Self-Supervised Contrastive Learning for Robust Audio – Sheet Music Retrieval Systems

Luís Carvalho*
Institute of Computational Perception
Johannes Kepler University Linz
Austria
luis.carvalho@jku.at

Tobias Washüttl
Institute of Computational Perception
Johannes Kepler University Linz
Austria

Gerhard Widmer
Institute of Computational Perception
LIT Artificial Intelligence Lab
Johannes Kepler University Linz
Austria

ABSTRACT

Linking sheet music images to audio recordings remains a key problem for the development of efficient cross-modal music retrieval systems. One of the fundamental approaches toward this task is to learn a cross-modal embedding space via deep neural networks that is able to connect short snippets of audio and sheet music. However, the scarcity of annotated data from real musical content affects the capability of such methods to generalize to real retrieval scenarios. In this work, we investigate whether we can mitigate this limitation with self-supervised contrastive learning, by exposing a network to a large amount of real music data as a pre-training step, by contrasting randomly augmented views of snippets of both modalities, namely audio and sheet images. Through a number of experiments on synthetic and real piano data, we show that pre-trained models are able to retrieve snippets with better precision in all scenarios and pre-training configurations. Encouraged by these results, we employ the snippet embeddings in the higher-level task of cross-modal piece identification and conduct more experiments on several retrieval configurations. In this task, we observe that the retrieval quality improves from 30% up to 100% when real music data is present. We then conclude by arguing for the potential of self-supervised contrastive learning for alleviating the annotated data scarcity in multi-modal music retrieval models. Code and trained models are accessible at https://github.com/luisfvc/ucasr.

CCS CONCEPTS

• **Information systems** → **Music retrieval**; • **Computing methodologies** → **Machine learning**.

KEYWORDS

multi-modal embedding spaces; audio–sheet music retrieval

ACM Reference Format:
Luís Carvalho, Tobias Washüttl, and Gerhard Widmer. 2023. Self-Supervised Contrastive Learning for Robust Audio – Sheet Music Retrieval Systems. In *Proceedings of the 14th ACM Multimedia Systems Conference (MMSys '23), June 7–10, 2023, Vancouver, BC, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3587819.3590968

*Corresponding author.

MMSys '23, June 7–10, 2023, Vancouver, BC, Canada

ACM ISBN 979-8-4007-0148-1/23/06.
https://doi.org/10.1145/3587819.3590968

1 INTRODUCTION

Extensive amounts of music-related contents are available nowadays in the digital domain, in diverse forms, including studio and live audio recordings, album covers, scanned sheet music, meta-data, and video clips. In addition, some of these contents are normally catalogued and/or curated with manual effort by organizations from different contexts, such as cultural institutes, digital libraries, music publishers and concert halls. Making such heterogeneous collections searchable and explorable in an automated and content-based way requires powerful technologies for cross-linking between items of different modalities. As an example, a musician may have an incomplete excerpt of an unlabeled recording and wishes to retrieve from a digital database all relevant items in all possible modalities that are related to the query excerpt.

A fundamental and challenging problem in many cross-modal music retrieval scenarios is referred to as *audio–score retrieval* [28]. This problem is centered around two modalities, the acoustic one which comprises audio content, and the respective visual counterparts represented by music scores. This task is defined as follows: given a query snippet in one modality (a short audio excerpt, for example), retrieve the corresponding item from a database in the other modality (a music score). The most extreme and realistic setup for this task occurs when no metadata or machine-readable information of any kind (such as MusicXML or MIDI data) is available. In this case, one is restricted to searching and retrieving only raw material, that is, audio recordings and digitized images of scanned sheet music. With the aforementioned constraints, the underlying problem can be referred to as *audio–sheet music retrieval* [15] and is illustrated in Figure 1.

Audio–score retrieval has been explored in numerous works in both intra- and inter-document scenarios and its applications are manifold. The former scenario applies when both audio and score are known beforehand, and one wishes to obtain a fine-grained mapping between them. This can be applied to score following [21], which is the real-time tracking of musical performances in the corresponding score, that can be employed in automatic sheet music page turning [5] for example. Another use includes audio–score alignment [2], where one wishes, for instance, to query a sequence of measures from a score and find the corresponding sections within an audio interpretation of the same piece [18].

As for inter-document cases, the majority of scenarios address the task of *audio–score piece identification* [6], which can be defined both search directions: given an audio query, find a corresponding score from a collection; and given an excerpt from a score, retrieve an appropriate audio recording. In the realm of modern digital

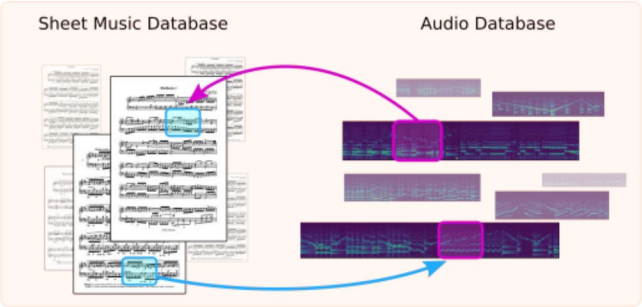

Figure 1: Illustration of the audio-sheet music snippet retrieval problem for both search directions. First, one wishes to query an audio excerpt (on the right), represented by its magnitude spectrogram, and retrieve the corresponding sheet music snippet from an image database (on the left). Analogously, one may wish to invert the search direction and retrieve items from an audio database given a sheet music snippet input. All music visualizations were extracted from the MSMD dataset [15].

music libraries, such technologies play an essential role in the indexing, navigation, browsing, and synchronization of multi-modal databases [28]. One example of such applications is the Piano Music Companion [3], a system that first tries to identify a piano piece that is being played, followed by synchonizing it within the corresponding score in real time. However, a critical limitation of these systems is that they require the score to be available in a symbolic, machine-readable form – e.g., MIDI or MusicXML – which is a serious problem in practical applications.

Recent approaches for snippet-level audio–sheet music retrieval attempt to overcome this limitation by learning low-dimensional embeddings directly from the multi-modal data – audio and scans or photographs of scores [7, 14, 15]. This is done by training a cross-modal deep convolution neural network (CNN) to project audio and score image snippets onto a shared space where semantically similar items of the two modalities will end up close together, whereas dissimilar ones far apart.

Being of a fully supervised nature, this approach has a number of limitations. First, it requires a large amount of labeled training data in order for a model to generalize to unseen data. Second, such annotated data is of complex and expensive nature: it requires fine-grained alignments between time stamps on the audio signal and pixel coordinates in sheet music images in order to obtain matching cross-modal snippets. The annotation process, besides being labor- and time-consuming, requires annotators with specialized musical training who are able to correctly identify and interpret music notation in sheet music images and match them to note onsets in audio recordings. For that reason current approaches rely solely on synthetic datasets, where both the scores and the audios – and the corresponding annotations – are generated from a symbolic score representation; this results in poor generalization to real data, as we will demonstrate in our experiments (see Section 5).

In this paper, we explore *self-supervised contrastive learning* as way to mitigate the data scarcity problem in audio–sheet music snippet retrieval. We propose to contrast differently augmented versions of short fragments of audio recordings and sheet music images, as a pre-training step. The data for this task needs no labels or annotations, so we have an almost infinite supply of this. The key idea is that by trying to solve the pretext problem, the model can learn useful low-level representations, which can then be used for the audio–sheet music snippet retrieval task, where only few annotated data are available.

We conduct several experiments in datasets of different natures to demonstrate that the pre-training stage effectively alleviates the performance gap between synthetic and real data. We then use the learned snippet embeddings for the downstream task of cross-modal *piece identification* and observe improved retrieval performance in all models that were pre-trained. We summarize our contributions as follows.

- We design a method for multi-modal self-supervised contrastive learning of audio–sheet music representations with publicly available music data, where the network responsible for each modality can be independently pre-trained and enabled for fine-tuning.
- We show through detailed experiments on diverse datasets that our models outperform the current state-of-the-art method by a significant margin in the task of snippet retrieval.
- As a proof of concept, we aggregate snippet embeddings to perform cross-modal piece identification and demonstrate the effectiveness of our improved models, which significantly outperform fully supervised methods.

2 RELATED WORK

One of the key challenges in audio–sheet music retrieval refers to its multi-modality nature: finding some shared representation that allows for an easy comparison between items from different modalities. The traditional methods for connecting printed scores to their relative audio recordings are based on common mid-level representations [23, 28], such as chroma-based features [6, 18], symbolic representations [4], or the bootleg score [39, 41], the latter being defined as a coarse codification of sequences of the main note-heads in a printed score. However generating these mid-level representations involves pre-processing stages which are prone to error, such as optical music recognition [9, 27, 40] and automatic music transcription [8, 34].

In order to avoid such unreliable pre-processing components, an alternative approach was proposed in [14, 15], by designing a two-modal network that is able to learn a shared space between short fragments of score scans and their corresponding audio excerpts. This is done by training the network to minimize the cosine distance between pairs of low-dimensional embeddings from snippets of audio and sheet music, and promising results on synthetic music data indicate the potential of replacing manually-designed common representations with learned spaces.

3 THE PROPOSED METHOD

In this section we first briefly describe how current approaches employ deep CNNs to learn a cross-modal embedding space from pairs of matching audio and sheet music snippets. Then we explain our proposed method, followed by describing the augmentation strategies for both sheet music and audio samples.

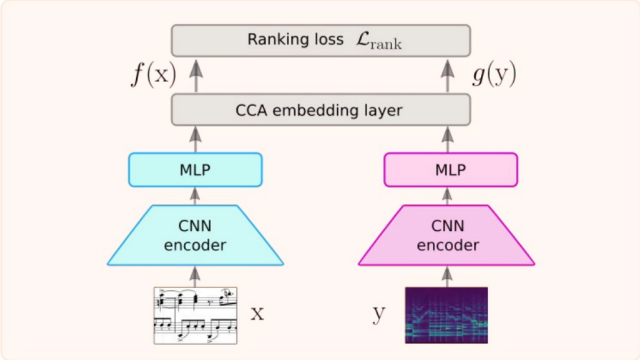

Figure 2: Illustration of audio–sheet retrieval model, adapted from [15]. The left and right independent pathways encode sheet music and audio snippets, respectively, by projecting together corresponding cross-modal pairs while maximizing the distance between non-corresponding ones.

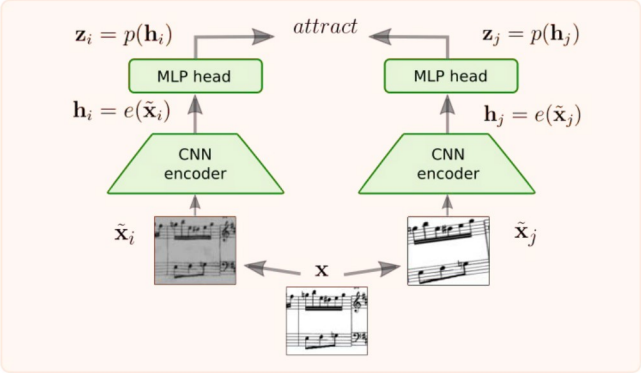

Figure 3: Sketch of our proposed self-supervised contrastive learning architecture, adapted from [11], for an example of sheet music snippet. Two different views are rendered using augmentation transforms (contrast and rotation, for example), which are fed to a CNN encoder followed by a MLP head, generating a positive pair of embeddings $(\mathbf{z}_i, \mathbf{z}_j)$, which should be projected close together.

3.1 Learning Audio–Sheet Music Embeddings

The fundamental approach to learn correspondences between short snippets of music recordings and sheet music images was first proposed in [14, 15]. This task is formulated as a cross-modal embedding learning problem, where a network is trained to optimize a shared space between the two modalities, by minimizing the cosine distance between musically similar snippets whereas maximizing the distance between non-corresponding items.

The network, which is illustrated in Figure 2, consists of two independent pathways, each responsible for one modality. Each pathway is composed of a VGG-style encoder [36], followed by a multi-layer perceptron layer (MLP) that learns higher-level abstractions from the encoder output. At the top of the network a canonically correlated (CCA) embedding layer [16] is placed, forcing the two pathways to learn representations that can be projected onto a 32-dimensional shared space.

Then a pairwise ranking loss [25] is employed to minimize the distance between embeddings from matching snippets of different modalities. Let $(\mathbf{x}, \mathbf{y})$ represent a pair of corresponding sheet music and audio snippets (positive pairs), as displayed in Figure 2. The sheet music pathway is represented by the function f, while g denotes the audio embedding function. The functions f and g map $\mathbf{x}$ and $\mathbf{y}$ to the shared low-dimensional space. Then the similarity function $\text{sim}(\cdot)$, defined as the cosine similarity, is used to compute the final ranking loss :

$$\mathcal{L}_{\text{rank}} = \sum_{(\mathbf{x},\mathbf{y})} \sum_{k=1}^{K} \max\Big\{0, \alpha - \text{sim}\Big(f(\mathbf{x}), g(\mathbf{y})\Big) + \text{sim}\Big(f(\mathbf{x}), g(\mathbf{y}_k)\Big)\Big\}, \tag{1}$$

where $\mathbf{y}_k$ for $k \in 1, 2, \ldots, K$ represent additional contrastive (negative) examples from K non-matching snippets within the same training mini-batch. This ranking loss is applied on all $(\mathbf{x}, \mathbf{y})$ pairs of each mini-batch iteration, and the margin parameter $\alpha \in \mathbb{R}_+$ in combination with the max $\{\cdot\}$ function penalize matching snippets that were poorly embedded.

After the training is done, the snippet retrieval task illustrated in Figure 1 can then be easily and efficiently performed via nearest-neighbor search in the shared space.

3.2 Self-Supervised Contrastive Learning

In this work we build on the SimCLR framework [11], a self-supervised contrastive method for image representations. The goal is to learn useful representations from unlabeled data using self-supervision . The idea is to train a network encoder to be as invariant as possible concerning a set of given augmentation transforms [17]. In order to do that, different augmentations are applied to a training sample so two distinct views thereof are generated (which constitute a "positive pair" that represent the same item). Then a Siamese network [12] encodes both views into embeddings, and a contrastive loss function is applied in order to bring together latent representations from the same sample, while pushing away embeddings of negative pairs.

This approach is sketched in Figure 3 for the case of sheet image snippets, however we stress the procedure is analogous for the audio case. More precisely, the following steps are performed:

- Given a sample $\mathbf{x}$ from the training mini-batch, two stochastic sets of data augmentation transforms are applied to $\mathbf{x}$ to render two different augmented views of the same sample (a "positive pair"), namely $\tilde{\mathbf{x}}_i$ and $\tilde{\mathbf{x}}_j$. (Our specific data augmentation pipeline for each modality is described in Section 4 below.)
- Then a CNN encoder $e(\cdot)$ is used to compute a latent representation $\mathbf{h}_i = e(\tilde{\mathbf{x}}_i)$ for each view.
- An MLP projection head $p(\cdot)$ maps the encoder latent embedding $\mathbf{h}_i$ to a final space $\mathbf{z}_i = p(\mathbf{h}_i)$ where the contrastive loss is used.
- Then the normalized-temperature cross-entropy ($NT-Xent$) loss function [37] is applied and summed over all positive

augmented pairs $(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j)$ within the mini-batch:

$$\mathcal{L} = \sum_{(i,j)} \log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{v=1}^{2N} \mathbb{1}_{[v \neq i]} \exp(\text{sim}(z_i, z_v)/\tau)}. \quad (2)$$

During training, two different augmented views are rendered from each sample within the mini-batch of size N, yielding a pool of $2N$ augmented views per mini-batch, over which the above loss function is applied. For each positive pair, all remaining $2N - 2$ samples within the mini-batch are considered negative samples, as indicated by the summation in the denominator of Equation (2). The temperature parameter $\tau \in \mathbb{R}_+$ works similarly to the margin parameter α in Equation (1), prioritizing poorly embedded samples.

In this architecture the MLP projection layer p is employed only during self-supervised learning. After the model is trained, this layer is discarded and only the encoder e is used as a pre-trained model for a given downstream task, which in our case is audio–sheet music retrieval. As discussed in [11], the reason is that empirical results show that applying the contrastive loss over a function p of the encoder embeddings $\mathbf{z}_i = p(\mathbf{h}_i)$ during training is beneficial because it improves the quality of learned representations.

An important difference between our approach and the method described in [11] is that in our setup we have two separate convolutional pathways, one responsible for encoding each modality (see Figure 2). We perform self-supervised contrastive learning separately in each of the modalities, in order to obtain two separate and independent pre-trained encoders. Since the pathways for audio and sheet music are independent, we can simply select the modality we wish to pre-train, and obtain a pre-trained encoder for the given modality. The encoder is then placed in the multi-modal network in Figure 2 and fine-tuned for the audio–sheet music retrieval task.

Our CNN encoder follow the setup in [15]. The encoder architecture is the same in each modality, and consists of a VGG-style network [36] with eight convolutional layers, each of them followed by a batch normalization layer [22] and exponential linear unit (ELU) [13] activation. A max pooling layer is applied every two consecutive convolutional layers in order to halve the dimensions of the hidden representations.

Our projection head p consists of an MLP with one hidden layer followed by batch normalization and rectified linear unit activation (ReLU) [1], from which the output embedding is L2-normalized and mapped to a 32-dimensional final representation, on which the contrastive loss is calculated.

4 DATA AUGMENTATIONS

In self-supervised learning, one wishes to optimize a model so it can be highly invariant in regards to a set of augmentation transforms. Therefore a proper composition of data augmentation operations is crucial for learning good representations [11]. In our system, an augmented view $\tilde{\mathbf{x}}_i$ is generated by applying a pipeline of M augmentation transforms on the original sample $\mathbf{x}$. Each augmentation transform $t_m()$ has an independent probability p_m to be applied to $\mathbf{x}$. Each time the transform $t_m()$ is selected, its hyper-parameters are stochastically sampled from a pre-defined distribution, which is particular for each transform.

In the following we provide details of the augmentations we employed during the self-supervised training of each modality, as well as information about the used datasets.

4.1 Sheet Music Augmentation Transforms

Augmentation strategies have proven to be powerful techniques to help machine learning models generalize to unseen data in image tasks [26, 33]. In sheet music analysis, augmentation transforms are chosen so that they can emulate document variations and degradations of various types [9, 27, 40]. We build on these works and define a set of nine transforms that are applied to the sheet music snippets, which are described as follows.

- We shift the snippet horizontally (1) and vertically (2) in relation to its positive pair. The horizontal shift is calculated in a way that positive pairs will share at least 80% of their content, and 75% for the vertical shift.
- The snippet is resized (3) to have between 90 and 110% of its original size.
- The snippet is rotated (4) to a maximum angle of 8 degrees, counter- or clockwise.
- We apply Additive White Gaussian Noise (AWGN) (5) and Gaussian Blur (GB) (6), to simulate noisy documents and poor resolution, respectively.
- Additive Perlin Noise (APN) (7) [30] is added to the sample. This transform generates big darker and lighter areas in the image, mimicking quality differences in the image snippet.
- Then random small (8) and large (9) elastic deformations (ED) [35] are applied, generating wave-like distortions to the image, whose strength and smoothing can be tuned. Small EDs are applied on small scales, with the effect of deforming the shapes of smaller symbols and lines. When large EDs are applied, the structure and orientation of bigger music symbols are modified, for example by skewing or bending bar lines and note symbols, and squeezing or elongating note heads.

The augmentations are applied in the presented order and we tune the hyper-parameters of each individual transform in a way that a snippet is highly degraded, but still legible. Figure 4 shows four examples of augmented snippet pairs when all nine transforms are stochastically applied to four sheet music snippets.

4.2 Audio Augmentation Transforms

Several works have successfully explored data augmentation for several audio and music learning tasks [29, 31, 32, 38]. We build on them and in the following define the sequence of eight audio transforms used to augment audio excerpts.

- We apply a time shift (1) between the two excerpts of a positive pair. The shift is calculated in a way that corresponding snippets will share at least 70% of their content.
- Polarity inversion (2) is applied to the audio excerpt by multiplying its amplitude by −1.
- Additive White Gaussian Noise (3) with a signal-to-noise ratio between 5 and 20 dB is added.
- A gain change (4) between −12 and 12 dB is applied to the signal.

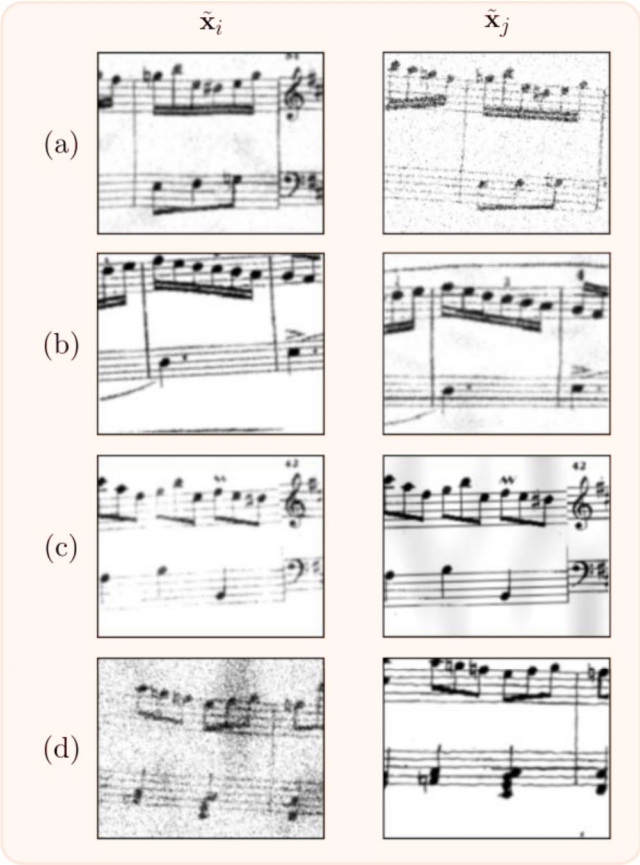

Figure 4: Examples of four pairs of augmented sheet music snippets after all nine transforms were randomly applied. One should note that, even though the excerpts were greatly corrupted, they are still readable. These examples were obtained from the MSMD dataset [15].

- We apply a seven-band parametric equalizer (5) in order to adjust the volume of seven different randomly-chosen frequency bands. [1]
- The audio excerpt is stretched in time (6) without modifying its pitch by changing the tempo with a random factor between 0.5 and 1.8.
- Time (7) and frequency (8) masks are applying to the audio snippet *à la* SpecAugment [29]. Both time and frequency largest masks correspond to 20% of the snippet duration and frequency range, respectively.

The augmentations are applied in the order they were declared above. The transforms 1-5 are applied directly on the waveform snippets, while transforms 6-8 are applied in the frequency domain due to computational benefits.

[1]https://iver56.github.io/audiomentations/waveform_transforms/seven_band_parametric_eq/

5 EXPERIMENTS AND RESULTS

In this section, we report on the experiments conducted to validate our proposed method. We first briefly elaborate on the preprocessing steps, dedicated datasets and training setup. Then we carry out experiments on cross-modal snippet retrieval and piece identification.

5.1 Snippet Preparation

In the following, we describe how the snippets are extracted, preprocessed and prepared for training.

5.1.1 Sheet Music Snippets. Our sheet music images are first rescaled to a 1181×835 resolution (pixels) per page. Then 160×200 snippets are selected in such a way that they comprise musical content, *i.e.* within the systems of the document (groups of two staves, for piano sheet music). When no annotation is available concerning pixel coordinates of note heads and/or system locations (*i.e.*, in the raw data for self-supervised learning), we use the Audiveris engine [2] to automatically detect the staff lines as a pre-processing stage. Manual inspections showed that Audiveris is able to properly identify system coordinates in printed piano scores with accuracy of over 99%, therefore it is unlikely that snippets will not exhibit musical content. Examples of sheet music snippets are depicted in Figures 2, 3 and 4

5.1.2 Audio Snippets. Our music datasets consist of audio signals with a sampling rate of 22.05 kHz. The log-frequency spectrogram of each signal is computed with a resolution of 20 frames per second and minimum and maximum frequencies of 30 Hz and 6 kHz respectively, generating 92 frequency bins per frame. We then cut out 84 frames of audio (approximately 4.2 seconds) to generate a snippet, which has a final shape of 92×84 (bins $\times$ frames). Examples of audio log-spectrograms and snippets are shown in Figures 1 and 2.

5.2 Datasets

To pre-train the sheet music encoder, we scrape data from the International Music Score Library Project (IMSLP) [3], an online plataform that hosts public domain music scores. We collect 3,485 scanned piano scores relating to 842 music pieces by several composers, which amounts to approximately 7,000 pages of sheet music. From these documents we extract over 700k snippets as described in 5.1 for training and validation. We will provide the IMSLP links to all music scores of our collection in the paper repository [4].

For self-supervised learning of the audio encoder, we use the recordings from MAESTRO [19], a public dataset with 1,276 piano recordings comprising around 200 hours of piano music. Since there is no test stage at pre-training, we merge the pre-defined MAESTRO test split into the train set, and generate around 840k audio snippets as described in 5.1 to train and validate the audio encoder.

To train the final audio–sheet music network, we use the Multi-Modal Sheet Music Dataset (MSMD) [15], which is a database of polyphonic piano music and scores. With over 400 pieces covering over 15 hours of audio, this dataset has fine-grained cross-modal

[2]https://audiveris.github.io/audiveris/
[3]https://imslp.org/wiki/Main_Page
[4]https://github.com/blinded_for_review.

alignments between audio note onsets and sheet music note-head coordinates, which makes it suitable for generating matching audio–sheet music snippets. This dataset is of fully artificial nature: audio recordings are synthesized from MIDI files using FluidSynth [5] and the scores are engraved with LilyPond [6]. The matching snippets are extracted in a way that they are centred around the same note event, being the note onset for the audio side and the note-head pixel coordinate for the sheet music side.

In our experiments, we wish to investigate how well pre-training helps to generalize from synthetic to real data. To this end, we evaluate on three datasets: on a (1) fully artificial one, and on datasets consisting (2) partially and (3) entirely of real data. For (1) we use the test split of MSMD and for (2) and (3) we combine the Zeilinger and Magaloff Corpora [10] with a collection of commercial recordings and scanned scores that we have access. These data account for more than a thousand pages of sheet music scans with fine-grained mappings to both MIDI files and over 20 hours of classical piano recordings. We then define the following evaluation sets. (2) *RealScores_Synth*: a partially real set, with *scanned* (real) scores of around 300 pieces aligned to notes of *synthesized* MIDI recordings. And (3) *RealScores_Rec*: an entirely real set, with *scanned* (real) scores of around 200 pieces with fine-grained alignments to *real audio* recordings.

5.3 Training Setup

Our learning pipeline is split into two stages: (i) self-supervised learning on each individual modality with a batch size of 256, followed by (ii) cross-modal training on pre-loaded encoders from either or both modalities, with a batch size of 128 pairs, where audio and sheet music snippets are project onto a 32-dimensional space.

In both stages we use the Adam optimizer [24] and He initialization [20] in all convolutional layers. The temperature parameter τ and triplet margin α are set to 0.5 and 0.6, respectively. We set the initial learning rates of (i) and (ii) to 0.001 and 0.0004 respectively. We observe the validation loss during training and halve the learning rate if there are no improvements over 10 consecutive epochs, apply early stopping when halving happens five times, and select the best model among all epochs for testing. For sake of simplicity, we leave the remaining details concerning topological design of the networks, further learning hyper-parameters, and augmentation probabilities and hyper-parameters, to our repository. [4]

5.4 Snippet Retrieval Experiments

In this section we evaluate a two-way snippet retrieval task: given a query excerpt, retrieve the corresponding snippet in the other modality. This is done by first embedding the query excerpt and all snippets of the target modality, and then selecting the query's nearest neighbor in the embedding space as the best match, based on their pairwise cosine distance.

For each of the three evaluation datasets introduced in section 5.2, we select a pool of 10,000 audio–sheet music snippet pairs for evaluation. We perform the retrieval task in both search directions: audio-to-sheet music (A2S) and sheet music-to-audio (S2A).

[5] https://www.fluidsynth.org/
[6] http://lilypond.org/

Table 1: Comparison of snippet retrieval results in both query directions on three types of datasets: (I) fully synthetic, (II) partially real and (III) entirely real. Boldfaced rows represent the best performing model per dataset.

	Audio-to-Score (A2S)				Score-to-Audio (S2A)			
	R@1	**R@25**	**MRR**	**MR**	**R@1**	**R@25**	**MRR**	**MR**
I MSMD (Fully synthetic)								
BL	0.54	0.91	0.653	1	0.60	0.94	0.704	1
BL+A	**0.59**	**0.93**	**0.699**	**1**	**0.61**	**0.95**	**0.723**	**1**
BL+S	0.56	0.92	0.676	1	0.61	0.94	0.717	1
BL+A+S	0.57	0.93	0.687	1	0.60	0.94	0.718	1
II RealScores_Synth (Sheet music scans and synthetic recordings)								
BL	0.28	0.67	0.375	7	0.36	0.77	0.467	3
BL+A	0.37	0.78	0.478	3	0.43	0.82	0.537	2
BL+S	0.34	0.75	0.447	4	0.43	0.84	0.544	2
BL+A+S	**0.37**	**0.79**	**0.481**	**3**	**0.44**	**0.84**	**0.548**	**2**
III RealScores_Rec (Sheet music scans and real recordings)								
BL	0.10	0.36	0.156	76	0.14	0.47	0.216	33
BL+A	0.13	0.44	0.200	41	0.17	0.55	0.261	18
BL+S	0.12	0.42	0.192	47	0.18	0.54	0.259	18
BL+A+S	**0.15**	**0.48**	**0.226**	**29**	**0.18**	**0.54**	**0.266**	**18**

As evaluation metrics we compute the *Recall@k* (R@k), *Mean Reciprocal Rank* (MRR) and the *Median Rank* (MR) for each experiment. The R@k measures the ratio of queries which were correctly retrieved within the top k results. The MRR is defined as the average value of the reciprocal rank over all queries, with the rank being the position of the correct match in the distance-ordered ranked list of candidates. MR is the median position of the correct match in the ranked list.

We perform snippet retrieval with the state-of-the-art method [15], which will be denoted as the baseline *BL*, and compare with all possible combinations of self-supervised pre-training as we proposed. Since in the cross-modal network the two convolutional pathways responsible for encoding each modality are independent, we can load either or both encoders with parameters that were pre-learned before training. We then define the following models:

- *BL+A*: the audio encoder is pre-trained
- *BL+S*: the sheet music encoder is pre-trained
- *BL+A+S*: both audio and sheet music encoders are pre-trained,

which are modified versions of the baseline.

Table 1 presents the snippet retrieval results of the four models defined above, evaluated on both search directions A2S and S2A. In the first section (I) we examine the completely synthetic set defined as the MSMD test split. Then in sections (II) and (III) we consider the partially and completely real scenarios, where audio excerpts are extracted from synthetic and real recordings, respectively, and sheet music snippets are derived from scans of real scores in both setups.

We first observe the performance of the current state-of-the-art model (*BL*) dropping sharply when moving from artificial to real data. In the fully synthetic set (I) it achieves MRRs of 0.653 and 0.704 in the directions A2S and S2A, respectively, correctly retrieving approximately 60% of the snippets as the best match in the S2A

task. The MRR drops at least 23% points for either A2S or S2A as we move to (II) and at least 48% at (III). The most extreme drop occurs at (III) in the A2S task: only 10% of the score snippets are on rank 1 (R@1 = 0.10). We additionally note that the retrieval quality of the S2A search direction is better than that of A2S for all evaluation metrics.

Our proposed models outperform the baseline in all scenarios for all evaluation metrics, indicating that self-supervising pre-training of either modality is beneficial in the problem we attempt to solve. We derive the following observations and discussions:

- The most significant improvements were observed in configurations with real music data, namely (II) and (III). We argue for the modest improvements on (I): the synthesized data of MSMD do not exhibit the degradations simulated by the augmentations transforms described in Sec 4, for either scores or recordings. Therefore it was not expected that our pre-training strategy would considerably benefit retrieval on artificial data.
- Pre-training both audio and score encoders (*BL+A+S*) generated the best retrieval metrics in scenarios with real data, with the largest improvements being observed in (II), where the MRR of the A2S and S2A tasks were increased by roughly 10% and 8% points, correspondingly. Moreover, it was not observed a substantial compound effect of pre-training both encoders (*BL+A+S*) when comparing to individual encoders (*BL+A* and *BL+S*): the improvements were merely marginal.
- In addition to the absolute improvements, the performance drop between evaluations on synthesized and real datasets was reduced: The MRR gap when moving from (I) to (II) and to (III) reduced by 7.2% and 2.6% points for the A2S direction, respectively; when retrieving the S2A direction these drops accounted for 6.7% and 3.6% points, correspondingly.
- The best models also reduced the overall performance gap between retrieval directions A2S and S2A, in all dataset configurations.

In an additional experiment, we take a closer look at the shared space properties of mapping matching snippets close together. Figure 5 depicts the distribution of pairwise cosine distances between 2,000 snippet pairs across each test dataset. When jointly analyzing Table 1, we observe that models which are capable of producing smaller distances between matching fragments generate better snippet retrieval quality. Moreover, we see that distances between snippet pairs from real data are on average mapped farther to each other than those from synthesized music data.

In all experimental scenarios, our pre-trained models were able to project corresponding snippets in the embedding space closer together, in comparison with the state-of-the-art method. From this we can point out the potential of self-supervised pre-training as a key component towards more powerful joint embedding spaces.

5.5 Cross-modal Piece Identification Experiments

In this set of experiments, we aggregate snippet embeddings generated by our models to perform cross-modal piece identification: given a full recording as a query, retrieve the corresponding score from a collection; or given a printed score, find an appropriate music

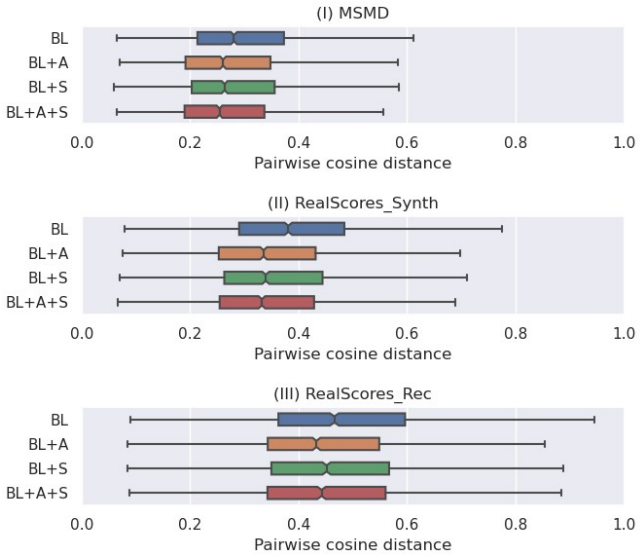

Figure 5: Distributions of pairwise cosine distances between corresponding pairs of audio-sheet music snippets, with 2,000 pairs randomly drawn from each evaluation set. Outliers are not directly visualized in order not to distort the plots. The vertical lines highlight the medians of the distribution of the baseline model *BL* for each dataset.

recording within a database. We evaluate this task as a proof-of-concept, to validate our proposed methods in a higher-level realistic retrieval scenario. As underlined in Section 1, piece identification is a key component of many audio–score retrieval systems, so we believe this evaluation can give us insights towards more robust systems.

The piece identification is done as in [15], with an approach that we will denote as *vote-based*: a matching procedure purely based on snippet retrieval and indexing. Let $\mathcal{D}$ be a collection of L documents and Q be a document query in the target and search modalities, respectively. Each document $D_i \in \mathcal{D}$ is sequentially cut into snippets, which are embedded via their respective pathway network of Figure 2, generating a set of embeddings $\{d_1^i, d_2^i, ..., d_{M_i}^i\}$, where each embedding d_j^i is indexed to its originating document D_i. We define hop sizes of 50 pixels and 10 frames (roughly 0.5 sec) for consecutive sheet music and audio snippets.

The document query is segmented into 100 equally-spaced excerpts, which are passed through the counterpart pathway of the model, producing a set of query embeddings $\{q_1, q_2, ..., q_{100}\}$. Then snippet retrieval is carried out as in Section 5.4 for each query embedding q_j, with the difference that now the top 25 nearest neighbors are retrieved per query embedding among all embeddings from the collection $\mathcal{D}$. Each nearest neighbor votes for its originating document, and a vote-based ranked list is created by aggregating all nearest neighbors from all 100 query embeddings. The document achieving the highest count among all 2,500 votes is selected as the best match.

In our piece identification experiments we evaluate on pieces of the same datasets as in Section 5.4. (I) The MSMD test split has 100

Table 2: Comparison of audio–sheet music piece identification results in both query directions on three types of datasets: (I) fully synthetic, (II) partially real and (III) entirely real. Boldfaced rows represent the best performing model per dataset.

		Audio-to-Score (A2S)				Score-to-Audio (S2A)			
	#	R@1	R@10	>R@10	MRR	R@1	R@10	>R@10	MRR
I MSMD (Fully synthetic)									
BL	100	0.76 (76)	0.98 (98)	0.02 (2)	0.846	0.87 (87)	1.00 (100)	0.00 (0)	0.927
BL+A	100	0.85 (85)	0.99 (99)	0.01 (1)	0.910	0.81 (81)	1.00 (100)	0.00 (0)	0.896
BL+S	100	0.84 (84)	1.00 (100)	0.00 (0)	0.898	0.87 (87)	1.00 (100)	0.00 (0)	0.928
BL+A+S	100	**0.87 (87)**	**1.00 (100)**	**0.00 (0)**	**0.918**	**0.93 (93)**	**1.00 (100)**	**0.00 (0)**	**0.961**
II RealScores_Synth (Sheet music scans and synthetic recordings)									
BL	314	0.49 (154)	0.84 (265)	0.16 (49)	0.609	0.65 (203)	0.90 (282)	0.10 (32)	0.734
BL+A	314	0.71 (223)	0.94 (294)	0.06 (20)	0.792	0.82 (256)	0.98 (307)	0.02 (7)	0.874
BL+S	314	0.70 (219)	0.93 (291)	0.07 (23)	0.781	0.82 (256)	0.97 (306)	0.03 (8)	0.871
BL+A+S	314	**0.80 (250)**	**0.96 (302)**	**0.04 (12)**	**0.857**	**0.88 (277)**	**0.98 (308)**	**0.02 (6)**	**0.919**
III RealScores_Rec (Sheet music scans and real recordings)									
BL	198	0.11 (22)	0.57 (113)	0.43 (85)	0.256	0.48 (95)	0.79 (156)	0.21 (42)	0.587
BL+A	198	0.21 (42)	0.69 (136)	0.31 (62)	0.361	0.62 (122)	0.87 (173)	0.13 (25)	0.714
BL+S	198	0.22 (44)	0.69 (137)	0.31 (61)	0.375	0.63 (125)	0.88 (175)	0.12 (23)	0.721
BL+A+S	198	**0.39 (78)**	**0.81 (161)**	**0.19 (37)**	**0.535**	**0.72 (143)**	**0.94 (187)**	**0.06 (11)**	**0.795**

pairs of both synthesized scores and their respective recordings; (II) has 314 pieces with their corresponding scanned sheet music and synthesized recordings; and (III) has 198 pairs of scanned sheet music and real recordings.

The cross-modal piece identification results are summarized in Table 2. We evaluate the same scenarios and models as for the two-way snippet retrieval task, in both search directions A2S and S2A. Moreover we include in the table (between parentheses) the actual number of pieces retrieved for each recall value.

The piece identification results exhibit a similar trend as in the previous experiments on snippet retrieval. The performance of the baseline model *BL* also declines abruptly as more real scenarios are evaluated. The mean reciprocal rank drops around 59% and 34% points when traversing from (I) to the most realistic case (III), for the retrieval directions A2S and S2A, correspondingly. The worst case happens at (III) for the A2S direction, when only approximately 11% of the scores (22 items among 198) are correctly retrieved as the best match.

We derive the following discussions and observations concerning the performance of our proposed methods:

- In configurations with real data, our methods outperformed the baseline *BL* in all evaluation metrics by a significant margin, with *BL+A+S* being the best model among them. For example, in the fully real scenario (III) the MRR of *BL+A+S* in the A2S direction increased from 0.256 to 0.535, which indicates a performance jump of more than 100% of the former value; in the S2A direction, now only 6% of the recordings are not correctly retrieved among the best ten matches.
- The compound effect of pre-training both encoders (*BL+A+S*) when comparing to individual encoders (*BL+A* and *BL+S*) was stronger than in the two-way snippet retrieval. In the (III)–(*BL+A+S*)–(A2S) configuration the MRR improvement accounted for more than the sum of the individual improvements observed for models *BL+A* and *BL+S*.
- In addition to the dataset-wise improvements, the performance gaps between synthesized and real datasets, and between A2S and S2A directions, were significantly reduced.
- Overall the boost in retrieval quality that our proposed models produced is significantly higher for cross-modal piece identification than for snippet retrieval. This indicates that a moderate performance boost in short fragment-level music retrieval tasks has great potential to escalate to greater improvements in higher-level retrieval problems if a proper post-processing method aggregating those fragments is employed.

To get a better understanding of the matching quality of our models on piece identification scenarios, we discuss on the *separation indicator*, introduced in [42]. This factor measures how distinct the relevant document is among the other items during the retrieval process. Given the vote-based ranked list created during the identification procedure of query Q, its counterpart document is retrieved at rank r. Defining δ_{D_i} as the number of votes the document ranked at i–th position received, the separation indicator $\rho \in \mathbb{R}_+$ is defined as:

$$\rho = \begin{cases} \delta_{D_2}/\delta_{D_1} & \text{if rank } r = 1, \\ \delta_{D_1}/\delta_{D_r} & \text{otherwise.} \end{cases} \tag{3}$$

In this metric, indicators below 1 point out to a correct match, with lower values indicating better matching quality. A $\rho > 1$ implies a wrong detection; the bigger its value, the lesser is number of votes received by the correct document in comparison with the top match.

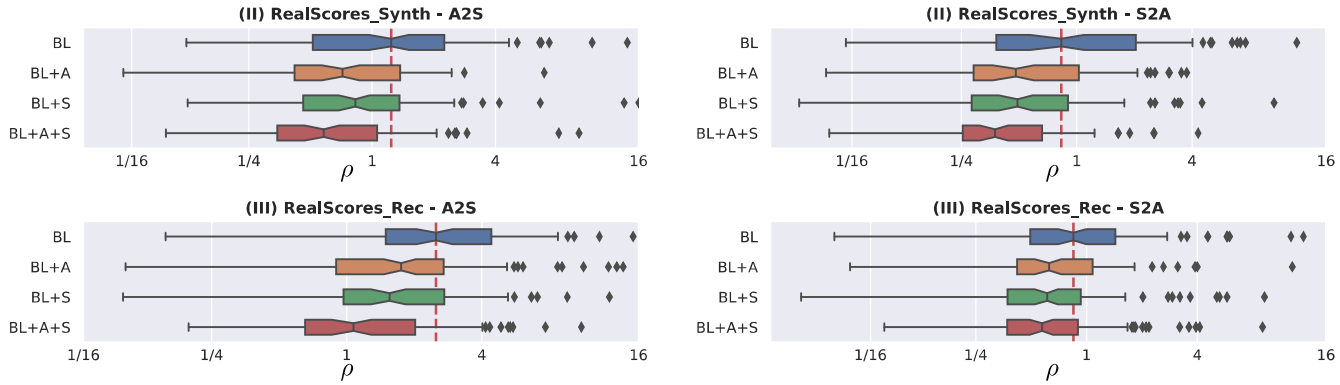

Figure 6: Distributions of the separation indicators produced for cross-modal piece identification on datasets (II) and (III), reducing the number of pieces to 100 per set. The horizontal axes are displayed in logarithmic scale.

Figure 6 visualizes the distribution of the separation indicators obtained when performing cross-modal piece identification on the datasets with real music data. In this experiment we reduce the number of documents of each dataset to 100 pairs of audio recordings and scanned scores.

A joint analysis with Table 1 reveals that overall the models with better piece identification results also exhibit a better matching quality statistics. Noteworthy is the poor matching quality of the (III)-A2S setup, the most realistic case in the audio–score search direction: the distributions of all models are strongly concentrated above $\rho = 1$. Our proposed methods generated overall smaller separation indicators for all audio–sheet music identification setups, indicating that self-supervised learning is a promising orientation for reliable audio–score retrieval systems.

6 CONCLUSION

In this work we designed a learning framework to alleviate labeled data scarcity in training networks to solve audio–score retrieval tasks. We proposed multi-modal self-supervised contrastive learning of short excerpts of sheet music images and audio recordings as a first pre-processing step. In this framework, the network responsible for encoding each modality can be independently pre-trained and enabled for fine-tuning, having the potential to adapt to different tasks that require different fine-tuning configurations. For that we define a pipeline of augmentation transforms specifically for audio and sheet music snippets, and employ publicly available music data to pre-train our networks.

Experiments on two-way snippet retrieval and subsequently on cross-modal piece identification evaluating diverse datasets showed that our proposed framework outperforms current state-of-the arts methods, specially in scenarios composed partially or entirely of real music data. Moreover, the self-supervised approach helped reducing the performance gap between synthetic and real data, which is one of the main challenges of audio–score retrieval problems.

Given the improved retrieval performance in realist configurations, in addition to larges amounts of publicly available music data that are available with easy access, we believe this is a promising research direction for the design of robust multi-modal music search and retrieval systems.

ACKNOWLEDGMENTS

This work is supported by the European Research Council (ERC) under the EU's Horizon 2020 research and innovation programme, grant agreement No. 101019375 ("Whither Music?"), and the Federal State of Upper Austria (LIT AI Lab).

REFERENCES

[1] Abien Fred Agarap. 2018. Deep Learning using Rectified Linear Units (ReLU). https://doi.org/10.48550/ARXIV.1803.08375
[2] Ruchit Agrawal, Daniel Wolff, and Simon Dixon. 2022. A Convolutional-Attentional Neural Framework for Structure-Aware Performance-Score Synchronization. *IEEE Signal Processing Letters* 29 (2022), 344–348.
[3] Andreas Arzt, Sebastian Böck, Sebastian Flossmann, Harald Frostel, Martin Gasser, Cynthia C.S. Liem, and Gerhard Widmer. 2014. The Piano Music Companion. In *Proceedings of the Conference on Prestigious Applications of Intelligent Systems (PAIS)*. Prague, Czechia.
[4] Andreas Arzt, Sebastian Böck, and Gerhard Widmer. 2012. Fast Identification of Piece and Score Position via Symbolic Fingerprinting. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. Porto, Portugal, 433–438.
[5] Andreas Arzt, Gerhard Widmer, and Simon Dixon. 2008. Automatic Page Turning for Musicians via Real-Time Machine Listening. In *In Proceedings of the 18th European Conference on Artificial Intelligence (ECAI)*. Patras, Greece, 241–245.
[6] Stefan Balke, Vlora Arifi-Müller, Lukas Lamprecht, and Meinard Müller. 2016. Retrieving Audio Recordings Using Musical Themes. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Shanghai, China, 281–285.
[7] Stefan Balke, Matthias Dorfer, Luis Carvalho, Andreas Arzt, and Gerhard Widmer. 2019. Learning Soft-Attention Models for Tempo-invariant Audio-Sheet Music Retrieval. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. Delft, Netherlands, 216–222.
[8] Sebastian Böck and Markus Schedl. 2012. Polyphonic piano note transcription with recurrent neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 121–124.
[9] Jorge Calvo-Zaragoza, Jan Hajič Jr., and Alexander Pacha. 2021. Understanding Optical Music Recognition. *Comput. Surveys* 53, 4 (2021).
[10] Carlos Eduardo Cancino-Chacón, Thassilo Gadermaier, Gerhard Widmer, and Maarten Grachten. 2017. An evaluation of linear and non-linear models of expressive dynamics in classical piano and symphonic music. *Machine Learning* 106, 6 (2017), 887–909.
[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*.
[12] Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Vol. 1. 539–546.
[13] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. 2016. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). In *International Conferen1ce on Learning Representations, (ICLR)*.

[14] Matthias Dorfer, Andreas Arzt, and Gerhard Widmer. 2017. Learning Audio-Sheet Music Correspondences for Score Identification and Offline Alignment. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. Suzhou, China, 115–122.

[15] Matthias Dorfer, Jan Hajič jr., Andreas Arzt, Harald Frostel, and Gerhard Widmer. 2018. Learning Audio–Sheet Music Correspondences for Cross-Modal Retrieval and Piece Identification. *Transactions of the International Society for Music Information Retrieval* 1, 1 (2018).

[16] Matthias Dorfer, Jan Schlüter, Andreu Vall, Filip Korzeniowski, and Gerhard Widmer. 2018. End-to-end cross-modality retrieval with CCA projections and pairwise ranking loss. *International Journal of Multimedia Information Retrieval* 7, 2 (01 6 2018), 117–128.

[17] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. 2014. Discriminative Unsupervised Feature Learning with Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, Vol. 27.

[18] Christian Fremerey, Michael Clausen, Sebastian Ewert, and Meinard Müller. 2009. Sheet Music-Audio Identification. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*. Kobe, Japan, 645–650.

[19] Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel, and Douglas Eck. 2019. Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset. In *International Conference on Learning Representations*.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *IEEE International Conference on Computer Vision (ICCV)*. 1026–1034.

[21] Florian Henkel and Gerhard Widmer. 2021. Real-Time Music Following in Score Sheet Images via Multi-Resolution Prediction. *Frontiers in Computer Science* 3 (2021).

[22] Sergey Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning (ICML)*. Lille, France, 448–456.

[23] Özgür Izmirli and Gyanendra Sharma. 2012. Bridging Printed Music and Audio Through Alignment Using a Mid-level Score Representation. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. Porto, Portugal, 61–66. http://ismir2012.ismir.net/event/papers/061-ismir-2012.pdf

[24] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*.

[25] Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. 2014. Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models. *arXiv preprint (arXiv:1411.2539)* (2014).

[26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*. 1097–1105.

[27] Juan C. López-Gutiérrez, Jose J. Valero-Mas, Francisco J. Castellanos, and Jorge Calvo-Zaragoza. 2021. Data Augmentation for End-to-End Optical Music Recognition. In *Proceedings of the 14th IAPR International Workshop on Graphics Recognition (GREC)*. Springer, 59–73.

[28] Meinard Müller, Andreas Arzt, Stefan Balke, Matthias Dorfer, and Gerhard Widmer. 2019. Cross-Modal Music Retrieval and Applications: An Overview of Key Methodologies. *IEEE Signal Processing Magazine* 36, 1 (2019), 52–62.

[29] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*. 2613–2617.

[30] Ken Perlin. 2002. Improving Noise. *ACM Transactions on Graphics (TOG)* 21, 3 (2002), 681–682.

[31] Justin Salamon and Juan Pablo Bello. 2017. Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification. *IEEE Signal Processing Letters* 24, 3 (2017), 279–283.

[32] Jan Schlüter and Thomas Grill. 2015. Exploring Data Augmentation for Improved Singing Voice Detection with Neural Networks. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. 121–126.

[33] Connor Shorten and Taghi M. Khoshgoftaar. 2019. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data* 6, 60 (2019). Issue 1.

[34] Siddharth Sigtia, Emmanouil Benetos, and Simon Dixon. 2016. An End-to-End Neural Network for Polyphonic Piano Music Transcription. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24, 5 (2016), 927–939.

[35] Patrice Simard, Dave Steinkraus, and John Platt. 2003. Best Practices for Convolutional Neural Networks. *International Conference on Document Analysis and Recognition (ICDAR)* 3 (2003), 958–962.

[36] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations (ICLR)*.

[37] Kihyuk Sohn. 2016. Improved Deep Metric Learning with Multi-class N-pair Loss Objective. In *Advances in Neural Information Processing Systems*. 1857–1865.

[38] Naoya Takahashi, Michael Gygli, Beat Pfister, and Luc Van Gool. 2016. Deep Convolutional Neural Networks and Data Augmentation for Acoustic Event Recognition. In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*. San Francisco, USA, 2982–2986.

[39] Timothy J. Tsai. 2020. Towards Linking the Lakh and IMSLP Datasets. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. 546–550.

[40] Eelco van der Wel and Karen Ullrich. 2017. Optical Music Recognition with Convolutional Sequence-to-Sequence Models. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. Suzhou, China, 731–737.

[41] Daniel Yang, Thitaree Tanprasert, Teerapat Jenrungrot, Mengyi Shan, and Timothy J. Tsai. 2019. MIDI Passage Retrieval Using Cell Phone Pictures of Sheet Music. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*. 916–923.

[42] Frank Zalkow, Stefan Balke, and Meinard Müller. 2019. Evaluating Salience Representations for Cross-Modal Retrieval of Western Classical Music Recordings. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. 331–335.