

RESEARCH

Learning Audio–Sheet Music Correspondences for Cross-Modal Retrieval and Piece Identification

Matthias Dorfer*, Jan Hajič Jr.†, Andreas Arzt*, Harald Frostel* and Gerhard Widmer*‡

This work addresses the problem of matching musical audio directly to sheet music, without any higher-level abstract representation. We propose a method that learns joint embedding spaces for short excerpts of audio and their respective counterparts in sheet music images, using multimodal convolutional neural networks. Given the learned representations, we show how to utilize them for two sheet-music-related tasks: (1) piece/score identification from audio queries and (2) retrieving relevant performances given a score as a search query. All retrieval models are trained and evaluated on a new, large scale multimodal audio–sheet music dataset which is made publicly available along with this article. The dataset comprises 479 precisely annotated solo piano pieces by 53 composers, for a total of 1,129 pages of music and about 15 hours of aligned audio, which was synthesized from these scores. Going beyond this synthetic training data, we carry out first retrieval experiments using scans of real sheet music of high complexity (e.g., nearly the complete solo piano works by Frederic Chopin) and commercial recordings by famous concert pianists. Our results suggest that the proposed method, in combination with the large-scale dataset, yields retrieval models that successfully generalize to data way beyond the synthetic training data used for model building.

Keywords: Multimodal embedding space learning; audio-sheet music retrieval

1 Introduction

Many important applications in Music Information Retrieval (MIR) – from retrieval scenarios to live score following to score-informed transcription – require an alignment between different representations of a piece, most often between printed score (sheet music) and recorded performance (audio). Consequently, there has been a lot of work on score-to-performance matching, with different approaches. Traditionally, automatic methods for linking audio and sheet music have relied on some common mid-level representation that allows for comparison and matching (e.g., by computation of distances or similarities) of time points in the audio and positions in the sheet music. Examples of mid-level representations are *symbolic event descriptions*, which involve the error-prone steps of automatic music transcription on the audio side (Böck and Schedl, 2012; Kelz et al., 2016; Sigtia et al., 2016; Cheng et al., 2016) and Optical Music Recognition (OMR) on the sheet music side (Wen et al., 2015; Hajič Jr and Pecina, 2017; Byrd and

Simonsen, 2015; Rebelo et al., 2012); or *spectral features* like pitch class profiles (chroma features), which avoid the explicit audio transcription step but still depend on variants of OMR on the sheet music side. For examples of the latter approach see, e.g., (Balke et al., 2016, 2015; Grachten et al., 2013; Kurth et al., 2007; Fremerey et al., 2009; Izmirli and Sharma, 2012).

To avoid these complications altogether, Dorfer et al. (2016) have proposed the idea of directly matching sheet music images and audio, with deep neural networks. Given short excerpts of audio and the corresponding sheet music, the network learned to predict which location in the given sheet image best matches the current audio excerpt. The potential of this idea was demonstrated in the context of score following.

The approach presented by Dorfer et al. (2017a) and further extended in the present article goes beyond that of Dorfer et al. (2016) in several respects. Most importantly, the original network required both sheet music and audio as input at the same time, in order to then decide which location in the sheet image best matches the current audio excerpt. We now address a more general scenario where both input modalities are required only at training time, for learning the relation between score and audio. This requires a different network architecture that can learn two separate projections, one for embedding the sheet music and one for embedding the audio, which can then be used independently of each other. For example,

* Johannes Kepler University Linz, Altenberger Str., A-4040 Linz, AT

† Charles University, Faculty of Mathematics and Physics, Malostranské nám. 25, Prague, CZ

‡ Austrian Research Institute for Artificial Intelligence, A-1010 Vienna, AT

Corresponding author: Matthias Dorfer (matthias.dorfer@jku.at)

we can first embed a reference collection of sheet music images using the image embedding part of the network, then embed a query audio and search for its nearest sheet music neighbours in the joint embedding space. This general scenario is referred to as *cross-modality retrieval* and supports different applications (two of which will be demonstrated in this paper).

Specifically, we use multimodal convolutional neural networks to learn correspondences directly between images of sheet music and their respective audio counterparts. Given short excerpts of audio and corresponding sheet music images (such as the ones shown in **Figure 1**), the networks are trained to learn an embedding space in which both modalities are represented as fixed-dimensional vectors which can then be compared, e.g., via their cosine distance. To obtain a latent representation that supports this comparison, the networks employ an optimization target that encourages joint embedding spaces where semantically similar items of both modalities live close to each other.

The central idea of this approach is to circumvent the problematic definition of mid-level features by replacing it (on both sides) with a learned transformation of audio and sheet music data to a common vector space. Dorfer et al. (2017a) demonstrated how to utilize this methodology for two sheet music-related real-world applications: (1) *piece identification* via cross-modality retrieval from audio queries, and (2) *audio-to-sheet music alignment* using Dynamic Time Warping (DTW) in the learned joint embedding space.

In the present work we continue the work of Dorfer et al. (2017a) and extend it with the following new contributions, which we hope will greatly facilitate and accelerate future music alignment and retrieval research in the MIR community.

Contribution 1: A New, Large, Open Multimodal Dataset. First experiments by Dorfer et al. (2017a) already indicated that the general approach seems to scale very well with the amount of training data available, and that it is important to have as diverse a dataset as possible to arrive at a robust model. (To that end they also applied various data augmentation strategies.) We will provide additional empirical evidence supporting this in Section 4 of the present paper. Motivated by this, we propose and publish MSMD (*Multimodal Sheet Music Dataset*), a new, free, large-scale, multimodal audio–sheet music dataset, with complete and detailed alignment ground-truth at the level of individual notes. The dataset is built on top of the Mutochia Project,¹ a collection of more than 2000

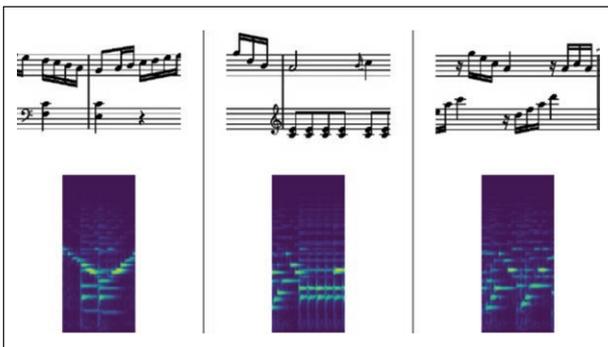


Figure 1: Audio-sheet music pairs presented to the network for embedding space learning.

scores, collected under Creative Commons licenses, which allows us to share and distribute the whole dataset to the research community.

Contribution 2: Experimental Setup, Software Tools, New Experimental Baseline. To allow for an objective benchmarking of further methodological improvements we suggest a specific experimental setup on how to perform evaluations with the dataset. Additionally, to lower the initial hurdles for working with the data, we release a complete set of tools for automatically preparing, viewing, and loading the data. We present extensive experiments, using the new dataset and the suggested experimental setup. The entire experimental code including our pre-trained retrieval models is available online.² We hope this will serve as a basis for further research in the community.

Contribution 3: First Experiments with Substantial Real-world Data. In our previous work (Dorfer et al., 2017a), all experiments were carried out on synthetic data, e.g., rendered sheet music and audio synthesized from MIDI. In this paper, we will report on first large-scale retrieval experiments using scanned images of real sheet music of high complexity (e.g., nearly the complete solo piano works by Frederic Chopin, from the Henle Urtext Edition) and real audio recordings by professional concert pianists. Our results suggest that the proposed method, in combination with the large-scale MSMD dataset (and appropriate data augmentation methods), yields retrieval models that successfully generalize to data way beyond the synthetic training data used for model building. This holds for sheet images and to quite some degree also for real performances.

The remainder of this article is structured as follows: In Section 2 we introduce the new multimodal audio–sheet music dataset. (A detailed description of how the data was produced is given in Appendix A). Section 3 describes how to train the proposed retrieval model on this dataset, along with the applied data augmentation strategies. Sections 4 and 5 present extensive experimental results on two different retrieval tasks (sheet/audio snippet retrieval and piece/performance identification). Section 6 explores how the method generalizes to complex real-world data (e.g. scanned sheet music and real performances). In Section 7, we summarize our results and propose an agenda for further research, which is made possible by the availability of this new dataset.

2 A Multimodal Sheet Music Dataset

In this section, we introduce the Multimodal Sheet Music Dataset (MSMD) used in our experiments. The dataset is based on the Mutochia collection of LilyPond-encoded³ pieces and is created entirely automatically from the Mutochia project repository.⁴ Some examples illustrating the variety of music in MSMD are shown in **Figure 2**.

MSMD contains 479 solo piano pieces of mostly classical music by 53 composers, for a total of 1,129 pages of music. The pieces are available in two modalities: as scores (sheet music), and as MIDI, both exported directly from LilyPond. We extract staff and notehead locations and pitches from the score, and synthesize audio from the MIDI file and compute spectrograms (see Section 2.1 below). Then, we align three modalities – noteheads in

score; MIDI events; audio/spectrogram timeline — using temporal and pitch information that is provided by the Lilypond-MIDI connection (see **Figure 3**).

What makes the dataset valuable, and makes the experiments described in this paper possible, is that the modalities are automatically aligned at a fine-grained level: each individual notehead in the scores is linked to its counterpart MIDI event(s) in the audio modality. Our models then learn from snippet pairs centered around the aligned notehead/note event pairs (see **Figure 1**). There is a total of 344,742 such aligned pairs. (This is of course not

the only setting for which the fine-grained alignment can be used, as discussed in Section 7.)

A full description of the dataset and the toolchain used to build it is given in Appendix A.

2.1 Spectrogram Computation

We generate up to 7 performances per piece, using various tempo ratios between 0.9 and 1.1 of the original MIDI tempo and four open-source piano soundfonts (this is relevant for Section 3.2 on data augmentation). One of the four soundfonts is reserved for testing (it is never used for spectrograms seen in training).

We compute log-frequency spectrograms of the audio files, with a sample rate of 22.05 kHz and an FFT window size of 2048 samples. For dimensionality reduction we apply a normalized logarithmic filterbank with 16 bands per octave, allowing only frequencies from 30 Hz to 6 kHz. This results in 92 frequency bins. The frame rate of the spectrogram is 20 frames per second.

2.2 Recommended Train/Test Splits

We consider three scenarios that motivate how MSMD should be split into a training/validation set and a test set. First, we consider simply a random mix of all the available pieces, denoted in the experimental results as *all*.

Next, for experiments that are to focus on a stylistically homogeneous body of music, we suggest using only the works of a single composer. In the case of MSMD, this would be Johann Sebastian Bach, since there are enough of his works in MSMD to allow training on this set, and their style is consistent. Experiments with this train/test split are labeled *bach-only*.

Finally, for specialized experiments targeted at the generalization to a previously unseen musical style, we propose to leave one composer (again, J. S. Bach) out of the training/validation data and use his pieces only for testing. Experiments with this train/test split are labeled *bach-out*.

The exact piece lists defining the splits, including the split of the training data into train and validation sets, are included with the dataset. The statistics for the splits are given in **Table 1**.

Figure 2: Example scores illustrating the range of music in MSMD, from simple to complex.

Table 1: MSMD statistics for the recommended train/test splits. Note that the numbers of noteheads, events, and aligned pairs do not match. This is because (a) not every notehead is supposed to be played, esp. tied notes; (b) some onsets do not get a notehead of their own, e.g. ornaments; (c) sometimes the alignment algorithm makes mistakes.

Split Name	# Pieces/Aln. Pairs	Part	# Pieces	# Pages	# Noteheads	# Events	# Aln. Pairs
all	479 / 344,742	train	360	970	316,038	310,377	308,761
		valid	19	28	6,907	6,583	6,660
		test	100	131	29,851	29,811	29,321
bach-only	173 / 108,316	train	100	251	77,834	75,283	74,769
		valid	23	40	10,805	10,379	10,428
		test	50	88	23,733	23,296	23,119
bach-out	479 / 344,742	train	281	725	235,590	233,041	231,617
		valid	25	25	4,834	4,772	4,809
		test	173	379	112,372	108,958	108,316

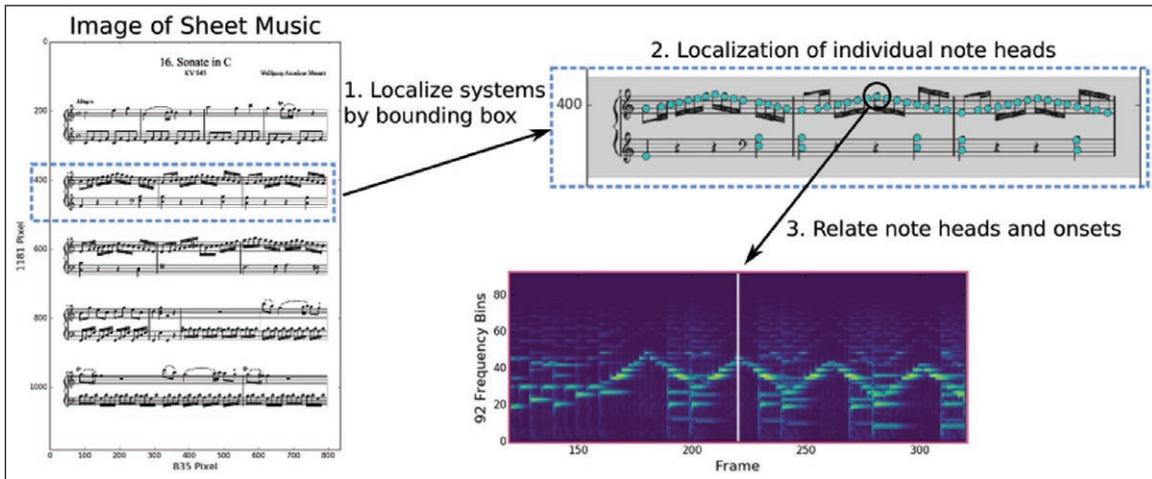


Figure 3: Core dataset workflow. For producing the alignment, it is necessary to “unroll” the score using individual staff systems, so that the ordering of noteheads in the score corresponds to the ordering of the notes in the MIDI file.

Besides the synthetic MSMD dataset, we will also use (in Section 6) a collection of scanned scores and recorded performances, to evaluate how our models generalize to real-world scenarios.

3 Learning Audio–Sheet Music Correspondences

Our approach is built around a neural network designed for learning the relationship between two different data modalities. The network learns its behavior solely from the examples presented at training time. We start this section by explaining how to prepare and post-process the MSMD dataset proposed in Section 2 in order to generate exactly these training examples. The final part of this section describes the underlying learning methodology in detail.

3.1 Data Preparation

The MSMD dataset already contains segmentations of the staff systems in all of its sheet music images (cf. **Figure 3**). In particular, we are given annotated bounding boxes around the individual systems along with the positions of the note heads associated with these systems. In addition to the scores we are also provided with audio renditions synthesized from MIDI files, or spectrograms computed from these. And most importantly, we get for each annotated note head in the image a pointer referring to its corresponding onset time in the audio. This means that we know for each notehead its location (in pixel coordinates) in the image, and its onset time in the audio. Based on this relationship and annotations, we cut out corresponding snippets of sheet music images (in our case 160×200 pixels) and short excerpts of audio represented by log-frequency spectrograms (92 bins \times 42 frames, ≈ 2 sec of music). **Figure 1** shows three examples of such audio – sheet music correspondences; these are the pairs presented to our multimodal networks for training.

3.2 Data Augmentation

To improve the generalization ability of the resulting networks, we propose several data augmentation strategies specialized to score images and audio. In machine learning, *data augmentation* refers to the application of (realistic)

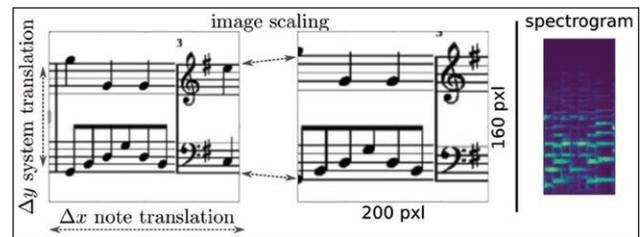


Figure 4: Overview of image augmentation strategies. The size of the sliding image window remains constant (160×200 pixels) but its content changes depending on the augmentations applied. The spectrogram remains the same for the augmented image versions.

data transformations in order to synthetically increase the effective size of the training set (Ronneberger et al., 2015; McFee et al., 2015). We already emphasize at this point that data augmentation is a crucial component for learning cross-modality representations that generalize to unseen music.

For **sheet image augmentation** we apply three different transformations, summarized in **Figure 4**. The first is *image scaling* where we resize the image between 95 and 105% of its original size. This should make the model robust to changes in the overall dimension of the scores. Secondly, in Δy system translation we slightly shift the system in the vertical direction by $\Delta y \in [-5, 5]$ pixels. We do this as the system detector will not detect each system in exactly the same way and we want our model to be invariant to such translations. In particular, it should not be the absolute location of a note head in the image that determines its meaning (pitch) but its relative position with respect to the staff. Finally, we apply Δx note translation, meaning that we slightly shift the corresponding sheet image window by $\Delta x \in [-5, 5]$ pixels in the horizontal direction. In our experiments, all sheet augmentation strategies are applied simultaneously as the individual effects were already investigated in (Dorfer et al., 2017a).

In terms of **audio augmentation**, we render the training pieces with three different sound fonts and additionally vary the tempo between 95 and 110 % of

its original tempo when preparing the MSMD dataset performances (see Section 2). For the test set, we only use performances rendered at the original preset tempo but using an *additional unseen soundfont* (no performances using this soundfont are used for training). The test set is kept fixed to reveal the impact of the different data augmentation strategies.

3.3 Embedding Space Learning

This subsection describes the underlying learning methodology. As mentioned above, the core of our retrieval approach is a neural network capable of learning cross-modal correspondences between short snippets of audio and sheet music images. In particular, we aim to learn a joint embedding space of the two modalities in which to perform nearest-neighbour search. One method for learning such a space, which has already proven to be effective in other domains such as text-to-image retrieval, is based on the optimization of a pairwise ranking loss (Kiros et al., 2014; Socher et al., 2014). Before explaining this optimization target, we first introduce the general architecture of our correspondence learning network. As shown in **Figure 5** the network consists of two separate pathways f and g taking two inputs at the same time. Input one is a sheet image snippet \mathbf{I} and input two is an audio excerpt \mathbf{A} . This means in particular that network f is responsible for processing the image part of an input pair and network g is responsible for processing the audio. The output of both networks (represented by the *Embedding Layer* in **Figure 5**) is a k -dimensional vector representation encoding the respective inputs. In our case the dimensionality of this representation is $k = 32$. We denote these hidden representations by $\mathbf{x} = f(\mathbf{I}, \Theta_f)$ for the sheet image and $\mathbf{y} = g(\mathbf{A}, \Theta_g)$ for the audio spectrogram, respectively, where Θ_f and Θ_g are the parameters of the two networks.

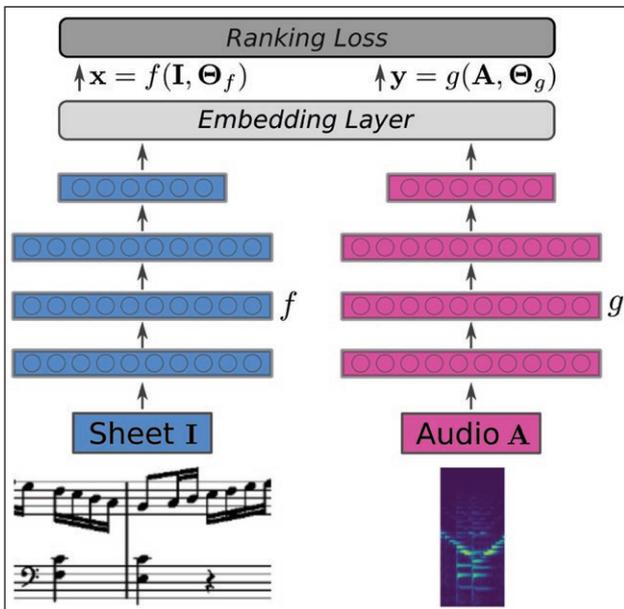


Figure 5: Architecture of correspondence learning network. The network is trained to optimize the similarity (in embedding space) between corresponding audio and sheet image snippets by minimizing a pair-wise ranking loss.

Given this network design, we now explain the pairwise ranking objective. Following Kiros et al. (2014) we first introduce a *scoring function* $s(\mathbf{x}, \mathbf{y})$ as the cosine similarity $\mathbf{x} \cdot \mathbf{y}$ between the two hidden representations (\mathbf{x} and \mathbf{y} are scaled to have unit norm). Based on this scoring function we optimize the following pairwise ranking objective ('hinge loss' (Rosasco et al., 2004)):

$$\mathcal{L}_{rank} = \sum_{\mathbf{x}} \sum_k \max\{0, \alpha - s(\mathbf{x}, \mathbf{y}) + s(\mathbf{x}, \mathbf{y}_k)\} \quad (1)$$

In our application \mathbf{x} is an embedded sample of a sheet image snippet, \mathbf{y} is the embedding of the matching audio excerpt and \mathbf{y}_k are the embeddings of the *contrastive* (mismatching) audio excerpts (in practice all remaining samples of the current training batch). When training our models we fix the mini-batch size to 100 samples. This means in particular, that for each \mathbf{x} we are given one positive matching sample \mathbf{y} and 99 contrastive samples \mathbf{y}_k . Mini-batches are drawn randomly from the entire training set without any sophisticated sampling strategy such as hard negative mining (Henriques et al., 2013). The hyperparameter α defines the margin of the loss function and is set to 0.7 for all our experiments. The intuition behind this loss function is to encourage an embedding space where the distance between matching samples is lower than the distance between mismatching samples. If this condition is roughly satisfied, we can then perform cross-modal retrieval by simple nearest neighbour search in the embedding space. This will be explained in detail in Section 4.

The network itself is implemented as a VGG-style convolution network (Simonyan and Zisserman, 2015) consisting of 3×3 convolutions followed by 2×2 max-pooling as outlined in detail in **Table 2**. The final convolution layer computes 32 feature maps and is subsequently processed with a global average pooling

Table 2: Audio – sheet music model. BN: Batch Normalization (Ioffe and Szegedy, 2015), ELU: Exponential Linear Unit (Clevert et al., 2015), MP: Max Pooling, Conv (3, pad-1)-16: 3×3 convolution, 16 feature maps and padding 1.

Sheet-Image 80×100	Audio (Spectrogram) 92×42
2 × Conv(3, pad-1)-24	2 × Conv(3, pad-1)-24
BN-ELU + MP(2)	BN-ELU + MP(2)
2 × Conv(3, pad-1)-48	2 × Conv(3, pad-1)-48
BN-ELU + MP(2)	BN-ELU + MP(2)
2 × Conv(3, pad-1)-96	2 × Conv(3, pad-1)-96
BN-ELU + MP(2)	BN-ELU + MP(2)
2 × Conv(3, pad-1)-96	2 × Conv(3, pad-1)-96
BN-ELU + MP(2)	BN-ELU + MP(2)
Conv(1, pad-0)-32-BN-LINEAR	Conv(1, pad-0)-32-BN-LINEAR
GlobalAveragePooling	GlobalAveragePooling
Embedding Layer + Ranking Loss	

layer (Lin et al., 2014) that produces a 32-dimensional vector for each input image and spectrogram, respectively. This is exactly the dimension of our retrieval embedding space. At the top of the network we put a canonically correlated embedding layer (Dorfer et al., 2018) combined with the ranking loss described above. The structure of the model is analogous to the one presented in (Dorfer et al., 2017a) with the single difference that the sheet-image snippet is downsized by factor two ($160 \times 200 \rightarrow 80 \times 100$) before being presented to the network. This downsized image still contains all musically relevant content but reduces the number of computations required in the sheet image stack of the network. The saved computation time is then invested in doubling the number of feature maps to increase the capacity of our models. In terms of optimization we use the *Adam* update rule (Kingma and Ba, 2015) with an initial learning rate of 0.002. We watch the performance of the network on the validation set and halve the learning rate if there is no improvement for 30 epochs. This procedure is repeated five times to finetune the model.

4 Evaluation 1: Two-Way Snippet Retrieval

In this section, we evaluate the ability of our model to retrieve the correct counterpart when given an instance of the other modality as a search query. This first set of experiments is carried out on the lowest possible granularity, namely, on sheet image snippets and spectrogram excerpts such as shown in **Figure 1**.

For easier explanation we describe the retrieval procedure from an *audio query point of view* but stress that the opposite direction works in exactly the same fashion. Given a spectrogram excerpt \mathbf{A} as a search query we want to retrieve the corresponding sheet image snippet \mathbf{I} . For retrieval preparation we first embed all candidate image snippets \mathbf{I}_j by computing $\mathbf{x}_j = f(\mathbf{I}_j)$ as the output of the image network. The candidate snippets originate from all unseen pieces from the respective test set. In a second step we embed the given query audio as $\mathbf{y} = g(\mathbf{A})$ using the audio pathway g of the network. Finally, we select the audio’s nearest neighbour \mathbf{x}^* from the set of embedded image snippets as

$$\mathbf{x}^* = \arg \min_{\mathbf{x}_i} \left(1.0 - \frac{\mathbf{x}_i \cdot \mathbf{y}}{\|\mathbf{x}_i\| \|\mathbf{y}\|} \right) \quad (2)$$

based on their pairwise cosine distance. **Figure 6** shows a sketch of this retrieval procedure.

4.1 Experimental Setup

We run retrieval experiments on all three training splits (compare Subsection 2.2) for the different combinations of data augmentation strategies described in Section 3.2. Results are presented for both retrieval direction, audio-to-sheet and sheet-to-audio retrieval. The unseen synthesizer and the tempo for the test set remain fixed for all settings. This allows us to directly investigate the influence of the different augmentation strategies. We further limit the number of retrieval test candidates to 2000 sheet snippets

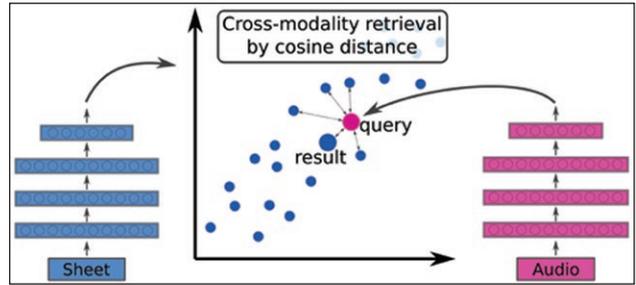


Figure 6: Sketch of sheet music-from-audio retrieval.

The blue dots represent the embedded candidate sheet music snippets. The red dot is the embedding of an audio query. The larger blue dot highlights the closest sheet music snippet candidate selected as retrieval result.

and audio excerpts respectively for all three splits. The 2000 candidates are randomly sampled across all of the test pieces. Having a fixed number of retrieval candidates makes performance of the learned models comparable across the different training splits.

As evaluation measures we compute the *Recall@k* ($R@k$), the *Mean Reciprocal Rank* (*MRR*), as well as the *Median Rank* (*MR*). The $R@k$ rate (high is better) is the percentage of queries which have the correct corresponding counterpart in the first k retrieval results. The *MR* (low is better) is the median position of the target in a cosine-similarity-ordered list of available candidates. Finally, we define the *MRR* (higher is better) as the mean value of $1/\text{rank}$ over all queries where rank is again the position of the target in the similarity ordered list of available candidates.

4.2 Experimental Results

Table 3 summarizes the results on all three training splits for the different data augmentation strategies. Additionally, to get a better intuition of the results we provide the random-retrieval baseline for the 2000 candidates.

The common observation consistent across all datasets, performance measures and retrieval directions is that data augmentation helps to significantly improve the performance of all models. When isolating the effects of the two individual augmentation strategies, we see that audio augmentation yields the largest gain in performance on the test set. Surprisingly, sheet augmentation only helps to improve the performance on the bach-set and even degrades the model on the bach-out set. We do not report results on the validation set, but note that this behavior is reversed on the validation set. The reason for this is that the validation split is synthesized with a soundfont also covered by the training set. This means that in order to get a high performance on the validation set it is not required to generalize to unseen audio (spectrogram) characteristics. This is different for the test set, as it is synthesized with a hold out soundfont, explaining the large performance gain in **Table 3** when applying audio augmentation. Finally, when combining both audio and sheet augmentation we get the best results for all of the models generalizing to unseen scores as well as unseen audio. When recalling that our query length is only 42 spectrogram frames (≈ 2 seconds of audio) per

Table 3: Snippet retrieval results. The table compares the influence of train/test splits and data augmentation on retrieval performance in both directions. For the audio augmentation experiments no sheet augmentation is applied and vice versa. *none* represents 1 sound font, with original tempo, and without sheet augmentation. We limit the number of retrieval candidates to 2000 for each of the splits to make the comparison across the different test sets fair.

Audio-to-Sheet Retrieval												
Aug.	bach-only				bach-out				all			
	R@1	R@25	MRR	MR	R@1	R@25	MRR	MR	R@1	R@25	MRR	MR
none	0.25	0.73	0.37	6	0.31	0.83	0.44	3	0.33	0.76	0.44	4
sheet	0.38	0.81	0.49	3	0.25	0.78	0.37	5	0.33	0.75	0.44	4
audio	0.48	0.87	0.59	2	0.38	0.83	0.50	2	0.46	0.82	0.57	2
full	0.52	0.87	0.62	1	0.46	0.86	0.57	2	0.50	0.83	0.60	2
rand-bl	0.00	0.01	0.0	1000	0.00	0.01	0.00	1000	0.00	0.01	0.00	1000

Sheet-to-Audio Retrieval												
Aug.	bach-only				bach-out				all			
	R@1	R@25	MRR	MR	R@1	R@25	MRR	MR	R@1	R@25	MRR	MR
none	0.34	0.81	0.46	3	0.35	0.83	0.48	3	0.39	0.80	0.51	2
sheet	0.45	0.85	0.57	2	0.28	0.80	0.42	4	0.40	0.79	0.52	2
audio	0.51	0.87	0.62	1	0.39	0.85	0.52	2	0.49	0.84	0.59	2
full	0.56	0.89	0.66	1	0.46	0.87	0.57	2	0.51	0.85	0.61	1
rand-bl	0.00	0.01	0.00	1000	0.00	0.01	0.00	1000	0.00	0.01	0.00	1000

excerpt and that we select from a set of 2000 available candidate snippets, achieving a MR of not more than 2 is an impressive result. In particular, given a short excerpt of audio, the median position of the exactly matching counterpart is either 1 or 2 depending on the data split. This is even more impressive when keeping in mind that music is highly repetitive and that we consider only the exactly matching counterpart as a correct retrieval result. When comparing the two retrieval directions we see that sheet-to-audio retrieval works slightly but consistently better than the opposite direction again across all of the datasets.

In the following sections, we will see that this retrieval performance is sufficient for performing higher level tasks such as piece identification from audio queries. Furthermore, we will show in additional experiments in Section 6 that the resulting *full augmentation models* reach a level of generalization that makes them useful in practical real-world applications operating on scanned sheet images completely out of the synthetic training data domain.

4.3 Influence of Dataset Size

In this additional experiment we investigate the influence of training set size on the final retrieval performance. For this purpose we retrain the same network architecture once with 10, 25, 50 and 75% of the original training examples in the no-augmentation setting of the bach-only split. We chose the no-augmentation setting for this experiment because we want to reveal the impact of the number of available training examples without cluttering the results with the effects of data augmentation.

Figure 7 compares the MRR on the test set for the respective proportions of training observations. The first,

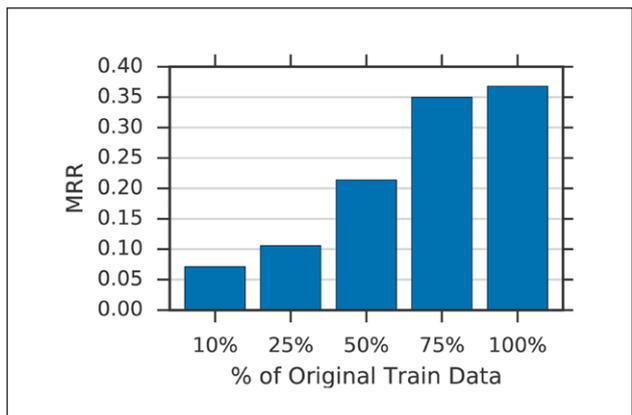


Figure 7: Influence of training set size on test set retrieval performance (MRR) evaluated on the bach-split in the no-augmentation setting.

however not surprising observation, is that the training set size has a severe impact on the final retrieval capabilities of the model. The MRR increases by almost 30 points when using only 10% of the data compared with the full dataset size. The second, more interesting observation, is that the relative improvement in performance is largest around 50% of the training set size (from 25% to 50% and from 50% to 75%). This indicates that there is a critical number of samples required to start generalizing to unseen sheet images. Finally, we observe that the gap between 75% and 100% of the data is fairly small compared to the remaining performance jumps. We interpret this as a positive outcome, suggesting that the full data set is sufficiently large to reach the full performance capabilities of the retrieval model.

5 Evaluation 2: Piece Identification and Performance Retrieval

Given the above model learning to express similarities between sheet music snippets and audio excerpts, we now describe how to use it for solving our targeted tasks: (1) identifying the respective piece of sheet music when given an entire audio recording as a query, and (2) given a score (sheet-image), retrieve a set of corresponding performances. The entire identification pipeline consists of two main stages summarized in **Figure 8**.

Score database preparation. Again, we describe the procedure from an audio query point of view and stress that the opposite direction works analogously. The first step is to prepare a sheet music retrieval database as follows: Given a set of sheet music images along with their annotated systems, we cut each piece of sheet music j into a set of image snippets $\{I_{ji}\}$ analogously to the snippets presented to our network for training. For each snippet, we store its originating piece j . We then embed all candidate image snippets into the retrieval embedding space by passing them through the image part f of the multimodal network. This yields, for each image snippet, a 32-dimensional embedding coordinate vector $\mathbf{x}_{ji} = f(I_{ji})$. The left part of **Figure 8** summarizes database preparation.

Retrieving sheet music at runtime. Once the database is prepared we perform piece retrieval as summarized in the right part of **Figure 8**. Given a whole audio recording as a search query, we aim to identify the corresponding piece of sheet music in our database.

First, we retrieve sheet snippets. As with the sheet image, we start by cutting the audio (spectrogram) into a set of excerpts $\{A_1, \dots, A_k\}$, again exhibiting the same dimensions as the spectrograms used for training, and embed all query spectrogram excerpts A_k with the audio network g . Then we proceed as described in Section 4 and select for each audio its nearest neighbours from the set of all embedded image snippets. In our experiments we consider for each query excerpt its top 25 retrieval results for piece selection.

Second, we combine the retrieved snippets to select the pieces. Since we know for each of the image snippets its originating piece j , we can now have the retrieved image snippets \mathbf{x}_{ji} vote for the piece. The piece achieving the highest count of votes is our final retrieval result. A similar procedure was used for example by Casey et al. (2008) for cover song identification.

5.1 Experimental Setup

We again carry out experiments on the three predefined data splits and compare the impact of data augmentation on the resulting retrieval (identification) performance. In addition to the results presented in Dorfer et al. (2017a) we also present results for the opposite retrieval direction, i.e., retrieving relevant performances given a score image as a search query. It is also important to note that here we are still evaluating on our synthesized data. This will change in Section 6 where we work with scanned sheet music and recordings of real performances. As a retrieval measure, we compute the ranks@k ($Rk@k$) as the number of pieces retrieved within the first k retrieval results. $Rk@1$ means that a piece is ranked at position one and therefore identified correctly. To be consistent and comparable with **Table 3** we also report the respective relative numbers ($R@k$) in brackets. Along with the data-splits we also report the number of candidate pieces (#) contained in the test set.

5.2 Experimental Results

Table 4 summarizes all piece identification results. The first observation is that the results regarding data augmentation are in line with the ones presented in Section 4. Looking at the different splits we see that a large fraction of the respective pieces is retrieved as the top retrieval result. When relaxing the retrieval measure and considering the $Rk@5$ we see that almost all of the pieces are contained in the set of top five results, especially in the direction of retrieving audio with a sheet image query. Although this is not the most sophisticated way of employing our network for piece retrieval, it clearly shows

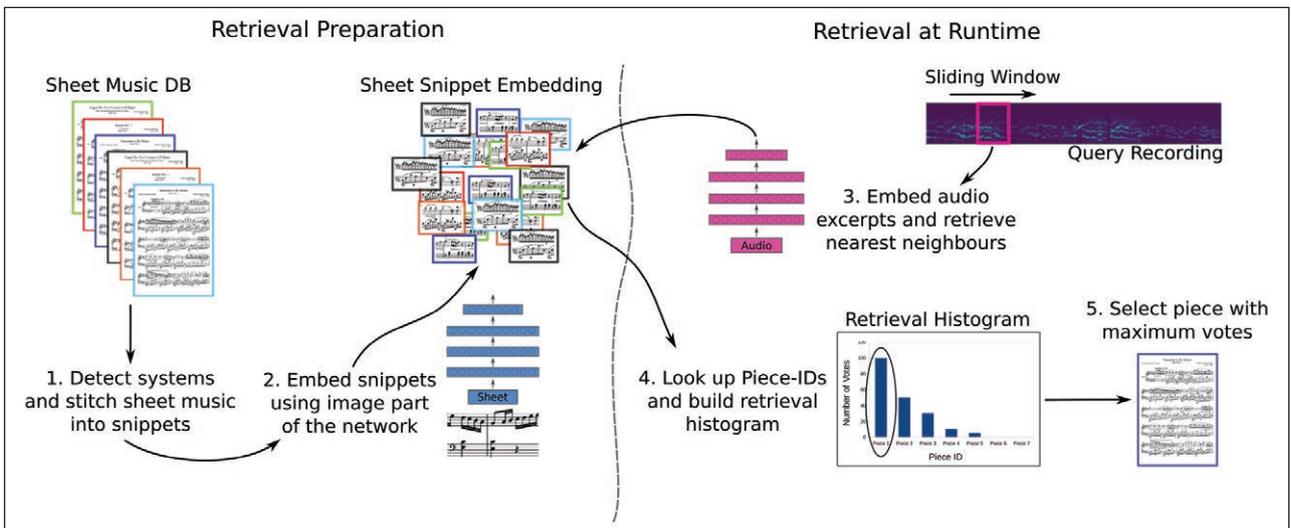
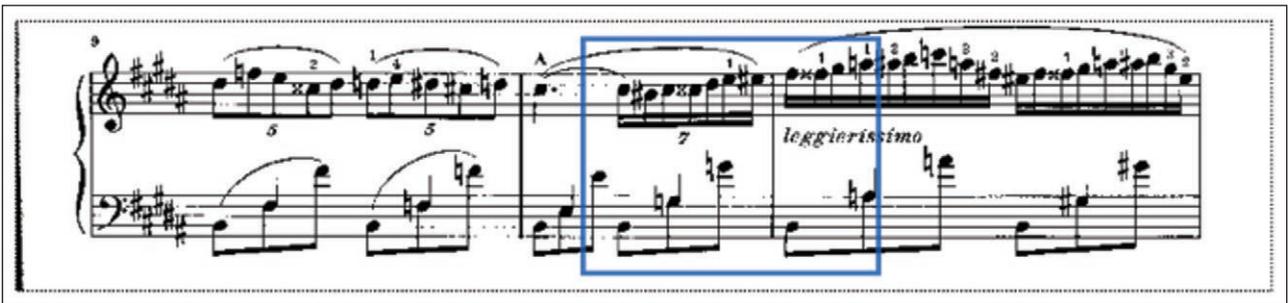


Figure 8: Piece retrieval concept from audio query. The entire pipeline consists of two stages: retrieval preparation and retrieval at runtime (best viewed in color, for details see Section 5).

Table 4: Piece and performance identification results on synthetic data for all three splits.

Train Split	#	Aug.	Synthesized-to-Score				Score-to-Synthesized			
			Rk@1	Rk@5	Rk@10	>Rk10	Rk@1	Rk@5	Rk@10	>Rk10
bach-only	50	none	33 (0.66)	46 (0.92)	48 (0.96)	2 (0.04)	39 (0.78)	48 (0.96)	49 (0.98)	1 (0.02)
		full	41 (0.82)	49 (0.98)	50 (1.00)	0 (0.00)	47 (0.94)	50 (1.00)	50 (1.00)	0 (0.00)
bach-out	173	none	125 (0.72)	158 (0.91)	163 (0.94)	10 (0.06)	145 (0.84)	164 (0.95)	166 (0.96)	7 (0.04)
		full	143 (0.83)	163 (0.94)	167 (0.97)	6 (0.03)	149 (0.86)	169 (0.98)	172 (0.99)	1 (0.01)
all	100	none	67 (0.67)	96 (0.96)	98 (0.98)	2 (0.02)	94 (0.94)	98 (0.98)	99 (0.99)	1 (0.01)
		full	82 (0.82)	97 (0.97)	99 (0.99)	1 (0.01)	92 (0.92)	99 (0.99)	100 (1.00)	0 (0.00)

**Figure 9:** Exemplar staff line automatically extracted from a scanned score version of Chopin's Nocturne Op. 9 No. 3 in B major (Henle Urtext Edition; reproduced with permission). The blue box indicates an example sheet snippet fed to the image part of the retrieval embedding network.

the usefulness of our model and its learned audio and sheet music representations for such tasks. The next steps towards making the identification process more robust will be to exploit the spatial and temporal structure (relation) of subsequent queries, as proposed by Balke et al. (2016).

6 Real-world Data: Retrieving Scanned Sheet Music and Real Performances

So far, both training the models and all of our experiments were carried out on synthetic data (rendered sheet music and synthesized MIDI performances). In this section, we present results on a set of additional experiments to answer the most prominent question: *How well do the models generalize to real data?*

6.1 Experimental Setup

Firstly, we clarify what we consider as real or realistic data in this context. Regarding sheet music, we use scanned images of scores from widely used commercial publishers such as Henle or Universal Edition. **Figure 9** shows an example staff system from a piece by Frederic Chopin (Nocturne Op. 9 No. 3 in B major) to give an impression of this kind of data. For the performances, we use commercial audio recordings by various famous pianists (e.g., Ashkenazy, Pollini, Arrau, Horowitz) that we happened to have in our music collection. We do not need any performance-to-score alignments if they are only used as test cases for piece retrieval. For further variability we have included music by different composers: Mozart (14 pieces; 88 score pages), Beethoven (29 pieces; 181 score pages), and Chopin (150 pieces; 871 score pages).

Retrieval preparation and retrieval itself follows exactly the descriptions outlined in Section 5 above. The sole difference in terms of data preparation is that for the scanned sheet music, we of course do not have the annotated system bounding boxes available. As the overall goal is to have the means to fully automatically index a large collection of scores, we developed an automatic system detection algorithm inspired by (Gallego and Calvo-Zaragoza, 2017; Dorfer et al., 2017b). Given the automatic system detection, we have all the tools to automatically create the database (cf. **Figure 8**). Note that we do not need to detect noteheads – they were only relevant in aligning the modalities for training. For retrieval, we use the embedding networks trained on the *all* split using full data augmentation, as this data is most diverse in terms of sheet music and audio.

6.2 Experimental Results

Table 5 summarizes our results in the real data setting. To isolate the effects of real sheet images and real performance audio we repeat the experiment in two configurations: first with real scores and synthesized audio and second with real scores and real performances. Looking at the first group of experiments (top part of **Table 5**) with scanned sheet music and synthesized audio, we retrieve in the case of Mozart 13 of 14 as the top candidate. The opposite retrieval direction works equally well. For Chopin we retrieve 127 out of 150 scanned scores at position one and 140 if we take the top five results into account. For the remaining sets and measures, we make similar observations. Given that

Table 5: Evaluation on real data: Piece retrieval results on scanned sheet music and recordings of real performances. The model used for retrieval is trained on the all-split with full data augmentation.

Composer	#	Synthesized-to-Real-Score				Real-Score-to-Synthesized			
		Rk@1	Rk@5	Rk@10	>Rk10	Rk@1	Rk@5	Rk@10	>Rk10
Mozart	14	13 (0.93)	14 (1.00)	14 (1.00)	0 (0.00)	13 (0.93)	14 (1.00)	14 (1.00)	0 (0.00)
Beethoven	29	24 (0.83)	27 (0.93)	27 (0.93)	2 (0.07)	25 (0.86)	27 (0.93)	29 (1.00)	0 (0.00)
Chopin	150	127 (0.85)	140 (0.93)	145 (0.97)	5 (0.03)	112 (0.75)	136 (0.91)	142 (0.95)	8 (0.05)

Composer	#	Performance-to-Real-Score				Real-Score-to-Performance			
		Rk@1	Rk@5	Rk@10	>Rk10	Rk@1	Rk@5	Rk@10	>Rk10
Mozart	14	5 (0.36)	14 (1.00)	14 (1.00)	0 (0.00)	12 (0.86)	13 (0.93)	13 (0.93)	1 (0.07)
Beethoven	29	16 (0.55)	25 (0.86)	27 (0.93)	2 (0.07)	20 (0.69)	28 (0.97)	28 (0.97)	1 (0.03)
Chopin	150	36 (0.24)	72 (0.48)	91 (0.61)	59 (0.39)	58 (0.39)	94 (0.63)	111 (0.74)	39 (0.26)

the model was trained on purely rendered sheet music, with different and very consistent typesetting properties and containing no image noise at all, we consider this a remarkable result. We conclude that our model, in combination with the proposed dataset, is able to learn representations that generalize to completely unseen sheet music of a different typesetting style, beyond the synthetic training data.

In a final step, we further increase the level of difficulty of the retrieval setting. Instead of audio synthesized from MIDI, we use commercial recordings of performances by famous concert pianists. The bottom part of **Table 5** lists our results in this configuration. Pieces and sheet music are identical to the experiments above to allow a direct comparison and to reveal the effects of the individual sources of potential problems. The general trend is that all performance measures drop compared to the synthetic audio settings. In terms of Rk@5 of Performance-to-Score retrieval, we are now able to retrieve 72 instead of 140 Chopin pieces, and 91 when considering Rk@10. Although this is a significant drop, it is in our opinion still a good result given the synthetic training data and the difficulty of the task. For the Mozart set, we are able to retrieve 12 out of 14 performances at position one given a scanned score as a query. Interestingly, the score-to-performance retrieval direction works better in this configuration for all three composers. We do not yet have a convincing explanation for this effect.

Based on these results we conclude that focusing on learning more robust audio representations is one of the main research challenges for future work (Section 7 contains a deeper discussion).

7 Discussion and Future Work

In this section, we summarize and discuss our main findings and outline a list of potential applications and research problems that can now be addressed with the proposed MSMD data set.

Our experiments on piece and performance identification on both synthetic and real data (see Sections 5 and 6) lead to the following observations: given the MSMD data set and the proposed methodology, we can learn retrieval models that clearly generalize beyond the synthetic training data domain. This holds especially for scanned images of unseen

sheet music. When dealing with real performances, we still achieve good retrieval results, but encounter a significant drop in all performance measures. We have to remember that our model (a multimodal convolutional neural network) has a fixed and limited field of view on both the audio (excerpt) and the sheet music (snippet). While this is not a problem on the sheet music side, it definitely is for performance audio, which may exhibit rather extreme tempo changes and differences (in addition to challenges such as asynchronous onsets, pedal, room acoustics, or dynamics). Given these facts about performance and our experimental findings, we believe that learning robust audio representations is one of the main open research problems to be addressed.

Regarding the retrieval (piece/performance identification) methodology, note that so far we completely ignore the strong temporal dependencies between subsequent queries, which are inherent in music. An obvious next step will be to extend the identification procedure in a way that exploits these spatio-temporal relationships (e.g., as in (Balke et al., 2016)).

Finally, we see a large number of potential applications of the MSMD dataset introduced in this work. Recall that the dataset comes with a rich set of annotations and alignments. In particular, we know for each note head in each sheet image its pixel position as well as its corresponding MIDI note-event and therefore also its onset time, pitch and duration in the synthesized audios. Consequently, we expect that MSMD will become a valuable resource for future work on topics such as:

- Optical Music Recognition (OMR)
- Off-line Alignment of Sheet Images to Audio
- (Real-time) Score-Following in Sheet Images
- Sheet-Informed Transcription (i.e., to detect errors in a performance while practicing)
- Piece and Performance Retrieval as a Service for Musicians.

We hope that the research community will make use of this dataset, which we believe brings many sheet music related MIR tasks in reach of state-of-the-art machine learning methods.

8 Conclusion

We have presented a methodology for learning correspondences between short snippets of sheet music and their counterparts in the music audio. The learned shared latent representation of the two modalities can be utilized for cross-modality retrieval, i.e., for identifying scores from full audio queries and vice versa. To improve the performance of our method and, more generally, to boost this promising research direction, we additionally introduced MSMD, a large-scale richly annotated multimodal audio-sheet music dataset. We make both the dataset and our experimental code (including pre-trained embedding models) freely available, hoping to reduce the initial hurdles for working with this kind of data. Finally, we showed that the proposed methodology in combination with the MSMD dataset leads to models that are beginning to generalize to real-world retrieval scenarios with scanned sheet music and real performance audios.

Reproducibility

To reproduce this paper:

- The code and walkthrough for reproducing the MSMD dataset can be found here: <https://github.com/CPJKU/msmd>.
- The code for training and evaluating our models is available here: https://github.com/CPJKU/audio_sheet_retrieval.

Notes

- ¹ <http://www.mutopiaproject.org>.
- ² https://github.com/CPJKU/audio_sheet_retrieval.
- ³ <http://www.lilypond.org>.
- ⁴ <https://github.com/MutopiaProject/MutopiaProject>, commit code e325d7.

Additional File

The additional file for this article can be found as follows:

- **Appendix.** Details on how to reproduce the MSMD dataset. DOI: <https://doi.org/10.5334/tismir.12.s1>

Acknowledgements

This research was supported in part by the European Research Council (ERC) under grant ERC-2014-AdG 670035 (ERC Advanced Grant, project “Con Espressione”). Jan Hajič Jr. wishes to acknowledge support by the Czech Science Foundation grant no. P103/12/G084 and Charles University Grant Agency grant no. 1444217.

Competing Interests

The authors have no competing interests to declare.

References

- Balke, S., Achankunju, S. P., & Müller, M.** (2015). Matching musical themes based on noisy OCR and OMR input. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 703–707. Brisbane, Australia. DOI: <https://doi.org/10.1109/ICASSP.2015.7178060>
- Balke, S., Arifi-Müller, V., Lamprecht, L., & Müller, M.** (2016). Retrieving audio recordings using musical themes. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 281–285. Shanghai, China. DOI: <https://doi.org/10.1109/ICASSP.2016.7471681>
- Böck, S., & Schedl, M.** (2012). Polyphonic piano note transcription with recurrent neural networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 121–124. Kyoto, Japan. DOI: <https://doi.org/10.1109/ICASSP.2012.6287832>
- Byrd, D., & Simonsen, J. G.** (2015). Towards a standard testbed for optical music recognition: Definitions, metrics, and page images. *Journal of New Music Research*, 44(3), 169–195. DOI: <https://doi.org/10.1080/09298215.2015.1045424>
- Casey, M. A., Rhodes, C., & Slaney, M.** (2008). Analysis of minimum distances in high-dimensional musical spaces. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(5), 1015–1028. DOI: <https://doi.org/10.1109/TASL.2008.925883>
- Cheng, T., Mauch, M., Benetos, E., & Dixon, S.** (2016). An attack/decay model for piano transcription. In *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 584–590. New York City, United States.
- Clevert, D., Unterthiner, T., & Hochreiter, S.** (2015). Fast and accurate deep network learning by exponential linear units (ELUs). *International Conference on Learning Representations (ICLR) (arXiv:1511.07289)*.
- Dorfer, M., Arzt, A., & Widmer, G.** (2016). Towards score following in sheet music images. In *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 789–795. New York City, United States.
- Dorfer, M., Arzt, A., & Widmer, G.** (2017a). Learning audio-sheet music correspondences for score identification and offline alignment. In *Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR)*, 115–122. Suzhou, China.
- Dorfer, M., Hajič, J., Jr., & Widmer, G.** (2017b). On the Potential of Fully Convolutional Neural Networks for Musical Symbol Detection. In *Proceedings of the 12th IAPR International Workshop on Graphics Recognition*, 53–54. Kyoto, Japan. DOI: <https://doi.org/10.1109/ICDAR.2017.274>
- Dorfer, M., Schlüter, J., Vall, A., Korzeniowski, F., & Widmer, G.** (2018). End-to-end cross-modality retrieval with CCA projections and pairwise ranking loss. *International Journal of Multimedia Information Retrieval*, 7(2), 117–128. DOI: <https://doi.org/10.1007/s13735-018-0151-5>
- Fremerey, C., Clausen, M., Ewert, S., & Müller, M.** (2009). Sheet music-audio identification. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, 645–650. Kobe, Japan.
- Gallego, A.-J., & Calvo-Zaragoza, J.** (2017). Staffline removal with selectional auto-encoders. *Expert*

- Systems with Applications*, 89, 138–148. DOI: <https://doi.org/10.1016/j.eswa.2017.07.002>
- Grachten, M., Gasser, M., Arzt, A., & Widmer, G.** (2013). Automatic alignment of music performances with structural differences. In *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*, 607–612. Curitiba, Brazil.
- Hajič, J., Jr., & Pecina, P.** (2017). The MUSCIMA++ Dataset for Handwritten Optical Music Recognition. In *14th International Conference on Document Analysis and Recognition (ICDAR)*, 39–46. New York, United States.
- Henriques, J. F., Carreira, J., Caseiro, R., & Batista, J.** (2013). Beyond hard negative mining: Efficient detector learning via block-circulant decomposition. In *IEEE International Conference on Computer Vision (ICCV)*, 2760–2767. Sydney, Australia. DOI: <https://doi.org/10.1109/ICCV.2013.343>
- Ioffe, S., & Szegedy, C.** (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 448–456. Lille, France.
- Izmirli, Ö., & Sharma, G.** (2012). Bridging printed music and audio through alignment using a midlevel score representation. In *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR)*, 61–66. Porto, Portugal.
- Kelz, R., Dorfer, M., Korzeniowski, F., Böck, S., Arzt, A., & Widmer, G.** (2016). On the potential of simple framewise approaches to piano transcription. In *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 475–481. New York City, United States.
- Kingma, D., & Ba, J.** (2015). Adam: A method for stochastic optimization. *International Conference on Learning Representations (ICLR) (arXiv:1412.6980)*.
- Kiros, R., Salakhutdinov, R., & Zemel, R. S.** (2014). Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint (arXiv:1411.2539)*.
- Kurth, F., Müller, M., Fremerey, C., Chang, Y., & Clausen, M.** (2007). Automated synchronization of scanned sheet music with audio recordings. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, 261–266. Vienna, Austria.
- Lin, M., Chen, Q., & Yan, S.** (2014). Network in network. *International Conference on Learning Representations (ICLR) (arXiv:1312.4400)*.
- McFee, B., Humphrey, E. J., & Bello, J. P.** (2015). A software framework for musical data augmentation. In *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, 248–254. Málaga, Spain.
- Rebelo, A., Fujinaga, I., Paszkiewicz, F., Marcal, A. R. S., Guedes, C., & Cardoso, J. S.** (2012). Optical music recognition: state-of-the-art and open issues. *International Journal of Multimedia Information Retrieval*, 1(3), 173–190. DOI: <https://doi.org/10.1007/s13735-012-0004-6>
- Ronneberger, O., Fischer, P., & Brox, T.** (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 234–241. Munich, Germany.
- Rosasco, L., Vito, E. D., Caponnetto, A., Piana, M., & Verri, A.** (2004). Are loss functions all the same? *Neural Computation*, 16(5), 1063–1076. DOI: <https://doi.org/10.1162/089976604773135104>
- Sigtia, S., Benetos, E., & Dixon, S.** (2016). An end-to-end neural network for polyphonic piano music transcription. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(5), 927–939.
- Simonyan, K., & Zisserman, A.** (2015). Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR) (arXiv:1409.1556)*.
- Socher, R., Karpathy, A., Le, Q. V., Manning, C. D., & Ng, A. Y.** (2014). Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2, 207–218.
- Wen, C., Rebelo, A., Zhang, J., & Cardoso, J.** (2015). A new optical music recognition system based on combined neural network. *Pattern Recognition Letters*, 58, 1–7. DOI: <https://doi.org/10.1016/j.patrec.2015.02.002>

How to cite this article: Dorfer, M., Hajič, J., Jr., Arzt, A., Frostel, H., & Widmer, G. (2018). Learning Audio-Sheet Music Correspondences for Cross-Modal Retrieval and Piece Identification. *Transactions of the International Society for Music Information Retrieval*, 1(1), pp. 22–33. DOI: <https://doi.org/10.5334/tismir.12>

Submitted: 25 January 2018

Accepted: 20 March 2018

Published: 04 September 2018

Copyright: © 2018 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

][*Transactions of the International Society for Music Information Retrieval* is a peer-reviewed open access journal published by Ubiquity Press.

OPEN ACCESS 