern Recognition* (2021), 1074–1083.
<https://doi.org/10.1109/CVPR46437.2021.00113>
[30] Aren Jansen, Manoj Plakal, Rathesh Pandya, Daniel P.W. Ellis, Shawn Hershey,
Jiayang Liu, R. Channing Moore, and Rif A. Saurous. 2018. Unsupervised Learning
of Semantic Audio Representations. *ICASSP, IEEE International Conference on
Acoustics, Speech and Signal Processing - Proceedings 2018-April* (2018), 126–130.
<https://doi.org/10.1109/ICASSP.2018.8461684>
[31] Deshun Yang Jiang, Chaoya and Xiaou Chen. 2020. Similarity Learning for
Cover Song Identification Using Cross-Similarity Matrices of Multi-Level Deep
Sequences. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal
Processing - Proceedings 2020-May* (2020), 26–30.
[32] Yannis Kalantidis, Mert Bulent Sarıyıldız, Noe Pion, Philippe Weinzaepfel, and
Diana Larlus. 2020. Hard negative mixing for contrastive learning. *Advances in
Neural Information Processing Systems* 2020-December, NeurIPS (2020), 1–12.
[33] Taejun Kim, Jongpil Lee, and Juhan Nam. 2018. Sample-Level CNN Architectures
for Music Auto-Tagging Using Raw Waveforms. *ICASSP, IEEE International
Conference on Acoustics, Speech and Signal Processing - Proceedings 2018-April*
(2018), 366–370. <https://doi.org/10.1109/ICASSP.2018.8462046>
[34] Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic
optimization. *3rd International Conference on Learning Representations, ICLR 2015
- Conference Track Proceedings* (2015), 1–15.
[35] Qiuqiang Kong, Iwnoa Sobieraj, Wenwu Wang, and Mark Plumley. 2016. Deep
neural network baseline for DCASE challenge 2016. *University of Surrey Septem-
ber* (2016), 4–8.
[36] Khaled Koutini, Hamid Eghbal-Zadeh, Matthias Dorfer, and Gerhard Widmer.
2019. The receptive field as a regularizer in deep convolutional neural networks
for acoustic scene classification. *European Signal Processing Conference 2019-
Septe* (2019). <https://doi.org/10.23919/EUSIPCO.2019.8902732>
[37] Edith Law, Kris West, Michael Mandel, Mert Bay, and J. Stephen Downie. 2009.
Evaluation of algorithms using games: The case of music tagging. *Proceedings of
the 10th International Society for Music Information Retrieval Conference, ISMIR
2009* Ismir (2009), 387–392.
[38] Jongpil Lee and Juhan Nam. 2017. Multi-Level and Multi-Scale Feature Aggrega-
tion Using Pre-trained Convolutional Neural Networks for Music Auto-tagging.
PMLR 2017 (2017), 1–5.
[39] Jongpil Lee, Jiyoun Park, Keunhyoung Luke Kim, and Juhan Nam. 2019. Sample-
level deep convolutional neural networks for music auto-tagging using raw
waveforms. *Proceedings of the 14th Sound and Music Computing Conference 2017,
SMC 2017* (2019), 220–226.
[40] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin,
and Baining Guo. 2022. Swin Transformer: Hierarchical Vision Transformer
using Shifted Windows. (2022), 9992–10002. <https://doi.org/10.1109/iccv48922.2021.00986>
[41] L. v. d. Maaten and G. Hinton. 2008. Visualizing Data using t-SNE. *Journal
of machine learning research* (2008).
[42] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. 2016. TUT Database
for Acoustic Scene Classification and Sound Event Detection. In *24th European
Signal Processing Conference 2016 (EUSIPCO 2016)* Budapest, Hungary.
[43] Ishan Misra and Laurens van der Maaten. 2020. Self-supervised learning of
pretext-invariant representations. *Proceedings of the IEEE Computer Society
Conference on Computer Vision and Pattern Recognition* (2020), 6706–6716.
<https://doi.org/10.1109/CVPR42600.2020.00674>
[44] Vinod Nair and Geoffrey E. Hinton. 2010. Rectified linear units improve restricted
boltzmann machines. In *International Conference on Machine Learning, 2010*
(2010).
[45] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio
Kashino. 2021. BYOL for Audio: Self-Supervised Learning for General-Purpose
Audio Representation. *IJCNN 2020* (2021). <http://arxiv.org/abs/2103.06695>
[46] Mehdi Noroozi and Paolo Favaro. 2016. Unsupervised learning of visual represen-
tations by solving jigsaw puzzles. *Lecture Notes in Computer Science (including*

- 1045 *subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9910 LNCS (2016), 69–84. https://doi.org/10.1007/978-3-319-46466-4_5
- 1046 [47] Jordi Pons, Oriol Nieto, Matthew Prockup, Erik Schmidt, Andreas Ehmann, and Xavier Serra. 2018. End-to-end learning for music audio tagging at scale. *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018* (2018), 637–644.
- 1047 [48] Jiezong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. 2020. GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2020), 1150–1160. <https://doi.org/10.1145/3394486.3403168>
- 1048 [49] Aaqib Saeed, David Grangier, and Neil Zeghidour. 2021. Contrastive Learning 1053 of General-Purpose Audio Representations. *ICASSP 2021* (2021), 3875–3879. <https://doi.org/10.1109/icassp39728.2021.9413528>
- 1049 [50] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. 1054 WAV2vec: Unsupervised pre-training for speech recognition. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH 2019-Septe* (2019), 3465–3469. <https://doi.org/10.21437/Interspeech.2019-1873>
- 1050 [51] Tatuya O. Sharpee, Craig A. Atencio, and Christoph E. Schreiner. 2011. Hierarchical 1055 representations in the auditory cortex. *Current Opinion in Neurobiology* 21, 5 (2011), 761–767. <https://doi.org/10.1016/j.conb.2011.05.027>
- 1051 [52] Kihyuk Sohn. 2016. Improved deep metric learning with multi-class N-pair loss objective. *Advances in Neural Information Processing Systems Nips* (2016), 1056 1857–1865.
- 1052 [53] Janne Spijkervet and John Ashley Burgoyne. 2021. Contrastive Learning of 1057 Musical Representations. (2021). <http://arxiv.org/abs/2103.09410>
- 1058 [54] Bob L. Sturm. 2013. The GTZAN dataset: Its contents, its faults, their effects on 1062 evaluation, and its future use. 11 (2013), 1–29. <https://doi.org/10.1080/09298215.2014.894533>
- 1059 [55] Shuyang Sun, Xiaoyu Yue, Song Bai, and Philip Torr. [n.d.]. Visual Parser : 1063 Representing Part-whole Hierarchies with Transformers. ([n. d.]).
- 1060 [56] Yonglong Tian, Dilip Krishnan, and Phillip Isola. 2020. Contrastive Multiview 1066 Coding. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 12356 LNCS (2020), 1069 776–794. https://doi.org/10.1007/978-3-030-58621-8_45
- 1070 [57] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre 1073 Sablayrolles, and Hervé Jégou. 2020. Training data-efficient image transformers 1074 distillation through attention. (2020). <http://arxiv.org/abs/2012.12877>
- 1071 [58] Zhuowen Tu, Alan Yuille, and Xiangrong Chen. 2006. Image Parsing: Segmentation, 1075 Detection, and Recognition. *Towards Category-Level Object Recognition* 63, 2 (2006), 545–576.
- 1072 [59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, 1078 Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems 2017-Decem*, Nips (2017), 5999–6009.
- 1073 [60] Luyu Wang and Aaron van den Oord. 2021. Multi-Format Contrastive Learning 1079 of Audio Representations. *NeurIPS workshop 2020* (2021). <http://arxiv.org/abs/2103.06508>
- 1080 [61] Luyu Wang and Aaron van den Oord. 2021. Multi-Format Contrastive Learning 1081 of Audio Representations. arXiv:2103.06508 (March 2021).
- 1082 [62] Wenhui Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong 1083 Lu, Ping Luo, and Ling Shao. 2022. Pyramid Vision Transformer: A Versatile 1084 Backbone for Dense Prediction without Convolutions. *ICCV 2021* (2022), 548–558. <https://doi.org/10.1109/iccv48922.2021.00061>
- 1085 [63] Minz Won, Sanghyuk Chun, Oriol Nieto, and Xavier Serrc. 2020. Data-Driven 1086 Harmonic Filters for Audio Representation Learning. *ICASSP, IEEE International 1087 Conference on Acoustics, Speech and Signal Processing - Proceedings 2020-May* (2020), 536–540. <https://doi.org/10.1109/ICASSP40776.2020.9053669>
- 1088 [64] Minz Won, Sanghyuk Chun, and Xavier Serra. 2019. Toward Interpretable Music 1089 Tagging with Self-Attention. (2019). <http://arxiv.org/abs/1906.04972>
- 1090 [65] Minz Won, Andres Ferraro, Dmitry Bogdanov, and Xavier Serra. 2020. Evaluation 1091 of CNN-based automatic music tagging models. *Proceedings of the Sound and 1092 Music Computing Conferences 2020-June* (2020), 331–337.
- 1093 [66] Zhihong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. 2018. Unsupervised 1094 Feature Learning via Non-parametric Instance Discrimination. *Proceedings of the IEEE 1095 Computer Society Conference on Computer Vision and Pattern Recognition* (2018), 3733–3742. <https://doi.org/10.1109/CVPR.2018.00393>
- 1096 [67] Xiaoshuo Xu, Xiaoou Chen, and Deshun Yang. 2018. Key-Invariant Convolutional 1097 Neural Network Toward Efficient Cover Song Identification. *Proceedings - IEEE 1098 International Conference on Multimedia and Expo 2018-July* (2018), 1–6. <https://doi.org/10.1109/ICME.2018.8486531>
- 1099 [68] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, 1100 and Jianfeng Gao. 2021. Focal Self-attention for Local-Global Interactions in 1101 Vision Transformers. *NeurIPS 2021* (2021), 1–21. <http://arxiv.org/abs/2107.00641>
- 1102 [69] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph Contrastive Learning with Augmentations. *NeurIPS* (2020), 1–12.

1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160

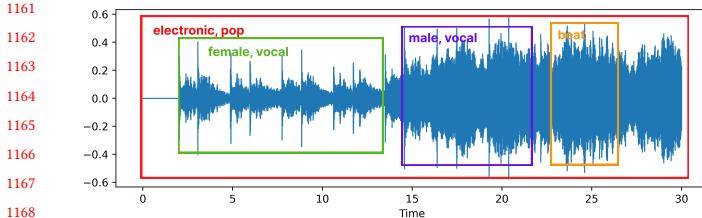


Figure 5: The waveform of a music audio selected from MTAT dataset. We use several boxes with different colors to tag its labels of the different parts.

A APPENDIX

We selected a specific music example named "mrdc-timecode-01-lust-175-204.mp3" from MTAT. This music audio file has several labels to portray its characteristics, including "electronic", "beat", "vocal", "male vocal", "female vocal", and "pop". Its waveform is shown in Figure 5. "Electronic" and "pop" labels are whole-level traits, which are tagged by the red box. "Female", "vocal", "male", and "beat" are part-level traits reflected in different clips, which are

tagged by the green, purple, and orange boxes, respectively. Therefore, we devised Music-PAW to represent part-whole hierarchies of music, which can help the model perceive the hierarchies of music characteristics more accurately. Our model can better understand music, thus improving the performance of downstream tasks.

In Section 4.3, Figure 4 shows that the Baseline model fails to distinguish music of different hierarchies, which are also irregularly distributed. In contrast, the representations of different hierarchies from the same music in Music-PAW are more distinguishable and exhibit noticeable intra-hierarchy clusters. Specifically, in the multi-label music classification task, the part-level representations of the Baseline model mix up with the whole-level representations. This shortcoming means that the Baseline model cannot clearly perceive the part-level traits and the whole-level traits of music when tagging music. In other words, part-level and whole-level representations can only reflect one of the part-level or whole-level traits. However, the part-level representations of Music-PAW are more distinguishable and are not far away from the whole-level representations simultaneously. Therefore, when Music-PAW utilizes the part-level representations to tag part-level traits for music, it can also perceive the whole-trait of music. Thus, Music-PAW can recognize the all traits of music more accurately.

1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218

1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276