A CROSS-VERSION APPROACH TO AUDIO REPRESENTATION LEARNING FOR ORCHESTRAL MUSIC

Michael Krause[1] **Christof Weiß**[2] **Meinard Müller**[1]
[1] International Audio Laboratories Erlangen, Germany
[2] University of Würzburg, Germany
{michael.krause,meinard.mueller}@audiolabs-erlangen.de, christof.weiss@uni-wuerzburg.de

ABSTRACT

Deep learning systems have become popular for tackling a variety of music information retrieval tasks. However, these systems often require large amounts of labeled data for supervised training, which can be very costly to obtain. To alleviate this problem, recent papers on learning music audio representations employ alternative training strategies that utilize unannotated data. In this paper, we introduce a novel cross-version approach to audio representation learning that can be used with music datasets containing several versions (performances) of a musical work. Our method exploits the correspondences that exist between two versions of the same musical section. We evaluate our proposed cross-version approach qualitatively and quantitatively on complex orchestral music recordings and show that it can better capture aspects of instrumentation compared to techniques that do not use cross-version information.

1. INTRODUCTION

Deep learning (DL) has become a common tool for approaching diverse tasks in music information retrieval (MIR). These approaches usually follow a supervised learning scheme, where a neural network is trained on the annotations of some dataset. For many MIR tasks, however, such annotations are costly to obtain. Recent work has investigated alternatives that require little or no annotations and enable training on large, unannotated datasets.

For certain music genres, there are datasets that contain several versions (i. e., recorded performances) of a musical work. For example, the same classical symphony or concerto can be performed by different orchestras, and several commercial recordings are often available. On such datasets, automatic music synchronization techniques can be used to find alignments between different versions of a work, requiring minimal annotation effort [1, 2].

In this paper, we introduce a conceptually novel approach to audio representation learning that exploits cross-version datasets, thus requiring only alignments between versions and no further human annotations. Our approach aims at learning embeddings of audio excerpts such that musically corresponding excerpts in different versions are mapped to close points in the embedding space (Figure 1).

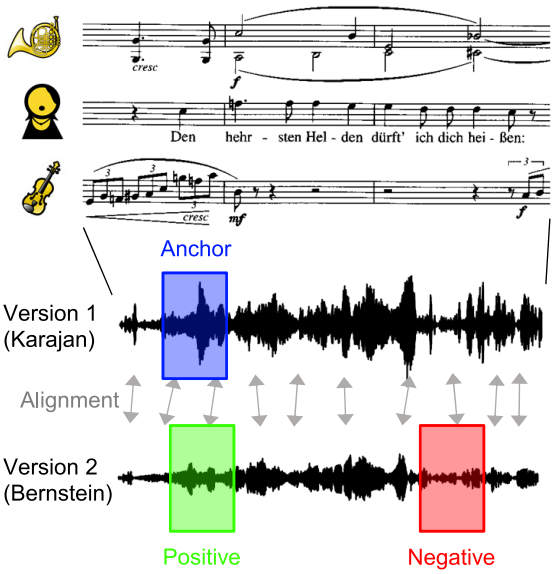

Figure 1: Visualization of our cross-version approach to representation learning for orchestral music. An anchor (blue) excerpt is selected from a music recording. The positive (green) and negative (red) excerpts are chosen from a different version of the same musical piece. For this, an alignment between versions is needed (gray arrows).

There are several musical aspects that stay roughly constant across most versions, e. g., pitches, harmonies or rhythm. For orchestral music, aspects of instrumentation (i. e., active instruments or instrument families) are another such property. Instrumentation represents a challenging MIR scenario given the complexity of instrument taxonomies and the difficulty of annotating instrument activity in orchestral music. In our experiments on a dataset of complex orchestral music, we show qualitatively and quantitatively that—by utilizing the correspondences between different versions of a musical section—our proposed representation learning technique is better at capturing aspects of instrumentation and instrument texture compared to approaches that do not exploit cross-version information.

The remainder of the paper is structured as follows: Section 2 covers related work on music audio representation learning, cross-version analysis, and instrumentation in orchestral music. In Section 3, we introduce our proposed approach. In Section 4, we describe our experimental setup, including datasets, our model architecture, and baselines. Section 5 contains qualitative and quantitative results and Section 6 concludes the paper with a discussion of possible future work.

2. RELATED WORK

Several recent contributions have explored so-called self-supervised strategies for learning representations from unannotated music recordings. Often, in these studies, excerpts from a music recording that are in close proximity are considered as positive pairs (i. e., should be mapped to similar representations) whereas excerpts that are further apart (or from other recordings) are negative pairs (i. e., should be mapped to dissimilar representations). This idea is also illustrated in Figure 2. McCallum [3] originally considered this with the aim of learning features for music structure analysis. Wang et al. [4] had a similar use case but used a supervised learning approach. Several authors employed such a strategy for learning more general purpose representations [5–10], often applying additional augmentations. Apart from using temporal proximity, other papers on music representation learning exploit audio-visual or audio-text correspondences [11, 12], use classical features as training targets [13], exploit metadata [14], or investigate music generation models [15].

The approach proposed in this paper is conceptually different since we utilize cross-version datasets, rather than relying on temporal proximity alone. Such datasets contain several recorded versions of a musical work, which may vary in aspects related to musical interpretation, recording conditions, and timbral characteristics of the instruments used. These datasets have been exploited for expressive performance rendering [16] or improved harmonic analysis [17]. Cross-version datasets also allow for investigating model biases and overfitting effects in MIR models through different dataset splits [18]. To our knowledge, the only other work utilizing cross-version information for embedding learning is by Zalkow et al. [19], whose aim was to compress chromagram excerpts for efficient music retrieval. In contrast, we propose to learn representations based on spectrogram-like input features and investigate them for capturing instrument texture.

In the wider machine learning literature, representations are often learned by masking a part of an input and predicting the masked content [20, 21]. Other strategies utilize multi-modal datasets, e. g., containing text–image [22] or audio–text pairs [23].

Orchestral music has been explored in the context of source separation [24] or melody extraction [25]. The authors in [26] considered instrument family recognition for classical, monotimbral recordings using a supervised learning approach. Other recent papers on instrument activity detection in music recordings [27–29] have also considered DL-based, supervised learning approaches, but not within orchestral scenarios.

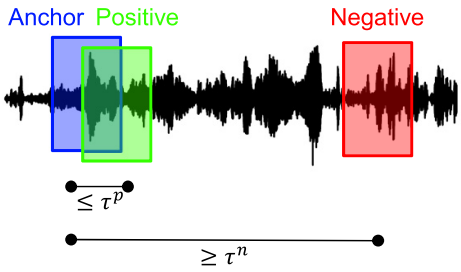

Figure 2: When forming triplets of audio excerpts, the anchor and positive/negative excerpts are chosen according to a maximum/minimum distance $\tau^{\mathrm{p}}/\tau^{\mathrm{n}}$.

3. CROSS-VERSION APPROACH TO AUDIO REPRESENTATION LEARNING

In this section, we formalize our proposed cross-version approach to representation learning. The key idea is to utilize correspondences between different versions (i. e., recorded performances played by different orchestras) of the same musical work. We aim to learn embeddings of audio excerpts such that the same musical section in different versions is represented by neighboring points in the embedding space and audio excerpts for unrelated musical sections are mapped to distant points in the embedding space. To this end, inspired by [19], we sample triplets of audio excerpts as in Figure 1, and apply a triplet loss for learning. Musical characteristics that stay roughly constant across different versions of an orchestral work include pitches and harmonies, as well as instrumentation. In later sections, we will analyze to what extent our approach captures pitches or aspects of instrumentation.

Single-Version Approach (SV)**.** We begin by formalizing a common approach to music representation learning that only utilizes temporal proximity inside a single version, see also Section 2 and Figure 2. Let $\mathcal{W}$ be a set of musical works and let V_w be the set of available versions for a work $w \in \mathcal{W}$. We first randomly select a work $w \in \mathcal{W}$ and some version of this work $v \in V_w$. Let T denote the length of v in seconds. We choose an anchor excerpt by uniformly sampling an anchor position $a \in [0, T]$ and extracting the excerpt $\mathbf{x}^{\mathrm{a}}$ of v that is centered around a. To obtain the positive and negative excerpts, we choose a position $p \in [0, T]$ for the positive excerpt $\mathbf{x}^{\mathrm{p}}$ of v such that $|a - p| \leq \tau^{\mathrm{p}}$. Thus, the positive excerpt is in temporal proximity of the anchor excerpt—up to a threshold of τ^{p} seconds—and is likely to correspond to a musically similar section. In the same way, we choose a position $n \in [0, T]$ for the negative excerpt $\mathbf{x}^{\mathrm{n}}$ of v such that $|a - n| \geq \tau^{\mathrm{n}}$. The negative excerpt is therefore a certain minimum distance of τ^{n} seconds away from the anchor position, likely corresponding to a musically dissimilar section. [1]

[1] Due to repetitions and other structural similarities, there may in fact be some musically related sections that are far apart temporally. In the majority of cases, however, the assumption underlying positive and negative sampling will hold [3].

Embedding Learning. We obtain embeddings by passing these excerpts through a neural network (described in Section 4.2), i. e.:

$$\mathbf{Y} = (\mathbf{y}^{\mathrm{a}}, \mathbf{y}^{\mathrm{p}}, \mathbf{y}^{\mathrm{n}}) = (f(\mathbf{x}^{\mathrm{a}}), f(\mathbf{x}^{\mathrm{p}}), f(\mathbf{x}^{\mathrm{n}})), \quad (1)$$

where f is a neural network that embeds an audio excerpt $\mathbf{x}$ into an embedding vector $\mathbf{y}$. Using this triplet, we can apply a standard triplet loss [30] such as:

$$\mathcal{L}(\mathbf{Y}) = \max\left(0, \|\mathbf{y}^{\mathrm{a}} - \mathbf{y}^{\mathrm{p}}\|_2^2 - \|\mathbf{y}^{\mathrm{a}} - \mathbf{y}^{\mathrm{n}}\|_2^2 + \alpha\right), \quad (2)$$

where $\alpha \in \mathbb{R}_{\geq 0}$ describes the desired minimum margin between the distance of embeddings for anchor and positive versus the distance of embeddings for anchor and negative.

Cross-Version Approach (`CV`)**.** For our proposed cross-version approach, we sample triplets in a different fashion. Since we utilize multiple versions per work, we now require $|V_w| \geq 2$. To form a triplet of excerpts, we randomly select some version $v_1 \in V_w$ of a work $w \in \mathcal{W}$. We then sample an anchor position $a_1 \in [0, T_1]$, where T_1 is the length of v_1 in seconds, and extract the corresponding excerpt $\mathbf{x}^{\mathrm{a}}$ of v_1. To obtain the positive and negative excerpts, we randomly select another version $v_2 \in V_w \setminus \{v_1\}$ of w. As before, let T_2 denote the length of v_2 in seconds. We can find the position $a_2 \in [0, T_2]$ in v_2 corresponding to the same musical position as the anchor a_1 in v_1 using music alignment techniques. With this, we choose a position $p \in [0, T_2]$ for the positive excerpt $\mathbf{x}^{\mathrm{p}}$ of v_2 such that $|a_2 - p| \leq \tau^{\mathrm{p}}$. Thus, the positive excerpt corresponds to the same musical section as the anchor, up to some tolerance of τ^{p} seconds (in addition to alignment inaccuracies). Similarly, we sample $n \in [0, T_2]$ (with $|a_2 - n| \geq \tau^{\mathrm{n}}$) and extract $\mathbf{x}^{\mathrm{n}}$. Note that only $\mathbf{x}^{\mathrm{a}}$ is an excerpt of the first version v_1, whereas both $\mathbf{x}^{\mathrm{p}}$ and $\mathbf{x}^{\mathrm{n}}$ are excerpts of the second version v_2. As before, we construct a triplet $\mathbf{Y}$ using these excerpts and apply a standard triplet loss.

4. EXPERIMENTAL SETUP

4.1 Dataset and Splits

To show the potential of our representation learning technique, we construct a cross-version dataset of commercial symphonic and opera music recordings, illustrated in Table 1. Our dataset contains an act from an opera (the first act from Richard Wagner's "Die Walküre") as well as orchestral pieces by Beethoven, Dvorak and Tschaikowsky. Counting each movement as an individual work, the dataset contains eleven different works in total. We choose the first movement of the Beethoven Symphony, the fourth movement of the Dvorak Symphony and the third movement of the Tschaikowsky Concerto for testing. Since we do not have multiple opera acts that could be split into train and test, we choose an excerpt of the Wagner opera act (measures 697 to 955, corresponding to around twelve minutes), omit this excerpt during training, and use it for testing. We further ensure that the train and test set contain different versions. By splitting our dataset in this fashion, we aim to avoid overfitting to specific musical compositions or recording conditions (the latter is also referred to as "album effect" [31]).

Composer	Work	Versions	
		Num.	Avg. Duration
Wagner	Die Walküre, Act 1	8	1 h
Beethoven	Symph. 3, Mvmts. 1–4	6	45 min
Dvorak	Symph. 9, Mvmts. 1, 2, 4	6	40 min
Tschaikowsky	Violin Concerto, Mvmts. 1–3	6	35 min
Total duration			20 h

Table 1: Our cross-version dataset containing several commercial recordings of different orchestral and opera compositions.

For the cross-version approach `CV`, we obtain an alignment between versions of the same work using state-of-the-art music synchronization techniques involving chroma onset features and multi-scale alignment [2]. For some experiments, we also require pitch-class and instrument activity annotations for our dataset. To this end, we manually encoded a score representation of "Die Walküre" and obtained further scores from the Mutopia project. [2] Again, we use music synchronization techniques to align score to audio and create the annotations.

4.2 Model

We implement all representation learning approaches using a convolutional neural network that takes a harmonic CQT representation (HCQT, [32]) of an audio excerpt as input and outputs a corresponding embedding vector. The HCQT input consists of 201 frames (at a frame rate of 43 Hz, i. e., roughly 4.7 seconds), three bins per semitone from C1 to B7 (leading to 252 bins), and five harmonics (with frequency multiples of $[0.5, 1, 2, 3, 4]$).

The model architecture is adapted from [33] and receives an HCQT input patch, processes it through several convolution and max-pooling layers, and outputs a single ℓ_2-normalized vector (length 128) per input. We take this output as the embedding vector for the center frame of the input patch. In total, the architecture has roughly 1.5 million learnable parameters. We train our network for 200 epochs (with 16 000 triples randomly sampled per epoch) using the Adam optimizer with a learning rate of 0.002. In the interest of reproducibility, we release code and trained models for our approach. [3]

In line with previous studies on audio representation learning [5–7], we apply a number of augmentations to excerpts during training, including time scaling, pitch shifting, random masking, adding noise and applying random equalization. For all experiments, we set $\tau^{\mathrm{p}} = 0.2\,\mathrm{s}$. With this, the maximal distance between anchor and positive excerpt is in the same order of magnitude as the typical alignment inaccuracy between versions. We further set

[2] `https://www.mutopiaproject.org/`

[3] `https://www.audiolabs-erlangen.de/resources/MIR/2023-ISMIR-CrossVersionLearning`

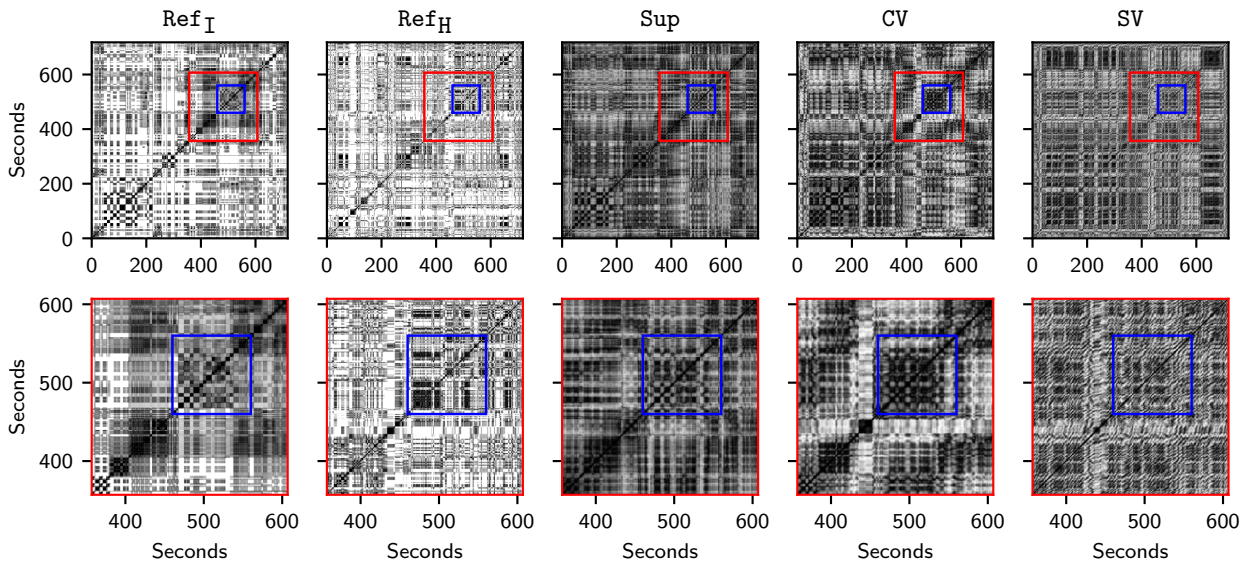

Figure 3: Self-similarity matrices constructed from instrument annotations ($\texttt{Ref}_{\texttt{I}}$) and pitch-class annotations ($\texttt{Ref}_{\texttt{H}}$), or obtained with a supervised learning system (Sup), the proposed cross-version approach (CV), and a baseline that does not incorporate cross-version information (SV). The lower row shows the sections highlighted in red from above.

$\tau^{\text{n}} = 10.0\,\text{s}$ and $\alpha = 1.0$. We found that results are stable for a broad range of settings of these parameters.

4.3 Baselines

To investigate the musical properties captured by the representation learning approaches CV and SV, we compare them to several optimistic baselines: First, we extract traditional music audio features. We use mel-frequency cepstral coefficients (MFCC), which are known to capture aspects of instrumentation [34], and Chroma features, which contain the dominant pitch-classes in the recording. Here, our goal is not to outperform MFCC or Chroma, but to compare them to our learned representations. If our learning approaches capture instrumentation, we expect them to behave similar to MFCCs. Likewise, in case they contain pitch-class information, we expect them to perform like Chroma features.

Second, we consider a supervised learning approach Sup where we train a model on instrument activity annotations and use its hidden representations as features. For this, we utilize the same model architecture as for CV and SV and only add a final dense layer with a number of outputs equal to the number of instruments to detect. Rather than using the triplet loss from Section 3, we train this approach by applying a sigmoid activation and binary cross-entropy loss. Note that in contrast to CV and SV, the Sup approach requires instrument activity annotations for the recordings in the training set.

5. RESULTS

5.1 Feature Analysis using Self-Similarity

In order to visualize and compare the representations learned by different techniques, we employ self-similarity matrices. Such matrices are commonly used for music structure analysis and allow for visualizing structures based on repetition and homogeneity in feature sequences [1]. Here, we use them to analyze our learned representations without the need for additional fine-tuning. This also allows us to directly compare approaches trained with a fixed instrument vocabulary (Sup) to others that are not informed about instruments. We provide an alternative evaluation in Section 5.4.

Given a sequence $X = (x_1, \ldots, x_N)$ of (learned) representations of N audio frames, we construct the corresponding self-similarity matrix $S \in \mathbb{R}^{N \times N}$ as follows. We first normalize all representations with respect to the ℓ_2-norm, yielding $\tilde{X} = (\tilde{x}_1, \ldots, \tilde{x}_N)$. We then compute $S(n, m) := \langle \tilde{x}_n, \tilde{x}_m \rangle$ for $n, m \in [1 : N]$. Thus, S contains the cosine similarities between elements of X, and all its entries lie in the interval $[-1, 1]$. By definition, all entries on the diagonal of S are equal to 1. In addition, repeated subsequences appear as path-like structures and homogeneous segments appear as block-like structures, see also [1].

We compare the self-similarity matrices obtained from learned representations to matrices created using reference annotations. First, we represent an instrument activity annotation as a sequence of multi-hot binary vectors (indicating the presence of instruments in different frames). By normalizing and computing the dot product as before, we obtain a matrix corresponding to instrument texture, where blocks indicate segments with similar instrumentation. We will refer to this matrix using the shorthand $\texttt{Ref}_{\texttt{I}}$. For example, the start of the middle measure in Figure 1 would be encoded as a vector $(1, 1, 1)^\top$, i. e., all instruments are active, and the end of that measure would be encoded as $(1, 1, 0)^\top$, i. e., only horn and soprano are active. After normalization, the dot product of these vectors is 0.82,

indicating similar instrumentation. Analogously, we construct another matrix $\texttt{Ref}_{\texttt{H}}$ from a sequence of pitch-class annotations. This matrix captures regions with similar harmonies and pitches.

5.2 Qualitative Results

Figure 3 shows several self-similarity matrices obtained through reference annotations or by different representation learning approaches. The excerpt shown in the upper row is the test excerpt from "Die Walküre" (similar results are obtained on other inputs). The lower row shows magnified sections from above. Darker color indicates higher similarity.

In the $\texttt{Ref}_{\texttt{I}}$ matrix, arising from instrument annotations as explained in Section 5.1, one can observe many block and checkerboard-like structures. For example, from seconds 460 to 560, different combinations of woodwind instruments are playing together, creating block and checkerboard-like patterns (highlighted in blue). White areas indicate $S(n,m) = 0$, i. e., no common instruments are playing. The matrix $\texttt{Ref}_{\texttt{H}}$, on the other hand, indicates harmonic similarities which are mostly distinct from the instrument similarities in $\texttt{Ref}_{\texttt{I}}$.

For the Sup system, many of the patterns in $\texttt{Ref}_{\texttt{I}}$ are replicated, albeit with less detail. This is expected, since this system has been trained on the same kind of annotations that have been used to create $\texttt{Ref}_{\texttt{I}}$. Interestingly, many of the patterns present in the $\texttt{Ref}_{\texttt{I}}$ and Sup matrices also appear for the proposed approach CV, which has not been trained using instrument annotations. In particular, the checkerboard pattern starting at second 460 is captured by CV, as well as many block structures.

There are fewer similarities between CV and $\texttt{Ref}_{\texttt{H}}$, indicating that the CV representations are more likely to capture instrumentation rather than pitch-class content. This behavior is encouraged by our augmentation strategy, where we randomly pitch-shift the anchor, positive and negative excerpts.

The matrix obtained through the SV approach is blurry and, unlike the results for CV, fails to capture many of the checkerboard-like patterns present in $\texttt{Ref}_{\texttt{I}}$. The example suggests that exploiting cross-version information during training is important for capturing aspects of instrumentation in learned representations.

5.3 Quantitative Results

In order to quantify the correlation between our learned representations and instrument texture, we now apply a procedure for detecting the boundaries of block-like structures in self-similarity matrices. We then compare block boundaries estimated on $\texttt{Ref}_{\texttt{I}}$ with boundaries from all other matrices. Such procedures are often used in the context of music structure analysis [1, 35].

To detect block boundaries, we first correlate a self-similarity matrix with a checkerboard kernel along the main diagonal, as proposed in [35]. From this, we obtain a novelty curve. We then apply a peak picking procedure using local thresholding on this novelty curve, yielding sparse positions of detected block structures. We do this for all approaches and reference matrices. We finally compare—with a tolerance of up to three seconds—the detected boundaries for all approaches to those of $\texttt{Ref}_{\texttt{I}}$, yielding a boundary F-measure. By adjusting the size of the checkerboard kernel in this procedure, we can identify changes of instrument texture on short or larger time scales. For more details on the boundary detection, peak picking, and evaluation procedure, we refer to [1].

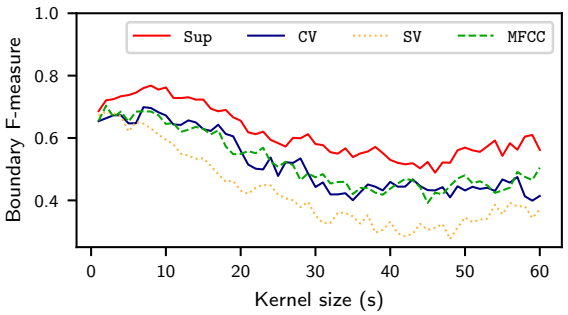

Figure 4: Results for different representation learning approaches when comparing estimated structure boundaries to boundaries from $\texttt{Ref}_{\texttt{I}}$.

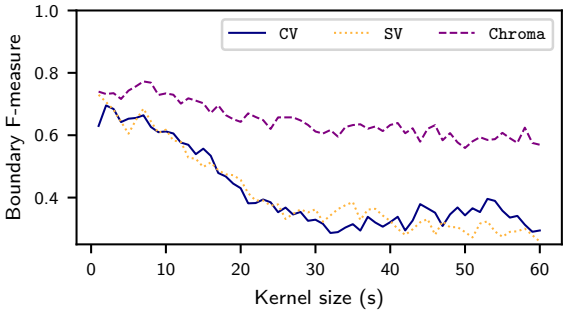

Figure 5: Results for comparing with $\texttt{Ref}_{\texttt{H}}$.

Figure 4 shows the results of our quantitative evaluation for different sizes of the checkerboard kernel. The F-measures given are averaged over all recordings in the test dataset. We observe that the supervised approach is best at capturing instrument texture (as encoded by $\texttt{Ref}_{\texttt{I}}$) compared to all others, with the highest F-measure of 0.77 for a kernel of eight seconds. CV and MFCC perform roughly on par. This is surprising, since CV is trained without any instrument annotations, while MFCC is known to capture instrumentation. Results for SV deteriorate with larger kernel sizes, dropping to as low as 0.28 F-measure for a kernel of 48 seconds. The proposed approach CV is better at capturing instrument texture than the alternative SV that does not utilize cross-version information.

To examine whether our representations capture information related with harmonies and pitches played, we perform the same evaluation procedure with boundaries from $\texttt{Ref}_{\texttt{H}}$ (see Figure 5). We obtain low F-measures for both CV and SV (dropping below 0.4 for kernel sizes above 20 seconds for both approaches). In particular, while we observe an advantage of CV over SV for capturing instrumentation, there is no such advantage with regard to

Scenario	AP	AUC	Micro Avg. F1	Micro Avg. S	Macro Avg. F1	Macro Avg. S
MFCC	0.777	0.780	0.600	0.890	0.450	0.847
SV	0.708	0.735	0.590	0.871	0.407	0.820
CV	0.753	0.795	0.657	0.872	0.514	0.835
Sup	0.838	0.881	0.772	0.894	0.714	0.874

Table 2: Results for different representation learning approaches when performing instrument classification.

Scenario	AP	AUC	Micro Avg. F1	Micro Avg. S	Macro Avg. F1	Macro Avg. S
Chroma	0.802	0.854	0.591	0.964	0.586	0.963
SV	0.427	0.568	0.001	1.000	0.001	1.000
CV	0.430	0.584	0.021	0.994	0.018	0.994
Sup	0.457	0.612	0.137	0.959	0.122	0.958

Table 3: Results for pitch-class classification using the learned representations.

pitch-classes. Additionally, standard Chroma features are clearly superior at capturing the structures in $\mathrm{Ref}_{\mathrm{H}}$. We conclude that the representations learned by our proposed approach CV indeed contain information about instrument texture rather than pitch-classes and harmonies.

5.4 Feature Analysis Using Classification

To gain further insights into the information captured by our learned representations, we also perform an indirect evaluation as typically done in representation learning. Previous studies often rely on training small classifiers on top of learned representations to investigate their usefulness for different downstream tasks [5, 15]. In this section, we complement our self-similarity-based analysis with such a classification-based evaluation strategy.

To this end, we pass individual representation vectors through a small network of dense layers with 128, 64, and 32 hidden units followed by leaky ReLU activations, respectively. The final layer produces outputs for every instrument annotated in our dataset, followed by a sigmoid activation. For each representation learning technique, we train and evaluate such a network using the dataset split as described in Section 4.1. Concretely, we minimize the mean binary cross-entropy loss over all instrument classes on the training set, using stochastic gradient descent with a learning rate of 10^{-4} for 10 epochs. We finally evaluate the classification results on the test set using standard metrics, including ranking-based average precision (AP), mean area under the ROC curves (AUC), F-measure (F1), and specificity (S). For F1 and S, we threshold the predicted probabilities at 0.5 and compute both micro and macro averages of the evaluation scores, where the macro average is not affected by imbalance among instrument classes.

The results of this experiment are shown in Table 2. We observe similar trends as in our self-similarity-based evaluation. As expected, the supervised baseline again yields best results. Our proposed cross-version approach CV clearly outperforms the traditional SV across all metrics (e. g., AP $= 0.753$ as opposed to 0.708 for SV). Furthermore, CV even improves upon the optimistic MFCC baseline in terms of AUC and F-measure (e. g., micro F1 $= 0.657$ instead of 0.600 for MFCC). Finally, SV performs worse than MFCC. Overall, the representations learned by our proposed approach CV are more effective for instrument classification compared to the standard SV approach that does not utilize cross-version information.

We repeat this experiment using pitch-classes as the classification targets instead of instruments. Table 3 shows the results of the modified experiment, which are inline with our conclusions from previous sections. Standard Chroma features strongly outperform all learned representations on this task. We conclude that our proposed approach captures instrumentation rather than pitches.

6. CONCLUSION

In this paper, we described a novel audio representation learning approach for cross-version music data and investigated its application to orchestral music. Our approach utilizes the correspondences between different versions of the same musical work. We showed qualitatively and quantitatively that the representations learned by our approach capture aspects of instrumentation. We outperform a standard training strategy that relies on temporal proximity alone.

Our approach can be applied to any kind of cross-version music dataset where alignments between versions can be obtained using standard music synchronization techniques. Future work may apply our approach to other musical scenarios and larger datasets, explore more complex feature extraction networks, investigate alternatives to our triplet loss formulation, or apply the learned representations in the context of different downstream tasks (such as structure analysis). One may also study the impact of design choices such as τ^{p} and τ^{n}, the pitch shifting augmentation, or the number of versions used for training.

Acknowledgments: This work was supported by the German Research Foundation (DFG MU 2686/7-2, MU 2686/11-2). The authors are with the International Audio Laboratories Erlangen, a joint institution of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and Fraunhofer Institute for Integrated Circuits IIS. The authors gratefully acknowledge the compute resources and support provided by the Erlangen Regional Computing Center (RRZE).

7. REFERENCES

[1] M. Müller, *Fundamentals of Music Processing – Using Python and Jupyter Notebooks*, 2nd ed. Springer Verlag, 2021.

[2] M. Müller, Y. Özer, M. Krause, T. Prätzlich, and J. Driedger, "Sync Toolbox: A Python package for efficient, robust, and accurate music synchronization," *Journal of Open Source Software (JOSS)*, vol. 6, no. 64, pp. 3434:1–4, 2021.

[3] M. C. McCallum, "Unsupervised learning of deep features for music segmentation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Brighton, UK, 2019, pp. 346–350.

[4] J. Wang, J. B. L. Smith, W. T. Lu, and X. Song, "Supervised metric learning for music structure features," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Online, 2021, pp. 730–737.

[5] J. Spijkervet and J. A. Burgoyne, "Contrastive learning of musical representations," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Online, 2021, pp. 673–681.

[6] C. Thomé, S. Piwell, and O. Utterbäck, "Musical audio similarity with self-supervised convolutional neural networks," in *Demos and Late Breaking News of the International Society for Music Information Retrieval Conference (ISMIR)*, Online, 2021.

[7] A. Saeed, D. Grangier, and N. Zeghidour, "Contrastive learning of general-purpose audio representations," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toronto, Canada, 2021, pp. 3875–3879.

[8] M. Buisson, B. McFee, S. Essid, and H. C. Crayencour, "Learning multi-level representations for hierarchical music structure analysis," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Bengaluru, India, 2022.

[9] M. A. V. Vásquez and J. A. Burgoyne, "Tailed U-Net: Multi-scale music representation learning," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Bengaluru, India, 2022.

[10] M. C. McCallum, F. Korzeniowski, S. Oramas, F. Gouyon, and A. Ehmann, "Supervised and unsupervised learning of audio representations for music understanding," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Bengaluru, India, 2022.

[11] B. Li and A. Kumar, "Query by video: Cross-modal music retrieval," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Delft, The Netherlands, 2019, pp. 604–611.

[12] I. Manco, E. Benetos, E. Quinton, and G. Fazekas, "Learning music audio representations via weak language supervision," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Singapore, 2022, pp. 456–460.

[13] H. Wu, C. Kao, Q. Tang, M. Sun, B. McFee, J. P. Bello, and C. Wang, "Multi-task self-supervised pre-training for music classification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toronto, Canada, 2021, pp. 556–560.

[14] P. Alonso-Jiménez, X. Serra, and D. Bogdanov, "Music representation learning based on editorial metadata from discogs," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Bengaluru, India, 2022.

[15] R. Castellon, C. Donahue, and P. Liang, "Codified audio language modeling learns useful representations for music information retrieval," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Online, 2021, pp. 88–96.

[16] H. Zhang, J. Tang, S. R. M. Rafee, S. Dixon, and G. Fazekas, "ATEPP: A dataset of automatically transcribed expressive piano performance," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Bengaluru, India, 2022.

[17] S. Ewert, M. Müller, V. Konz, D. Müllensiefen, and G. A. Wiggins, "Towards cross-version harmonic analysis of music," *IEEE Transactions on Multimedia*, vol. 14, no. 3-2, pp. 770–782, 2012.

[18] H. Schreiber, C. Weiß, and M. Müller, "Local key estimation in classical music audio recordings: A cross-version study on Schubert's Winterreise," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Barcelona, Spain, 2020, pp. 501–505.

[19] F. Zalkow and M. Müller, "Learning low-dimensional embeddings of audio shingles for cross-version retrieval of classical music," *Applied Sciences*, vol. 10, no. 1, 2020.

[20] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. B. Girshick, "Masked autoencoders are scalable vision learners," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, 2022, pp. 15 979–15 988.

[21] W. Hsu, B. Bolte, Y. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[22] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the International Conference on Machine Learning (ICML)*, Virtual, 2021, pp. 8748–8763.

[23] A. Guzhov, F. Raue, J. Hees, and A. Dengel, "AudioCLIP: Extending clip to image, text and audio," in *Proceedings of the IEEE International Conference on*

Acoustics, Speech, and Signal Processing (ICASSP), Singapore, 2022, pp. 976–980.

[24] M. Miron, J. J. Carabias-Orti, J. J. Bosch, E. Gómez, and J. Janer, "Score-informed source separation for multichannel orchestral recordings," *Journal of Electrical and Computer Engineering*, vol. 2016, no. 8363507, 2016.

[25] Z. Tang and D. A. A. Black, "Melody extraction from polyphonic audio of Western opera: A method based on detection of the singer's formant," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Taipei, Taiwan, October 2014, pp. 161–166.

[26] M. Taenzer, J. Abeßer, S. I. Mimilakis, C. Weiß, H. Lukashevich, and M. Müller, "Investigating CNN-based instrument family recognition for Western classical music recordings," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Delft, The Netherlands, 2019, pp. 612–619.

[27] Y.-N. Hung and Y.-H. Yang, "Frame-level instrument recognition by timbre and pitch," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, 2018, pp. 135–142.

[28] S. Gururani, C. Summers, and A. Lerch, "Instrument activity detection in polyphonic music using deep neural networks," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, 2018, pp. 569–576.

[29] Y. Han, J. Kim, and K. Lee, "Deep convolutional neural networks for predominant instrument recognition in polyphonic music," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 1, pp. 208–221, 2017.

[30] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 815–823.

[31] A. Flexer, "A closer look on artist filters for musical genre classification," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Vienna, Austria, 2007, pp. 341–344.

[32] R. M. Bittner, B. McFee, J. Salamon, P. Li, and J. P. Bello, "Deep salience representations for F0 tracking in polyphonic music," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Suzhou, China, 2017, pp. 63–70.

[33] C. Weiß, J. Zeitler, T. Zunner, F. Schuberth, and M. Müller, "Learning pitch-class representations from score–audio pairs of classical music," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Online, 2021, pp. 746–753.

[34] H. Terasawa, M. Slaney, and J. Berger, "The thirteen colors of timbre," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, 2005, pp. 323–326.

[35] J. Foote, "Automatic audio segmentation using a measure of audio novelty," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, New York, NY, USA, 2000, pp. 452–455.