LEARNING AUDIO – SHEET MUSIC CORRESPONDENCES FOR SCORE IDENTIFICATION AND OFFLINE ALIGNMENT

Matthias Dorfer* **Andreas Arzt*†** **Gerhard Widmer*†**
*Department of Computational Perception, Johannes Kepler University Linz, Austria
†The Austrian Research Institute for Artificial Intelligence (OFAI), Austria
matthias.dorfer@jku.at

ABSTRACT

This work addresses the problem of matching short excerpts of audio with their respective counterparts in sheet music images. We show how to employ neural network-based cross-modality embedding spaces for solving the following two sheet music-related tasks: retrieving the correct piece of sheet music from a database when given a music audio as a search query; and aligning an audio recording of a piece with the corresponding images of sheet music. We demonstrate the feasibility of this in experiments on classical piano music by five different composers (Bach, Haydn, Mozart, Beethoven and Chopin), and additionally provide a discussion on why we expect multi-modal neural networks to be a fruitful paradigm for dealing with sheet music and audio at the same time.

1. INTRODUCTION

Traditionally, automatic methods for linking audio and sheet music data are based on a common mid-level representation that allows for comparison (i.e., computation of distances or similarities) of time points in the audio and positions in the sheet music. Examples of mid-level representations are symbolic descriptions, which involve the error-prone steps of automatic music transcription on the audio side [2, 4, 12, 20] and optical music recognition (OMR) on the sheet music side [3, 9, 19, 24], or spectral features like pitch class profiles (chroma features), which avoid the explicit audio transcription step but still depend on variants of OMR. For examples of the latter approach see, e.g., [8, 11, 15].

In this paper we present a methodology to *directly* learn correspondences between complex audio data and images of the sheet music, circumventing the problematic definition of a mid-level representation. Given short snippets of audio and their respective sheet music images, a cross-modal neural network is trained to learn an embedding space in which both modalities are represented as 32-dimensional vectors. which can then be compared, e.g., via their cosine distance. Essentially, the neural network replaces the complete feature computation process (on both sides) by learning a transformation of data from the audio and from the sheet music to a common vector space.

The idea of matching sheet music and audio with neural networks was recently proposed in [6]. The approach presented here goes beyond that in several respects. First, the network in [6] requires both sheet music and audio as input at the same time to predict which location in the sheet image best matches the current audio excerpt. We 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. These can then be used independently of each other. For example, 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 are demonstrated in this paper). The second aspect in which we go beyond [6] is the sheer complexity of the musical material: while [6] was restricted to simple monophonic melodies, we will demonstrate the power of our method on real, complex pieces of classical music.

We demonstrate the utility of our approach via preliminary results on two real-world tasks. The first is *piece identification:* given an audio rendering of a piece, the corresponding sheet music is identified via cross-modal retrieval. (We should note here that for practical reasons, in our experiments the audio data is synthesized from MIDI – see below). The second task is *audio-to-sheet-music alignment*. Here, the trained network acts as a complex distance function for given pairs of audio and sheet music snippets, which in turn is used by a dynamic time warping algorithm to compute an optimal sequence alignment.

Our main contributions, then, are (1) a methodology for learning cross-modal embedding spaces for relating audio data and sheet music data; (2) data augmentation strategies which allow for training the neural network for this complex task even with a limited amount of data; and (3) first results on two important MIR tasks, using this new approach.

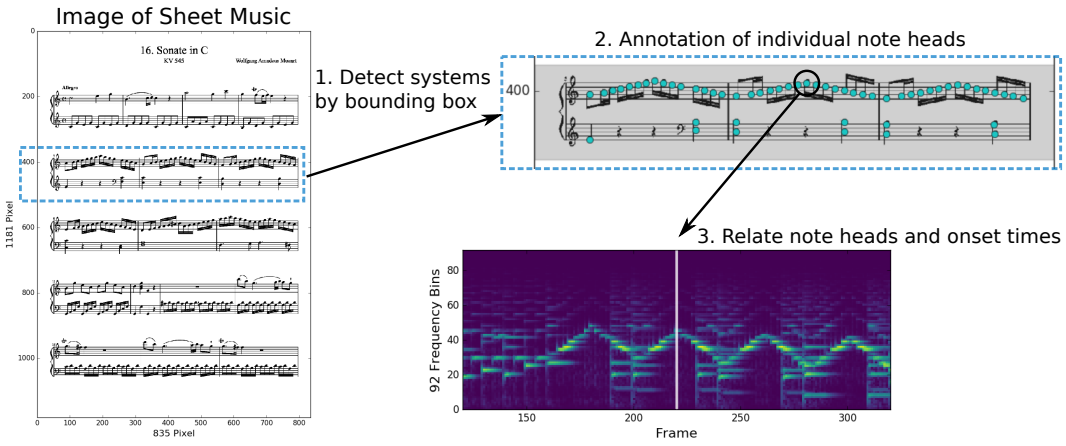

Figure 1. Work flow for preparing the training data (correspondences between sheet music images and the respective music audio). Given the relation between the note heads in the sheet music image and their corresponding onset times in the audio signal we sample audio-sheet-music pairs for training our networks. Figure 2 shows four examples of such training pairs.

2. DESCRIPTION OF DATA

Our approach is built around a neural network designed for learning the relationship between two different data modalities. The network learns its behaviour solely from the examples shown for training. As the presented data is crucial to make this class of models work, we dedicate this section to describing the underlying data as well as the necessary preparation steps needed to generate training examples for optimizing our networks.

2.1 Sheet-Music-Audio Annotation

As already mentioned, we want to address two tasks: (1) sheet music (piece) identification from audio queries and (2) offline alignment of a given audio with its corresponding sheet music image. Both are multi-modal problems involving sheet music images and audio. We therefore start by describing the process of producing the ground truth for learning correspondences between a given score and its respective audio. Figure 1 summarizes the process.

Step one is the localization of staff systems in the sheet music images. In particular, we annotate bounding boxes around the individual systems. Given the bounding boxes we detect the positions of the note heads within each of the systems [1] . The next step is then to relate the note heads to their corresponding onset times in the audio.

Once these relations are established, we know for each note head its location (in pixel coordinates) in the image, and its onset time in the audio. Based on this relationship we cut out corresponding snippets of sheet music images (in our case 180×200 pixels) and short excerpts of audio represented by log-frequency spectrograms (92 bins $\times$ 42 frames). Figure 2 shows four examples of such sheet-music-audio correspondences; these are the pairs presented to our multi-modal networks for training.

[1] We of course do not annotate all of the systems and note heads by hand but use a note head and staff detector to support this tasks (again a neural network trained for this purpose).

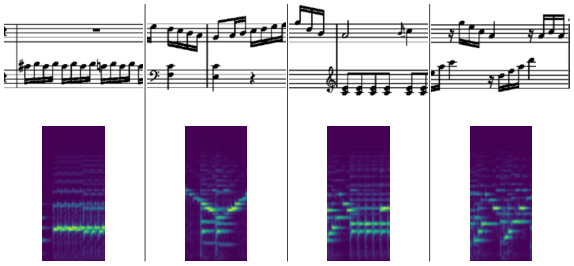

Figure 2. Sheet-music audio correspondences presented to the network for retrieval embedding space learning.

2.2 Composers, Sheet Music and Audio

For our experiments we use classical piano music by five different composers: Mozart (14 pieces), Bach (16), Beethoven (5), Haydn (4) and Chopin (1). To give an impression of the complexity of the music, we have, for instance, Mozart piano sonatas (K.545 1st mvt., K.331 3rd) and symphony transcriptions for piano (Symphony 40 K.550 1st), preludes and fugues from Bach's WTC, Beethoven piano sonata movements and Chopin's Nocturne Op.9 No.1. In terms of experimental setup we will use *only* the 13 pieces of Mozart for training, Mozart's K.545 mvt.1 for validation, and all remaining pieces for testing. This results in 18,432 correspondences for training, 989 for validating, and 11,821 for testing. Our sheet music is collected from *Musescore* [2] where we selected only scores having a 'realistic' layout close to the typesetting of professional publishers [3] . The reason for using Musescore for initial experiments is that along with the sheet music (as *pdf* or image files) Musescore also provides the corresponding *midi* files. This allows us to synthesize the music for each piece of sheet music and to com-

[2] `https://musescore.com`

[3] This is an example of a typical score we used for the experiment (Beethoven Sonata Op.2 No.1): `https://musescore.com/classicman/scores/55331`

pute the exact note onset times from the midis, and thus to establish the required sheet-music audio correspondences.

In terms of audio preparation we compute log-frequency spectrograms of the audios, with a sample rate of 22.05kHz, a FFT window size of 2048 samples, and a computation rate of 20 frames per second. For dimensionality reduction we apply a normalized 16-band logarithmic filterbank allowing only frequencies from $30Hz$ to $16kHz$, which results in 92 frequency bins.

2.3 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) data transformations in order to synthetically increase the effective size of the training set. We already emphasize at this point that data augmentation is a crucial component for learning cross-modality representations that generalize to unseen music, especially when little data is available.

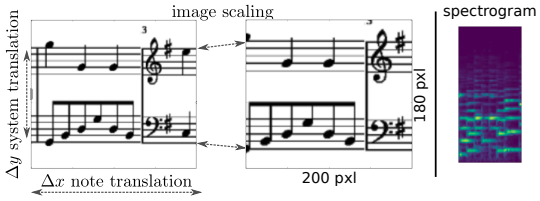

Figure 3. Overview of image augmentation strategies. The size of the sliding image window remains constant (180×200 pixels) but its content changes depending on the augmentations applied. The spectrogram remains the same for the augmented image versions.

For **sheet image augmentation** we apply three different transformations, summarized in Figure 3. 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 terms of **audio augmentation** we render the training pieces with three different sound fonts and additionally vary the tempo between 100 and 130 beats per minute (bpm). The test pieces are all rendered at a rate of 120 bpm using an *additional unseen sound font*. The test set is kept fixed to reveal the impact of the different data augmentation strategies.

3. AUDIO - SHEET MUSIC CORRESPONDENCE LEARNING

This section describes the underlying learning methodology. As mentioned above, the core of our approach is a cross-modality retrieval neural network capable of learning relations between short snippets of audio and sheet music images. In particular, we aim at learning 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 [14, 22]. Before explaining this optimization target, we first introduce the general architecture of our correspondence learning network.

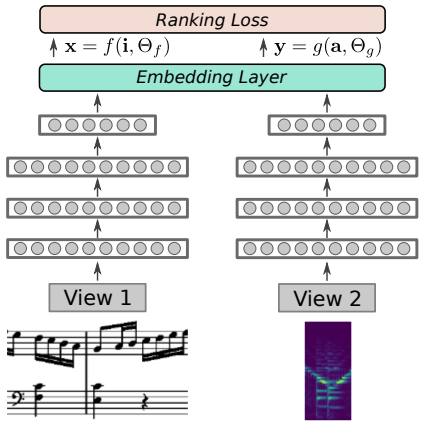

Figure 4. Architecture of correspondence learning network. The network is trained to optimize the similarity (in embedding space) between corresponding audio and sheet image snippets by minimizing a pair-wise ranking loss.

As shown in Figure 4 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 4) is a k-dimensional vector representation encoding the respective inputs. In our case the dimensionality of this representation is 32. We denote these hidden representations by $\mathbf{x} = f(\mathbf{i}, \mathbf{\Theta_f})$ for the sheet image and $\mathbf{y} = g(\mathbf{a}, \mathbf{\Theta_g})$ for the audio spectrogram, respectively, where Θ_f and Θ_g are the parameters of the two networks.

Given this network design, we now explain the pairwise ranking objective. Following [14] 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'):

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

In our application $\mathbf{x}$ is an embedded sample of a sheet image snippet, $\mathbf{y}$ is the embedding of the matching audio ex-

cerpt and $\mathbf{y}_k$ are the embeddings of the *contrastive* (mismatching) audio excerpts (in practice all remaining samples of the current training batch). 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-modality 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 [21] consisting of 3×3 convolutions followed by 2×2 max-pooling as outlined in detail in Table 1. The final convolution layer computes 32 feature maps and is subsequently processed with a global average pooling layer [16] 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 [7] combined with the ranking loss described above. In terms of optimization we use the *adam* update rule [13] 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 ten times to finetune the model.

Table 1. Audio-sheet-music model. BN: Batch Normalization [10], ELU: Exponential Linear Unit [5], MP: Max Pooling, Conv(3, pad-1)-16: 3×3 convolution, 16 feature maps and padding 1.

Sheet-Image 180×200	Audio (Spectrogram) 92×42
2×Conv(3, pad-1)-12	2× Conv(3, pad-1)-12
BN-ELU + MP(2)	BN-ELU + MP(2)
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)-48	2× Conv(3, pad-1)-48
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	

4. EVALUATION OF AUDIO - SHEET CORRESPONDENCE LEARNING

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 2. 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. In the present case, these candidate snippets originate from the 26 unseen test pieces by Bach, Haydn, Beethoven and Chopin. 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

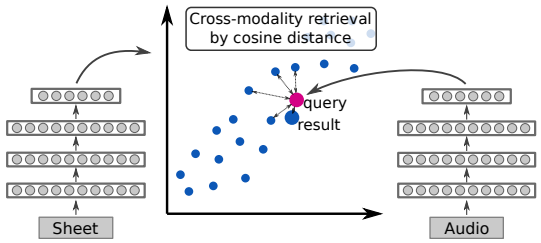

Figure 5. Sketch of sheet-music-from-audio retrieval. The blue dots represent the embedded candidate sheet music snippets. The red dot is the embedding of an audio query. The larger blue dot highlights the closest sheet music snippet candidate selected as retrieval result.

the audio's nearest neighbour $\mathbf{x}_j$ from the set of embedded image snippets as

$$\mathbf{x}_j = \arg\min_{\mathbf{x}_i} \left(1.0 - \frac{\mathbf{x}_i \cdot \mathbf{y}}{||\mathbf{x}_i||\,||\mathbf{y}||}\right) \tag{2}$$

based on their pairwise cosine distance. Figure 5 shows a sketch of this retrieval procedure.

In terms of experimental setup we use the 13 pieces of Mozart for training the network, and the pieces of the remaining composers for testing. As evaluation measures we compute the *Recall@k (R@k)* 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.

Table 2 summarizes the results for the different data augmentation strategies described in Section 2.3. 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. The results are grouped into audio augmentation, sheet augmentation, and applying all or no data augmentation at all. On first sight the retrieval performance appears to be very poor. In particular the *MR* seems hopelessly high in view of our target applications. However, we must remember that our query length is only 42 spectrogram frames ($\approx$ 2 seconds of audio) per excerpt and we select from a set of $11,821$ available candidate snippets. And we will see in the following sections that this retrieval performance is still sufficient to perform tasks such as piece identification. Taking the performance of *no augmentation* as a baseline we observe that all data augmentation strategies help improve the retrieval performance. In terms of audio augmentation we observe that training the model with different synthesizers and varying the tempo works best. From the set of image augmentations, the Δy *system translation* has the highest impact on retrieval performance. Overall we get the best retrieval model when applying *all* augmentation strategies. Note also the large gap between *no augmentation* and *full augmentation*. The median rank, for example, drops from 1042 in case of no augmentation to 168 for full augmentation, which is a substantial improvement.

Audio Augmentation	R@1	R@10	R@25	MR
1 Synth, 100-130bpm	0.37	3.73	7.05	771
3 Synth, 120bpm	0.75	6.26	11.52	559
3 Synth, 100-130bpm	0.87	8.23	15.29	332
Sheet Augmentation				
image scaling	0.75	5.60	10.14	524
Δy system translation	0.91	6.57	12.21	449
Δx note translation	0.44	3.66	7.19	808
full sheet augmentation	0.70	5.72	11.03	496
no augmentation	0.33	2.88	5.71	1042
full augmentation	1.70	11.67	21.17	168
random baseline	0.00	0.03	0.21	5923

Table 2. Influence of data augmentation on audio-to-sheet retrieval. For the audio augmentation experiments no sheet augmentation is applied and vice versa. *no augmentation* represents 1 Synth, 120bpm without sheet augmentation.

A final note: for space reasons we only present results on audio-to-sheet music retrieval, but that the opposite direction using image snippets as search query works analogously and shows similar performance.

5. PIECE IDENTIFICATION

Given the above model that learns to express similarities between sheet music snippets and audio excerpts, we now describe how to use this to solve our first targeted task: identifying the respective piece of sheet music when given an entire audio recording as a query (despite the relatively poor recall and MR for individual queries).

5.1 Description of Approach

We start by preparing 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 $\{\mathbf{i}_{ji}\}$ 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 $\mathbf{x}_{ji} = f(\mathbf{i}_{ji})$.

Sheet snippet retrieval from audio: Given a whole audio recording as a search query we aim at identifying the corresponding piece of sheet music in our database. As with the sheet image we start by cutting the audio (spectrogram) into a set of excerpts $\{\mathbf{a}_1, ..., \mathbf{a}_K\}$ again exhibiting the same dimensions as the spectrograms used for training, and embed all query spectrogram excerpts $\mathbf{a}_k$ with the audio network g. Then we proceed as described in Section 4 and select for each audio its nearest neighbour from the set of all embedded image snippets.

Augmentation	R1	R2	R3	>R3
no augmentation	4	7	1	14
full augmentation	24	2	0	0

Table 3. Influence of data augmentation on piece retrieval.

Piece selection: Since we know for each of the image snippets its originating piece j, we can now have the retrieval image snippets $\mathbf{x}_{ji}$ *vote* for the piece. The piece achieving the highest count of votes is our final retrieval result. In our experiments we consider for each query excerpt its top 25 retrieval results for piece voting.

5.2 Evaluation of Approach

Table 3 summarizes the piece identification results on our test set of Bach, Haydn, Beethoven and Chopin (26 pieces). Again, we investigate the influence of data augmentation and observe that the trend of the experiments in Section 4 is directly reflected in the piece retrieval results. As evaluation measure we compute Rk as the number of pieces ranked at position k when sorting the result list by the number of votes. Without data augmentation only four of the 26 pieces are ranked first in the retrieval lists of the respective full audio recording queries. When making use of data augmentation during training, this number increases substantially and we are able to recognize 24 pieces at position one; the remaining two are ranked at position two. Although this is not the most sophisticated way of employing our network for piece retrieval, it clearly shows the usefulness of our model and its learned audio and sheet music representations for such tasks.

6. AUDIO-TO-SHEET-MUSIC ALIGNMENT

As a second usage scenario for our approach we present the task of audio-to-sheet-music alignment. Here, the goal is to align a performance (given as an audio file) to its respective score (as images of the sheet music), i.e., computing the corresponding location in the sheet music for each time point in the performance, and vice versa.

6.1 Description of Approach

For computing the actual alignments we rely on Dynamic Time Warping (DTW), which is a standard method for sequence alignment [18], and is routinely used in the context of music processing [17]. Generally, DTW takes two sequences as input and computes an optimal non-linear alignment between them, with the help of a local cost measure that relates points of the two sequences to each other.

For our task the two sequences to be aligned are the sequence of snippets from the sheet music image and the sequence of audio (spectrogram) excerpts, as described in Section 2.2. The neural network presented in Section 3 is then used to derive a local cost measure by computing the pairwise cosine distances between the embedded sheet

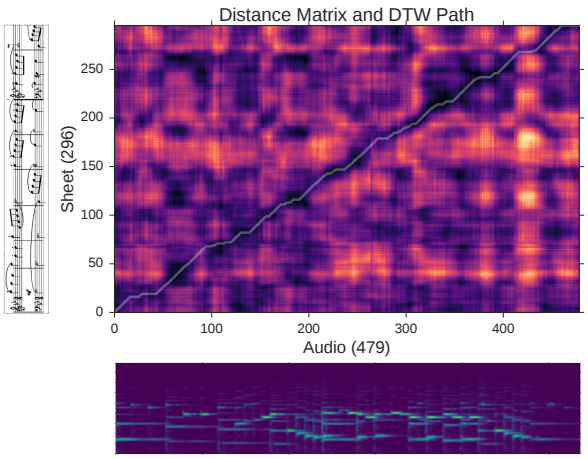

Figure 6. Sketch of audio-to-sheet-music alignment by DTW on a similarity matrix computed on the embedding representation learned by the multi-modal matching network. The white line highlights the path of minimum costs through the sheet music given the audio.

snippets and audio excerpts (see Equation 2). The resulting cost matrix that relates all points of both sequences to each other is shown in Figure 6, for a short excerpt from a simple Bach minuet. Then, the standard DTW algorithm is used to obtain the optimal alignment path.

6.2 Evaluation of Approach

For the evaluation we rely on the same dataset and setup as described above: learning the embedding only on Mozart, then aligning test pieces by Bach, Haydn, Beethoven, Chopin. As evaluation measure we compute the absolute *alignment error* (distance in pixels) of the estimated alignment to its ground truth alignment for each of the sliding window positions. We further normalize the errors by dividing them by the sheet image width to be independent of image resolution. As a naive baseline we compute a linear interpolation alignment which would correspond to a straight line diagonal in the distance matrix in Figure 6. We consider this as a valid reference as we do not consider repetitions for our experiments, yet (in which case things would become somewhat more complicated). We further emphasize that the purpose of this experiment is to provide a proof of concept for this class of models in the context of sheet music alignment tasks, not to compete with existing specialized algorithms for music alignment.

The results are summarized by the boxplots in Figure 7. The median alignment error for the linear baseline is 0.213 normalized image widths ($\approx$ 45 mm in a printed page of sheet music). When computing a DTW path through the distance matrix inferred by our mutimodal audio-sheet-music network this error decreases to 0.041 ($\approx$ 9 mm). Note that values above 1.0 normalized page widths are possible as we handle a piece of sheet music as one single unrolled (concatenated) staff.

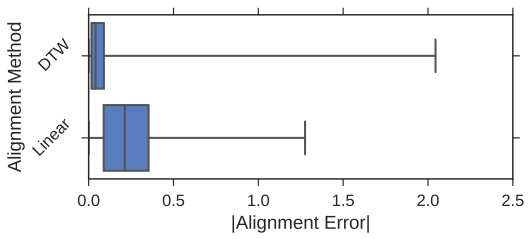

Figure 7. Absolute alignment errors normalized by the sheet image width. We compare the linear baseline with a DTW on the cross-modality distance matrix computed on the embedded audio snippets and spectrogram excerpts.

7. DISCUSSION AND CONCLUSION

We have presented a method for matching short excerpts of audio to their respective counterparts in sheet music images, via a multi-modal neural network that learns relationships between the two modalities, and have shown how to utilize it for two MIR tasks: score identification from audio queries and offline audio-to-sheet-music alignment. Our results provide a proof of concept for the proposed learning-retrieval paradigm and lead to the following conclusions: First, even though little training data is available, it is still possible to use powerful state of the art image and audio models by designing appropriate (task specific) data augmentation strategies. Second, as the best regularizer in machine learning is still a large amount of training data, our results strongly suggest that annotating a truly large dataset will allow us to train general audio-sheet-music-matching models. Recall that for this study we trained on only 13 Mozart pieces, and our model already started to generalize to unseen scores by other composers.

Another aspect of our method is that it works by projecting observations from different modalities into a very low-dimensional joint embedding space. This compact representation is of particular relevance for the task of piece identification as our scoring function – the cosine distance – is a *metric* that permits efficient search in large reference databases [23]. This identification-by-retrieval approach permits us to circumvent solving a large number of local DTW problems for piece identification as done, e.g., in [8].

For now, we have demonstrated the approach on sheet music of realistic complexity, but with synthesized audio (this was necessary to establish the ground truth). The next challenge will be to deal with real audio and real performances, with challenges such as asynchronous onsets, pedal, and varying dynamics.

Finally, we want to stress that our claim is by no means that our proposal in its current stage is competitive with engineered approaches [8, 11, 15] or methods relying on symbolic music or reference performances. These methods have already proven to be useful in real world scenarios, with real performances [1]. However, considering the progress that has been made in terms of score complexity (compared for example to the simple monophonic music used in [6]) we believe it is a promising line of research.

8. ACKNOWLEDGEMENTS

This work is supported by the Austrian Ministries BMVIT and BMWFW, and the Province of Upper Austria via the COMET Center SCCH, and by the European Research Council (ERC Grant Agreement 670035, project CON ESPRESSIONE). The Tesla K40 used for this research was donated by the NVIDIA corporation.

9. REFERENCES

[1] Andreas Arzt, Harald Frostel, Thassilo Gadermaier, Martin Gasser, Maarten Grachten, and Gerhard Widmer. Artificial intelligence in the concertgebouw. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[2] Sebastian Böck and Markus Schedl. Polyphonic piano note transcription with recurrent neural networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 121–124, Kyoto, Japan, 2012.

[3] Donald Byrd and Jakob Grue Simonsen. Towards a standard testbed for optical music recognition: Definitions, metrics, and page images. *Journal of New Music Research*, 44(3):169–195, 2015.

[4] Tian Cheng, Matthias Mauch, Emmanouil Benetos, Simon Dixon, et al. An attack/decay model for piano transcription. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2016.

[5] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *International Conference on Learning Representations (ICLR) (arXiv:1511.07289)*, 2015.

[6] Matthias Dorfer, Andreas Arzt, and Gerhard Widmer. Towards score following in sheet music images. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2016.

[7] Matthias Dorfer, Jan Schlüter, Andreu Vall, Filip Korzeniowski, and Gerhard Widmer. End-to-end cross-modality retrieval with cca projections and pairwise ranking loss. *arXiv preprint (arXiv:1705.06979)*, 2017.

[8] Christian Fremerey, Michael Clausen, Sebastian Ewert, and Meinard Müller. Sheet music-audio identification. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2009.

[9] Jan Hajič jr, Jirı Novotnỳ, Pavel Pecina, and Jaroslav Pokornỳ. Further steps towards a standard testbed for optical music recognition. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2016.

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[11] Özgür İzmirli and Gyanendra Sharma. 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)*, 2012.

[12] Rainer Kelz, Matthias Dorfer, Filip Korzeniowski, Sebastian Böck, Andreas Arzt, and Gerhard Widmer. On the potential of simple framewise approaches to piano transcription. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2016.

[13] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations (ICLR) (arXiv:1412.6980)*, 2015.

[14] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint (arXiv:1411.2539)*, 2014.

[15] Frank Kurth, Meinard Müller, Christian Fremerey, Yoon-ha Chang, and Michael Clausen. Automated synchronization of scanned sheet music with audio recordings. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 261–266, 2007.

[16] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *CoRR*, abs/1312.4400, 2013.

[17] Meinard Müller. *Fundamentals of Music Processing*. Springer Verlag, 2015.

[18] Lawrence Rabiner and Bing-Hwang Juang. *Fundamentals of Speech Recognition*. Prentice Hall Signal Processing Series, 1993.

[19] Ana Rebelo, Ichiro Fujinaga, Filipe Paszkiewicz, Andre R. S. Marcal, Carlos Guedes, and Jaime S. Cardoso. Optical music recognition: state-of-the-art and open issues. *International Journal of Multimedia Information Retrieval*, 1(3):173–190, 2012.

[20] S. Sigtia, E. Benetos, and S. Dixon. An end-to-end neural network for polyphonic piano music transcription. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(5):927–939, 2016.

[21] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR) (arXiv:1409.1556)*, 2015.

[22] Richard Socher, Andrej Karpathy, Quoc V Le, Christopher D Manning, and Andrew Y Ng. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218, 2014.

[23] Stijn Van Dongen and Anton J Enright. Metric distances derived from cosine similarity and pearson and spearman correlations. *arXiv preprint arXiv:1208.3145*, 2012.

[24] Cuihong Wen, Ana Rebelo, Jing Zhang, and Jaime Cardoso. A new optical music recognition system based on combined neural network. *Pattern Recognition Letters*, 58:1–7, 2015.