LEARNING AUDIO – SHEET MUSIC CORRESPONDENCES

Matthias Dorfer

Department of Computational Perception

JOHANNES KEPLER
UNIVERSITY LINZ

Short Introduction ...

I am a PhD Candidate in the Department of Computational Perception at Johannes Kepler University Linz (JKU).

My supervisor
Prof. Gerhard Widmer

Short Introduction ...

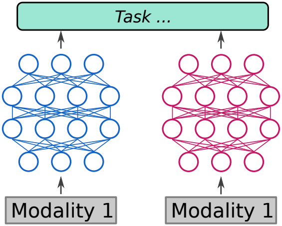
I am a PhD Candidate in the Department of Computational Perception at Johannes Kepler University Linz (JKU).

My supervisor
Prof. Gerhard Widmer

"***Basic and applied research in machine learning**, pattern recognition, knowledge extraction, and generally Artificial and Computational Intelligence. ... **focus is on intelligent audio (specifically: music) processing**."*

This Talk Is About ...

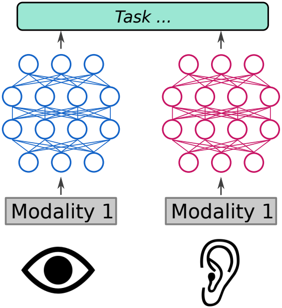
Multi-Modal Neural Networks



This Talk Is About ...

Multi-Modal Neural Networks

Audio-Visual
Representation Learning

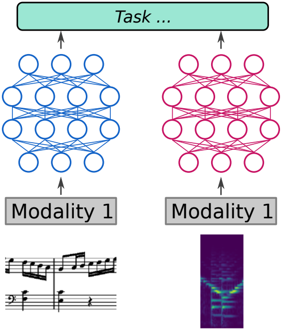


This Talk Is About ...

Multi-Modal Neural Networks

Audio-Visual
Representation Learning

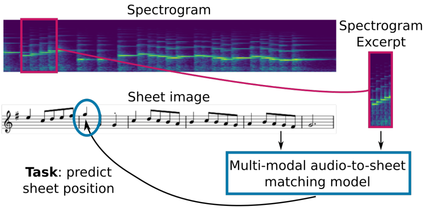
Learning Correspondences
between Audio and Sheet-Music



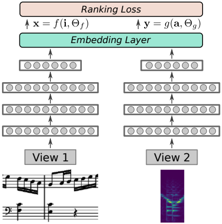
OUR TASKS

Our Tasks

Score Following (Localization)



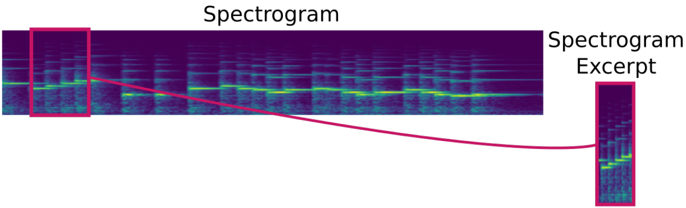
Cross-Modality Retrieval



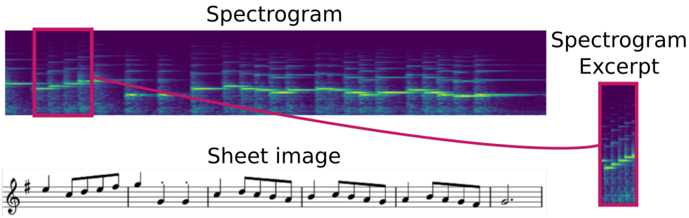
Task - Score Following

Score Following is the process of following a musical performance (audio) with respect to a known **symbolical representation** (e.g. a score).

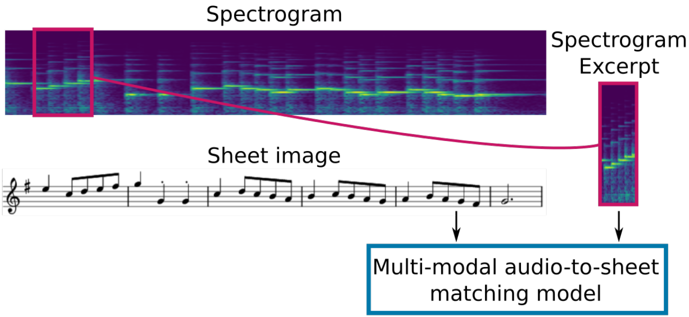
The Task: Audio to Sheet Matching



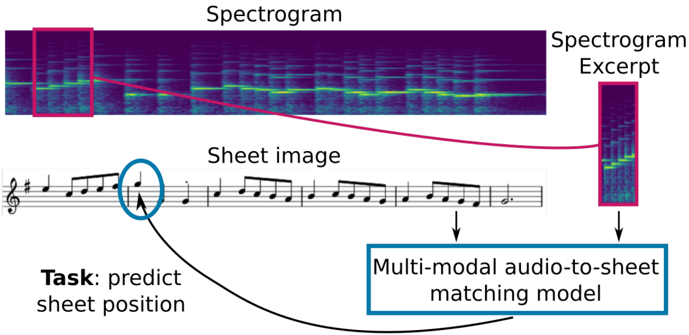
The Task: Audio to Sheet Matching



The Task: Audio to Sheet Matching



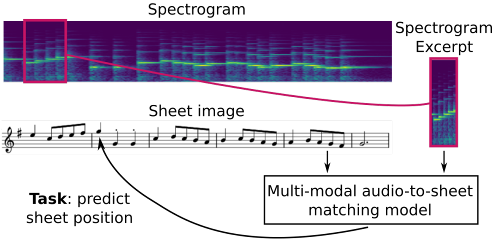
The Task: Audio to Sheet Matching



The Task: Audio to Sheet Matching

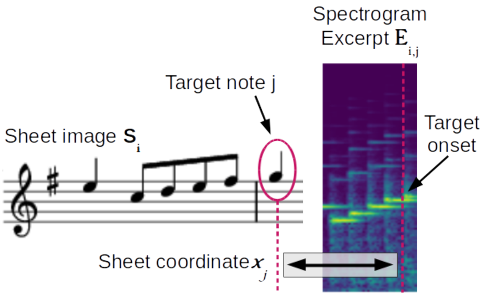
Simultaneously **learn** (in end-to-end neural network fashion) to

- read notes from images (pixels)
- listen to music
- match played music to its corresponding notes



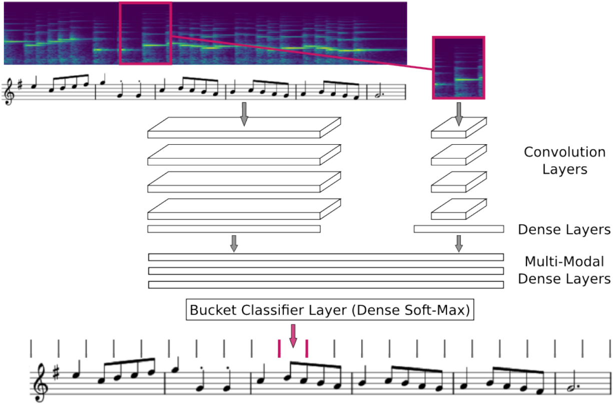
METHODS

Spectrogram to Sheet Correspondences



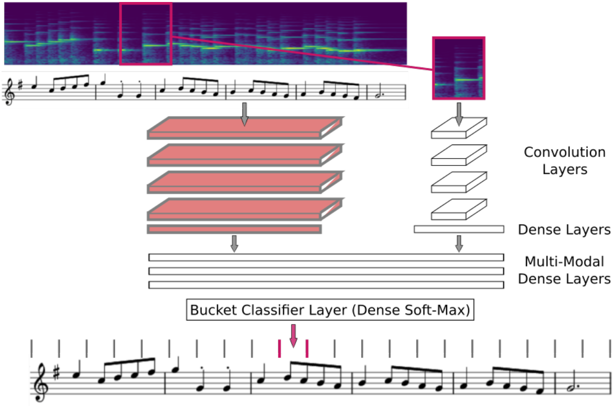
- Rightmost onset is target note onset
- Temporal context of 1.2 sec into the past

Multi-modal Convolution Network



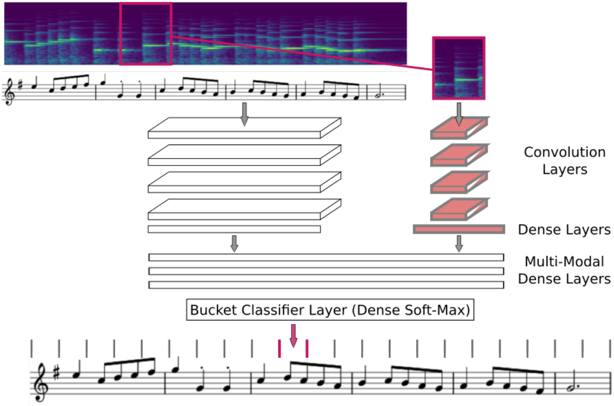
The output layer is a B**-way soft-max**!

Multi-modal Convolution Network



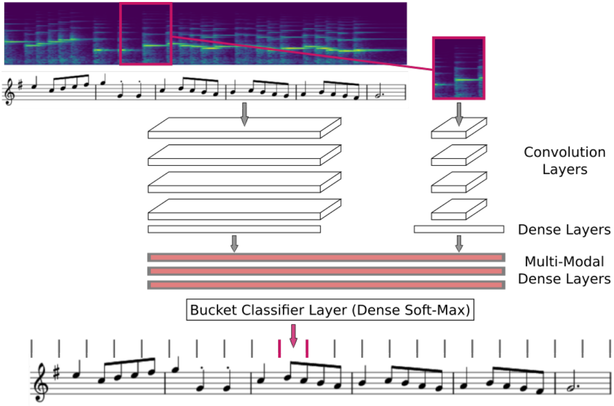
The output layer is a B-**way soft-max**!

Multi-modal Convolution Network



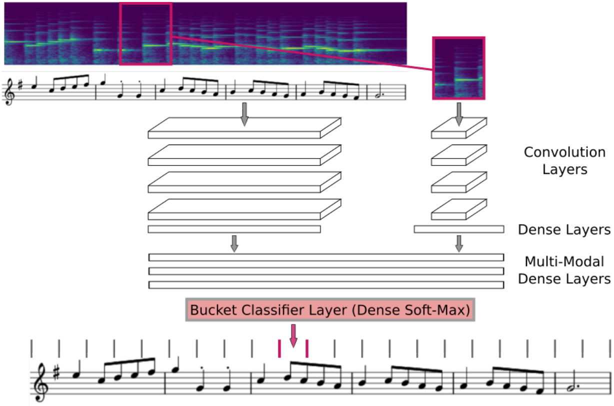
The output layer is a B**-way soft-max**!

Multi-modal Convolution Network



The output layer is a B**-way soft-max**!

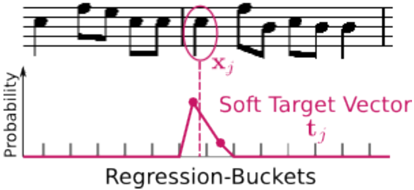
Multi-modal Convolution Network



The output layer is a B**-way soft-max**!

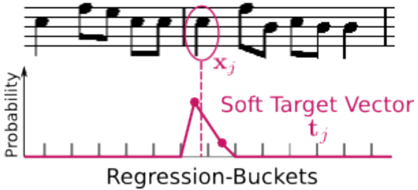
Soft Target Vectors

- Staff image is quantized into buckets
- Each bucket is represented by one output neuron
- Buckets hold probability of containing the note
- Neighbouring buckets share probability $\rightarrow$ soft targets



Soft Target Vectors

- Staff image is quantized into buckets
- Each bucket is represented by one output neuron
- Buckets hold probability of containing the note
- Neighbouring buckets share probability $\rightarrow$ soft targets



- Used as target values for training our networks

Optimization Objective

Output activation: B-way soft-max

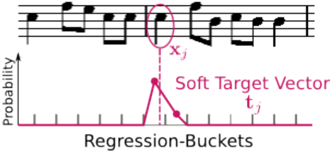
$$\phi(y_{j,b}) = \frac{e^{y_{j,b}}}{\sum_{k=1}^{B} e^{y_{j,k}}}$$

Optimization Objective

Output activation: B-way soft-max

$$\phi(y_{j,b}) = \frac{e^{y_{j,b}}}{\sum_{k=1}^{B} e^{y_{j,k}}}$$

Soft targets $\mathbf{t}_j$

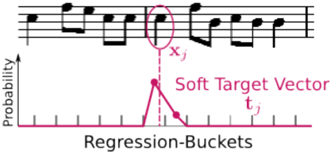


Optimization Objective

Output activation: B-way soft-max

$$\phi(y_{j,b}) = \frac{e^{y_{j,b}}}{\sum_{k=1}^{B} e^{y_{j,k}}}$$

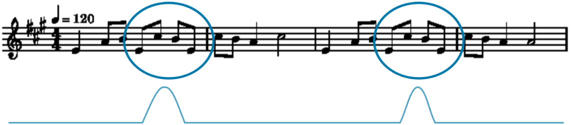
Soft targets $\mathbf{t}_j$



Loss: Categorical Cross Entropy

$$l_j(\Theta) = -\sum_{k=1}^{B} t_{j,k}\, log(p_{j,k})$$

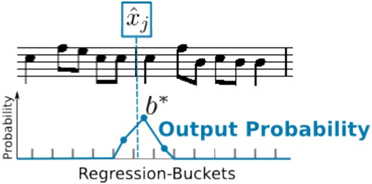
Discussion: Choice of Objective



- Allows to model uncertainties
  (e.g. repetitive structures in music)
- Our experience: Much nicer to optimize than
  MSE regression or Mixture Density Networks

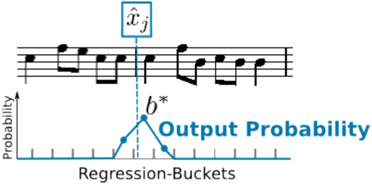
Sheet Location Prediction

At test time: Predict expected location $\hat{x}_j$ of audio snippet with target note j in sheet image.



Sheet Location Prediction

At test time: Predict expected location $\hat{x}_j$ of audio snippet with target note j in sheet image.



Probability weighted localization

$$\hat{x}_j = \sum_{k \in \{b^*-1, b^*, b^*+1\}} w_k c_k$$

- bucket b^* with highest probability $\mathbf{p}_j$
- weights $\mathbf{w} = \{p_{j,b^*-1}, p_{j,b^*}, p_{j,b^*+1}\}$,
- bucket coordinates c_k

EXPERIMENTS / DEMO

Train / Evaluation Data

Matthias Dorfer, Andreas Arzt, and Gerhard Widmer.
"Towards Score Following in Sheet Music Images."
In Proc. of 17th International Society for Music Information Retrieval Conference, 2016.

- Trained on monophonic piano music
- Localization of staff lines
- Synthesize midi-tracks to audio
- Signal processing
  - Spectrogram (22.05 kHz, 2048 window, 31.25 fps)
  - Filterbank: 24 band logarithmic (80 Hz to 8 kHz)

Model Architecture and Optimization

Sheet-Image 40×390	Spectrogram 136×40
VGG style image model	**VGG style audio model**
3×3 Conv, BN, ReLU	3×3 Conv, BN, ReLU
Max pooling	Max pooling
Dense, BN, ReLu, Drop-Out	Dense, BN, ReLu, Drop-Out
Multi-modality merging	
Concatenation-Layer	
Dense, BN, ReLu, Drop-Out	
Dense, BN, ReLu, Drop-Out	
B-way Soft-Max Layer	

Model Architecture and Optimization

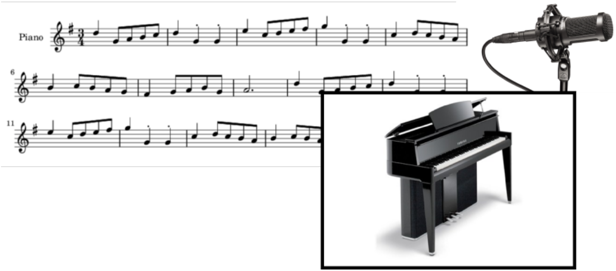
Sheet-Image 40×390	Spectrogram 136×40
VGG style image model	**VGG style audio model**
3×3 Conv, BN, ReLU	3×3 Conv, BN, ReLU
Max pooling	Max pooling
Dense, BN, ReLu, Drop-Out	Dense, BN, ReLu, Drop-Out
Multi-modality merging	
Concatenation-Layer	
Dense, BN, ReLu, Drop-Out	
Dense, BN, ReLu, Drop-Out	
B-way Soft-Max Layer	

- Mini-batch stochastic gradient descent with momentum
  - Mini-batch size: 100
  - Learning rate: 0.1 (divided by 10 every 10 epochs)
  - Momentum: 0.9
  - Weight decay: 0.0001

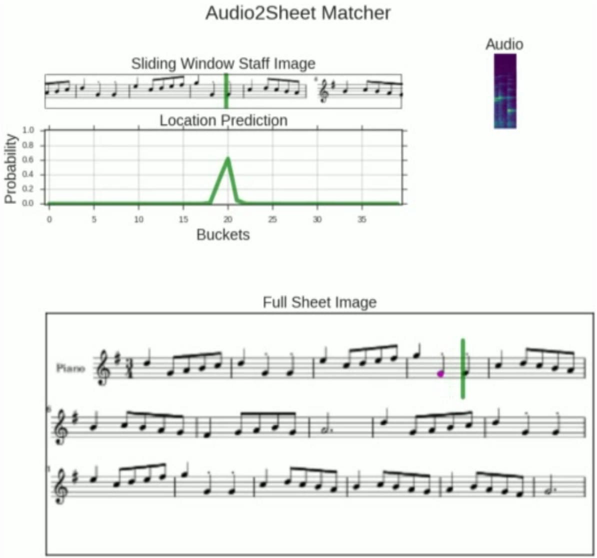
Demo with Real Music

Minuet in G Major (*BWV Anhang 114, Johann Sebastian Bach*)

- Played on Yamaha AvantGrand N2 hybrid piano
- Recorded using a single microphone



Demo with Real Music



So far so good ...

Model works well on monophonic music
and seems to learn reasonable representations.

Important observation: No temporal model required!

What to do next?

Switch to "Real Music"

Switch to "Real Music"

Switch to "Real Music"

Composers, Sheet Music and Audio

- Pieces from **MuseScore** (annotating becomes feasible)
- **Classical Piano Music** by Mozart (14 pieces), Bach (16), Beethoven (5), Haydn (4) and Chopin (1)
- Experimental Setup:
  train / **validate**: Mozart | **test**: all composers
- Audio is synthesized

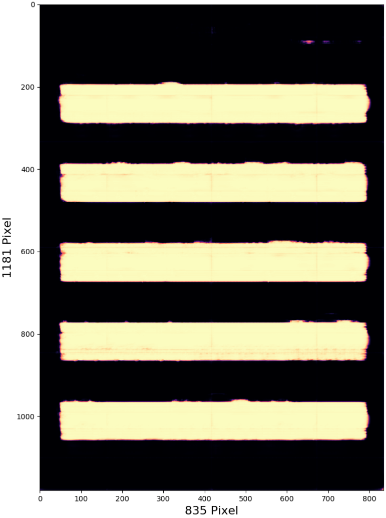
ANNOTATION PIPELINE

Fully Convolutional Segmentation Networks

Optical Music Recognition (OMR) Pipeline

1. Input Image

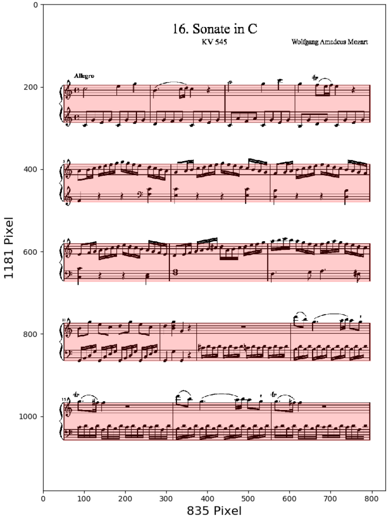
Fully Convolutional Segmentation Networks



Optical Music Recognition (OMR) Pipeline

1. Input Image
2. System Probability Maps

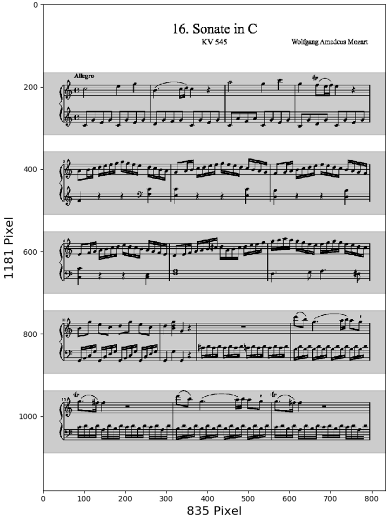
Fully Convolutional Segmentation Networks



Optical Music Recognition (OMR) Pipeline

1. Input Image
2. System Probability Maps
3. Systems Recognition

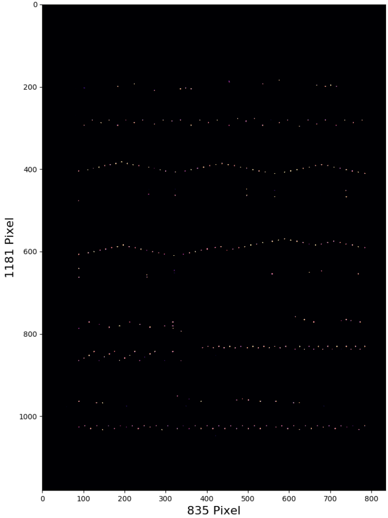
Fully Convolutional Segmentation Networks



Optical Music Recognition (OMR) Pipeline

1. Input Image
2. System Probability Maps
3. Systems Recognition
4. Regions of Interest

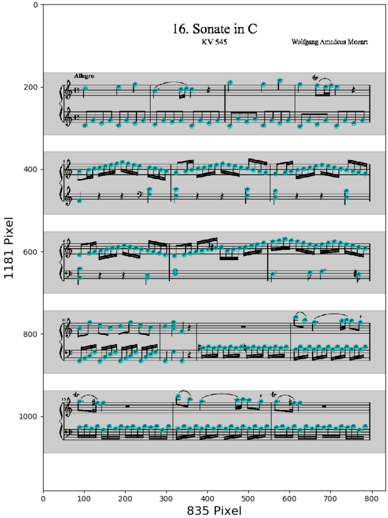
Fully Convolutional Segmentation Networks



Optical Music Recognition (OMR) Pipeline

1. Input Image
2. System Probability Maps
3. Systems Recognition
4. Regions of Interest
5. Note Probability Maps

Fully Convolutional Segmentation Networks

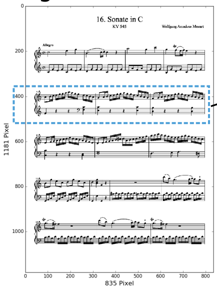


Optical Music Recognition (OMR) Pipeline

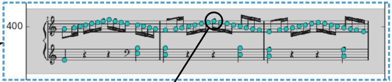
1. Input Image
2. System Probability Maps
3. Systems Recognition
4. Regions of Interest
5. Note Probability Maps
6. Note Head Recognition

Annotation Pipeline

Image of Sheet Music

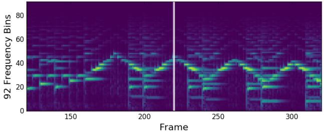


2. Annotation of individual note heads

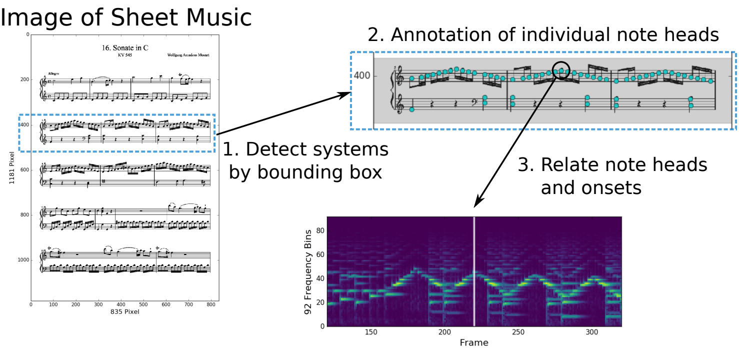


1. Detect systems
by bounding box

3. Relate note heads
and onsets



Annotation Pipeline

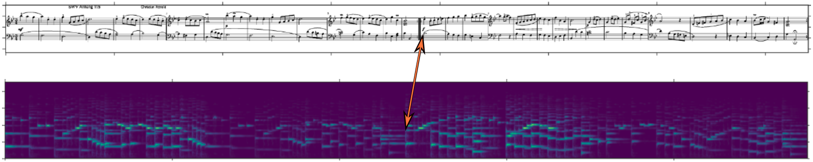


Now we know

- the locations of staff systems and note heads and for each note head its onset time in the audio.
- overall **63836** annotated correspondences of **51 pieces**.

Train Data Preparation

We unroll the score and have the relations to the audio



This is all we need to train our models!

Demo

W.A. Mozart
Piano Sonata K545, 1st Movement

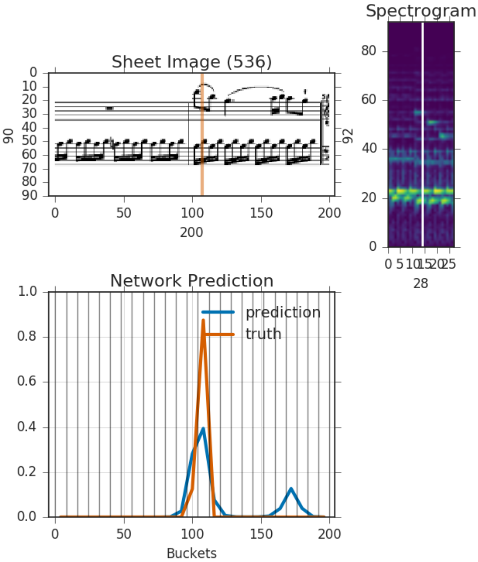
Plain, Frame-wise
Multi-Modal Convolution Network

Observations

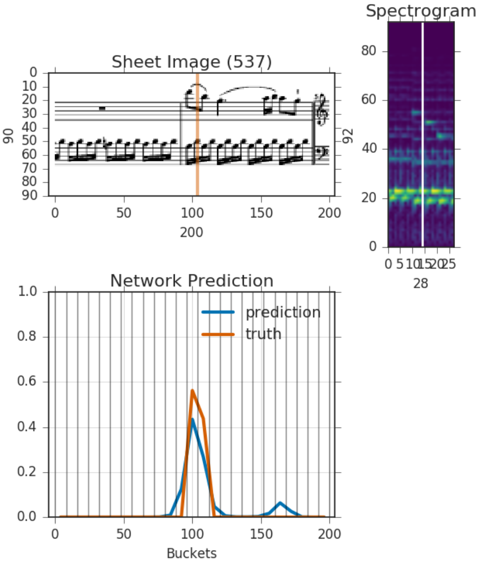
- Sometimes a bit shaky
- **Score following fails** at the beginning of second page!

But why?

Failure

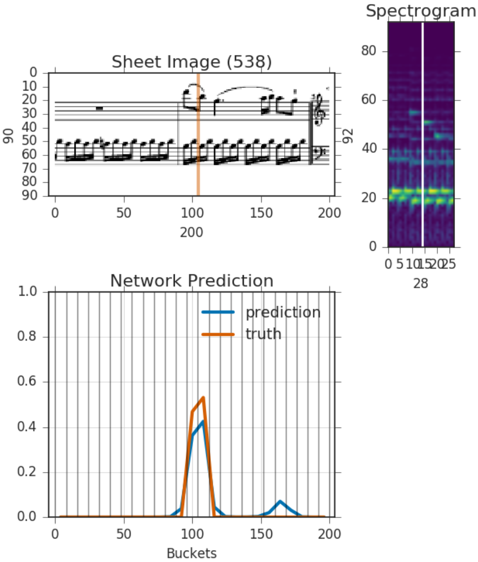

Sheet Image (536)
Spectrogram
Network Prediction
prediction
truth
Buckets
200
90
92
28

Failure

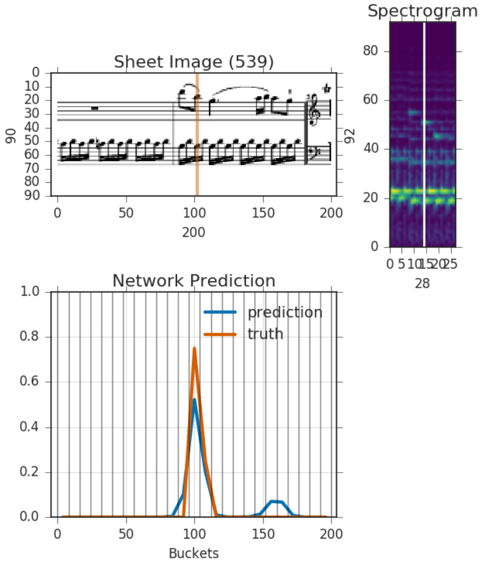


Sheet Image (537)
Spectrogram
Network Prediction
prediction
truth
Buckets

Failure

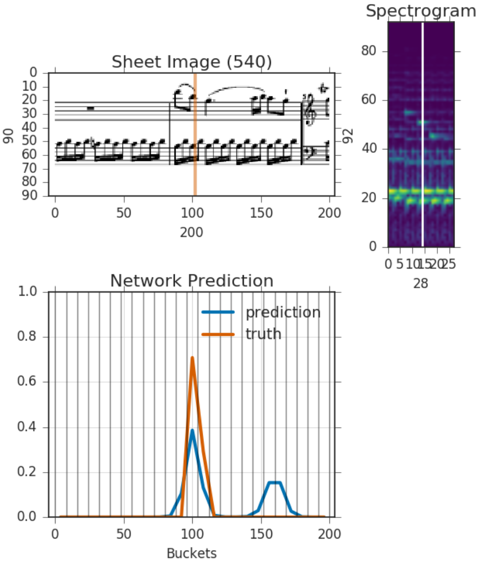

Sheet Image (538)
Spectrogram
Network Prediction
prediction
truth
Buckets
200
90
92
28
0 50 100 150 200
0 10 20 30 40 50 60 70 80 90
0 20 40 60 80
0 5 10 15 20 25
0.0 0.2 0.4 0.6 0.8 1.0

Failure

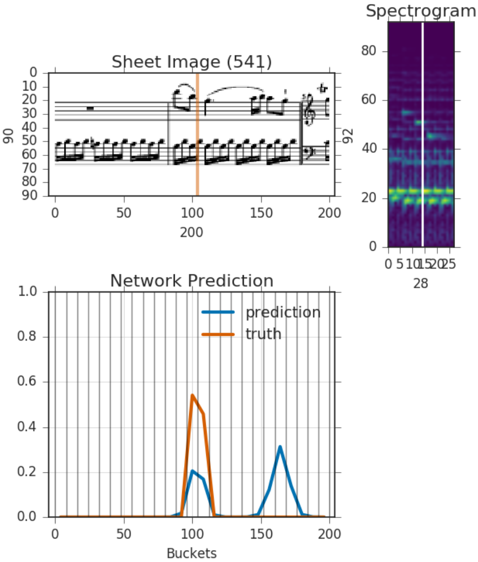


Sheet Image (539)
0
10
20
30
40
50
60
70
80
90
90
0
50
100
150
200
200
92
Spectrogram
80
60
40
20
0
0 5 10 15 20 25
28
Network Prediction
1.0
0.8
0.6
0.4
0.2
0.0
0
50
100
150
200
Buckets
prediction
truth

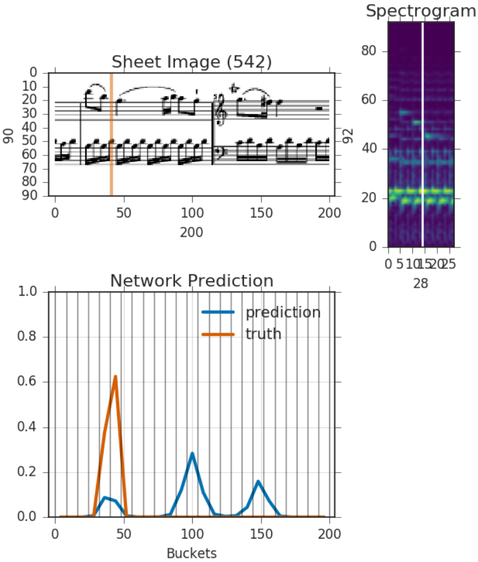
Failure


Sheet Image (540)
Spectrogram
Network Prediction
prediction
truth
Buckets
90
92
200
28

Failure

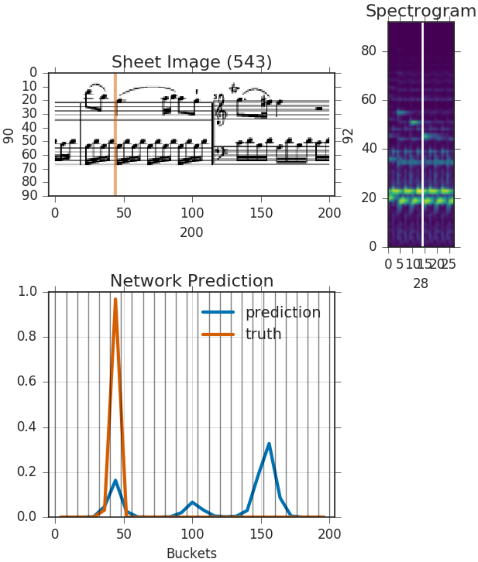

Sheet Image (541)
Spectrogram
Network Prediction
prediction
truth
Buckets
90
92
200
28

Failure



Sheet Image (542)
0
10
20
30
40
50
60
70
80
90
0 50 100 150 200
200
90
92
Spectrogram
0 20 40 60 80
0 5 10 15 20 25
28
Network Prediction
prediction
truth
0.0 0.2 0.4 0.6 0.8 1.0
0 50 100 150 200
Buckets

Failure



Sheet Image (543)
Spectrogram
Network Prediction
prediction
truth
Buckets
200
90
92
28

NET DEBUGGING

Guided Back-Propagation

Springenberg et al., "Striving for Simplicity - The All Convolutional Net", 2016.

Saliency Maps for **understanding trained models**

Guided Back-Propagation

Springenberg et al., "Striving for Simplicity - The All Convolutional Net", 2016.

Saliency Maps for **understanding trained models**

Given a **trained network** $\mathbf{f}$ and a **fixed input** $\mathbf{X}$ we compute the gradient of network prediction $\mathbf{f}(\mathbf{X}) \in \mathbb{R}^k$ with respect to its input

$$\frac{\partial \max(\mathbf{f}(\mathbf{X}))}{\partial \mathbf{X}} \tag{1}$$

Determines those parts of the input having the highest effect on the prediction when changed.

Guided Back-Propagation

Springenberg et al., "Striving for Simplicity - The All Convolutional Net", 2016.

Saliency Maps for **understanding trained models**

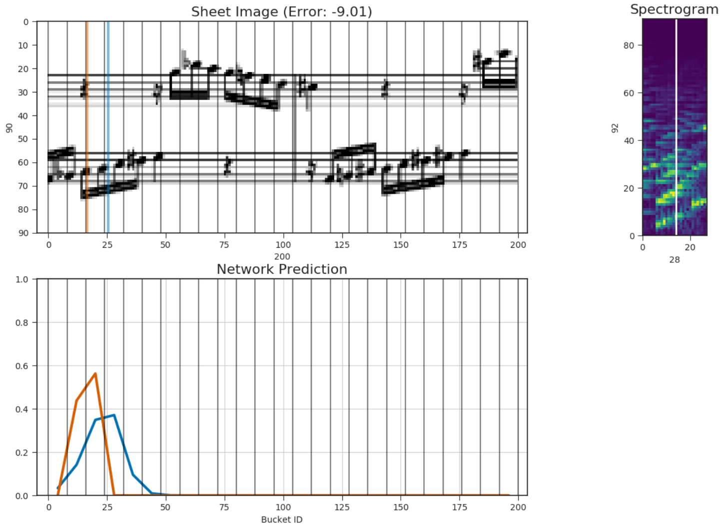
Given a **trained network** $\mathbf{f}$ and a **fixed input** $\mathbf{X}$ we compute the gradient of network prediction $\mathbf{f}(\mathbf{X}) \in \mathbb{R}^k$ with respect to its input

$$\frac{\partial \max(\mathbf{f}(\mathbf{X}))}{\partial \mathbf{X}} \tag{1}$$

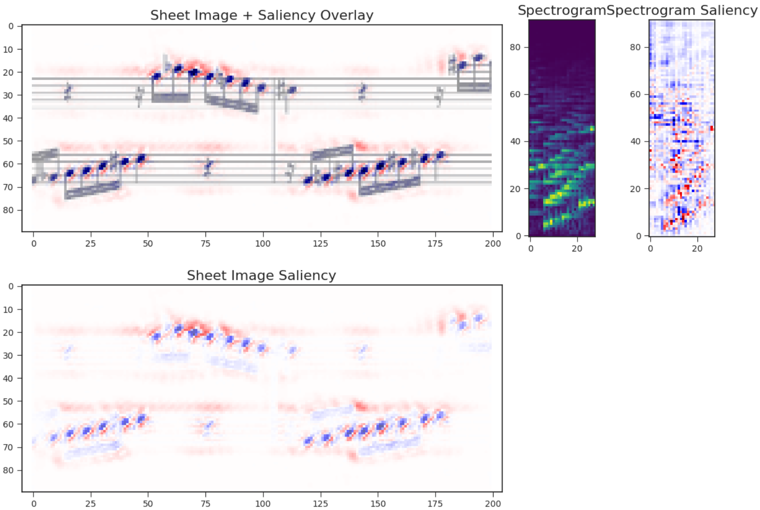
Determines those parts of the input having the highest effect on the prediction when changed.

Guided back-propagation with rectified linear units only back-propagates positive error signals $\delta_{l-1} = \delta_l \, \mathbf{1}_{x>0} \, \mathbf{1}_{\delta_l>0}$

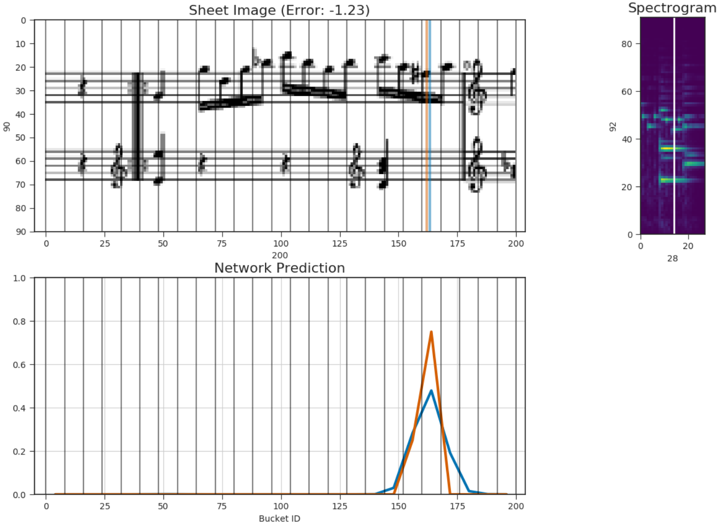
Net Debugging



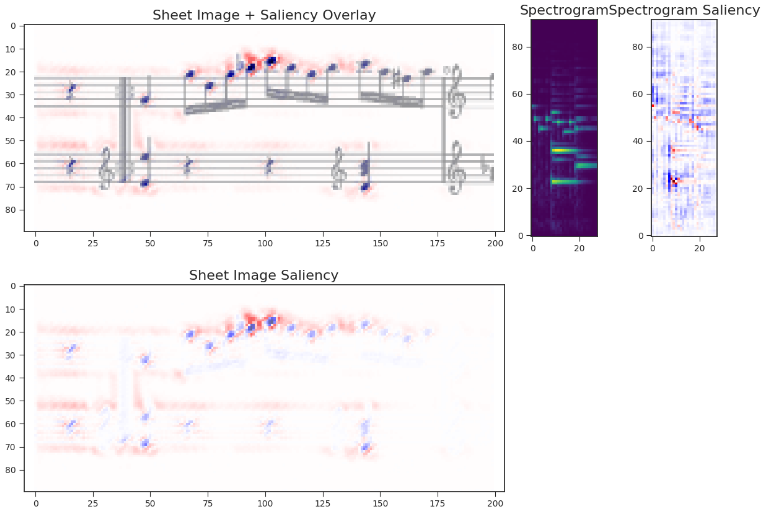
Net Debugging



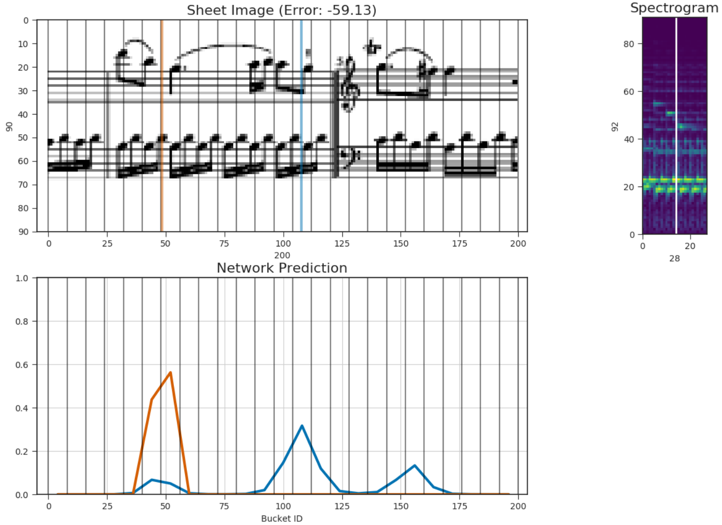
Net Debugging



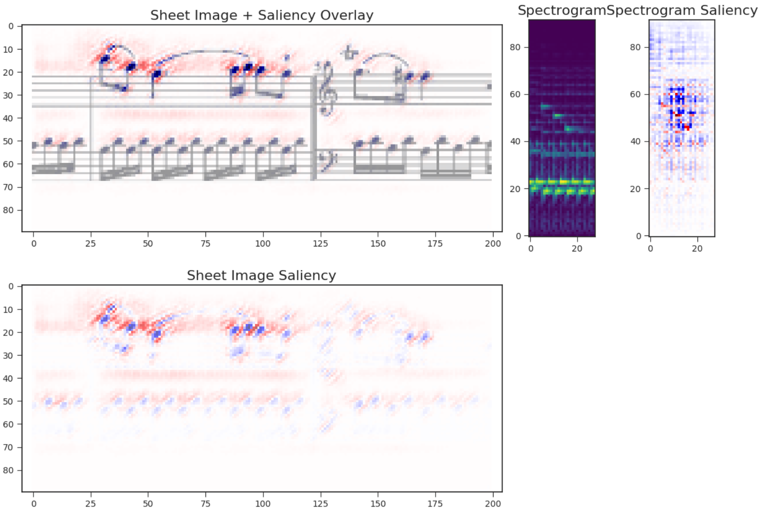
Net Debugging



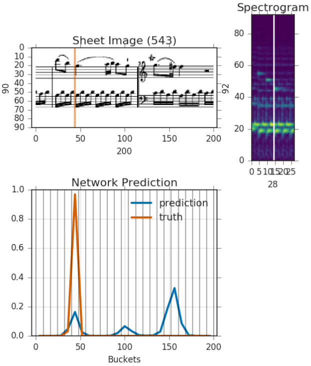
Net Debugging



Net Debugging



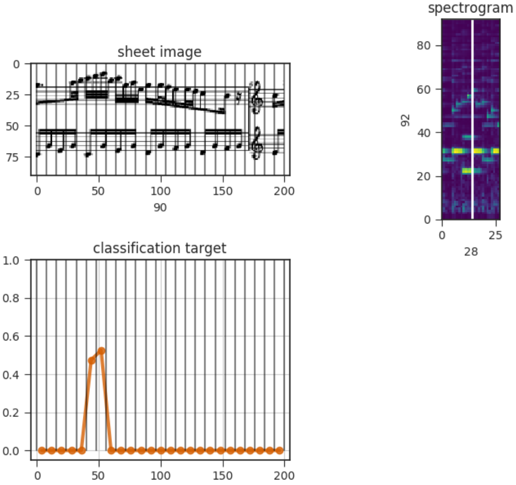
Failure Analysis Continued



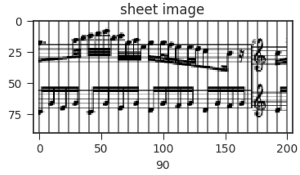
- Network pays attention to note heads but does not seem to be pitch sensitive
- However, exploiting **temporal relations** inherent in music could fix the problem!

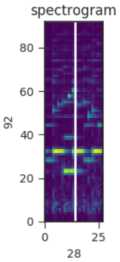
RECURRENT NEURAL NETWORKS!

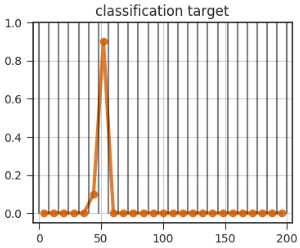
RNN Training Examples



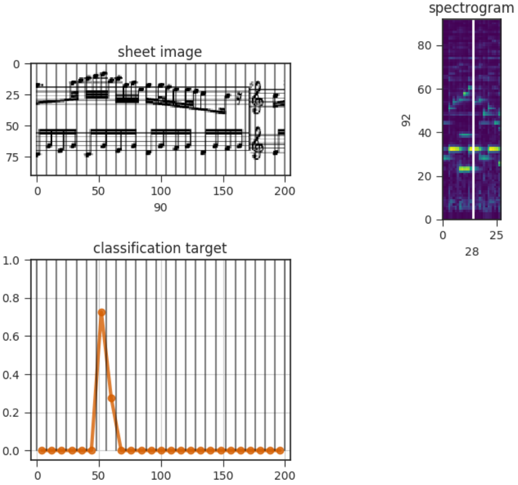
RNN Training Examples



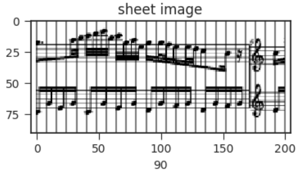


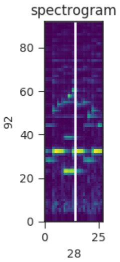


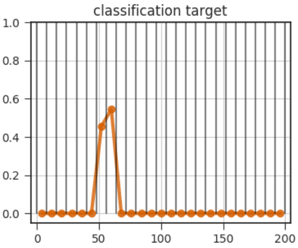
RNN Training Examples



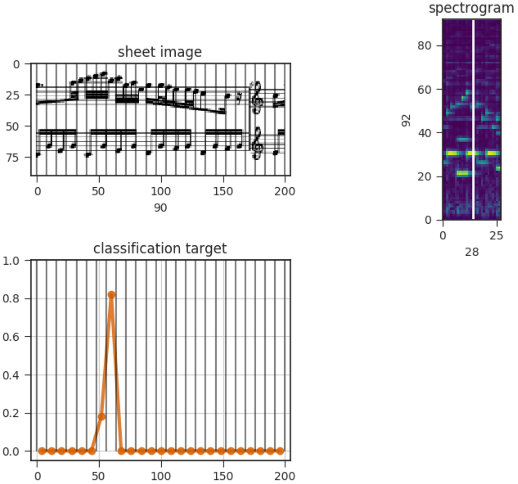
RNN Training Examples



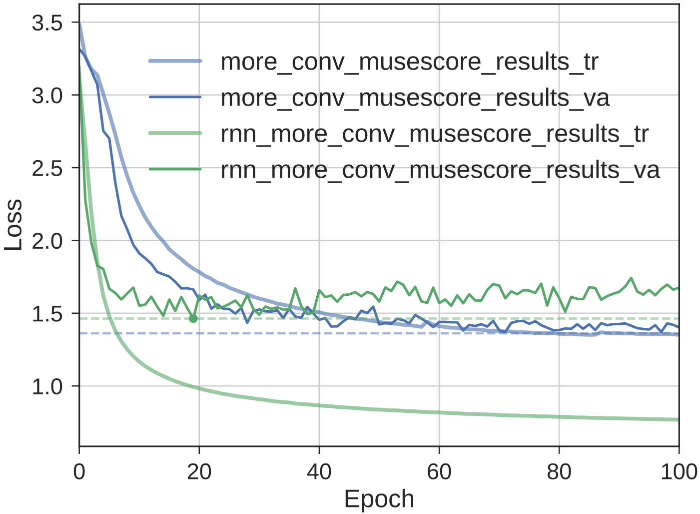




RNN Training Examples



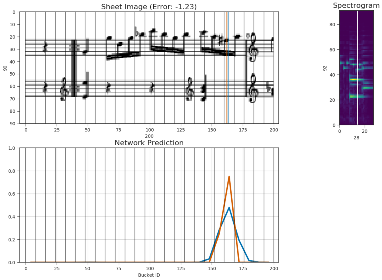
RNN Learning Curves



HIDDEN MARKOV MODELS (HMMS)

Hidden Markov Models

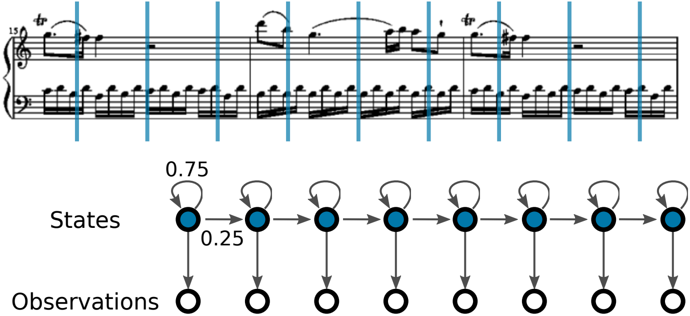
Enforce spatial and temporal structure into single-time-step prediction score-following-model.



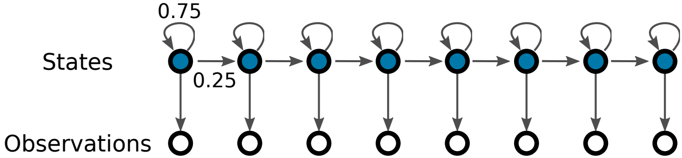
HMM - Design

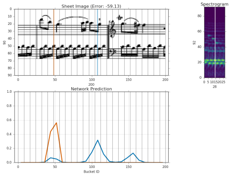
HMM - Design

HMM - Design



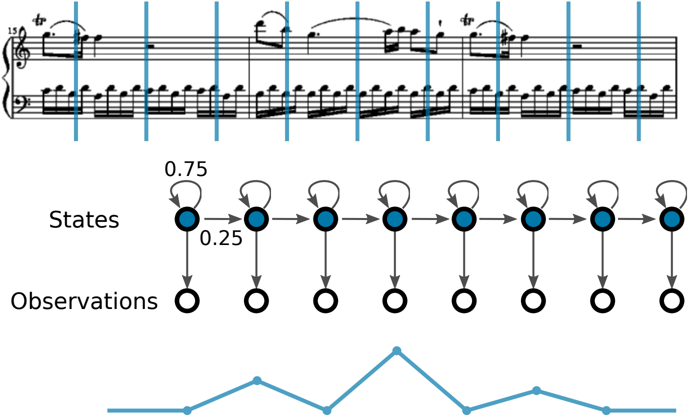
HMM - Design





Map **Local Predictions**
to Global Sheet Image
and
use them as **Observations**

HMM - Design



Apply HMM Filtering / Tracking Algorithm

HMM - Demo

W.A. Mozart
Piano Sonata K545, 1st Movement

HMM-Tracker
Multi-Modal Convolution Network

CONCLUSIONS

Conclusions

Learning multi-modal representations in the context of music-audio and sheet-music is a challenging application.

Conclusions

Learning multi-modal representations in the context of music-audio and sheet-music is a challenging application.

Multi-Modal Convolution Networks are the right direction.

Conclusions

Learning multi-modal representations in the context of music-audio and sheet-music is a challenging application.

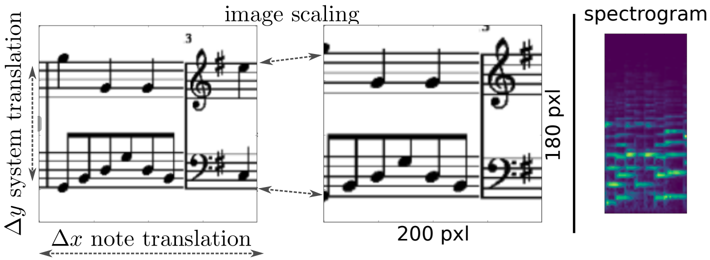
Multi-Modal Convolution Networks are the right direction.

However there are many open problems left:

- **Learning Temporal Relations** from training data
- Real audio and real performances, (asynchronous onsets, pedal, and varying dynamics)
- **More training data!**
- ...

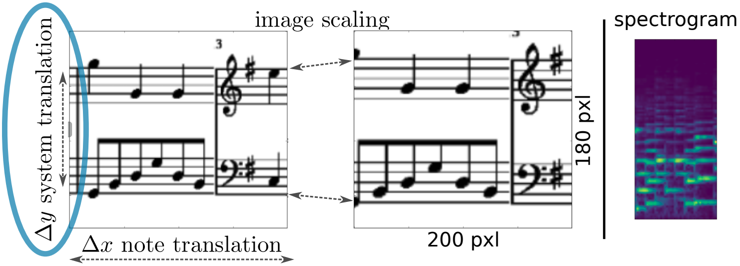
Data Augmentation

Image augmentation:



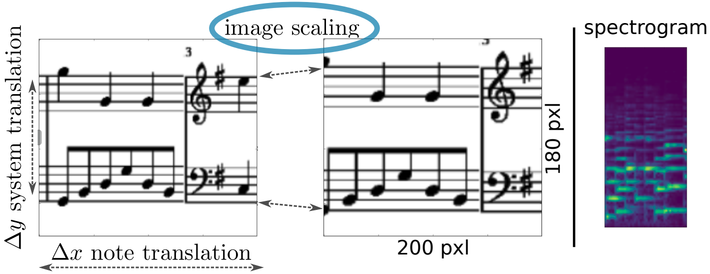
Data Augmentation

Image augmentation:



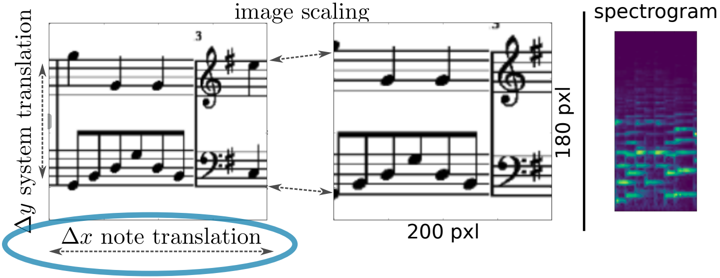
Data Augmentation

Image augmentation:



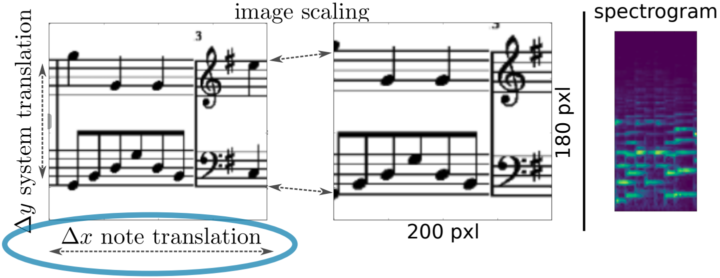
Data Augmentation

Image augmentation:



Data Augmentation

Image augmentation:



Audio augmentation

Different tempi and sound founts

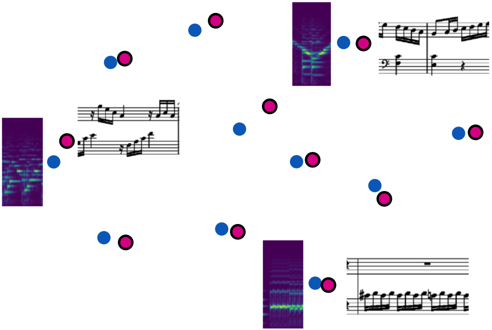
AUDIO - SHEET MUSIC CROSS-MODALITY RETRIEVAL

The Task

Our Goal: **Find a common vector representation** of both audio and sheet music (low dimensional embedding)

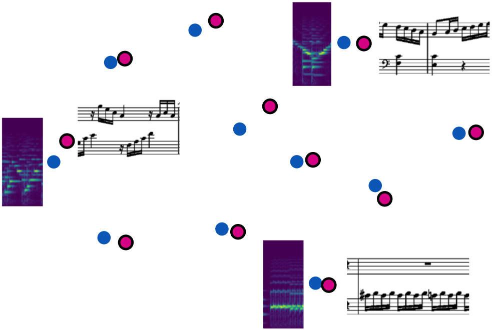
The Task

Our Goal: **Find a common vector representation** of both audio and sheet music (low dimensional embedding)



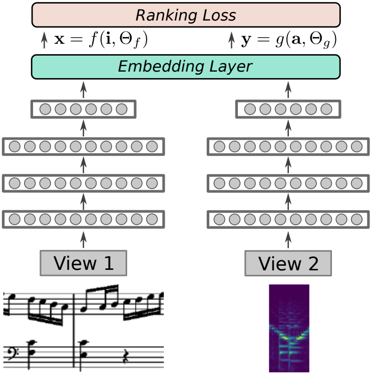
The Task

Our Goal: **Find a common vector representation** of both audio and sheet music (low dimensional embedding)



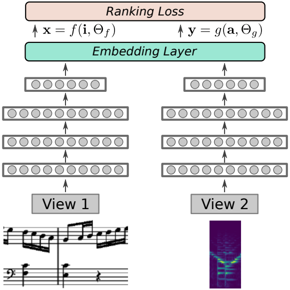
Why would we like this: **to make them comparable.**

Cross-Modality Retrieval Neural Network



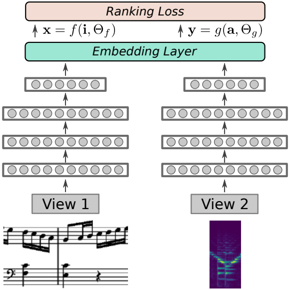
Optimizes the similarity (in embedding space) between corresponding audio and sheet image snippets

Model Details and Optimization



- Uses CCA Embedding Layer
- Trained with Pairwise Ranking Loss
- 32-dimensional embedding

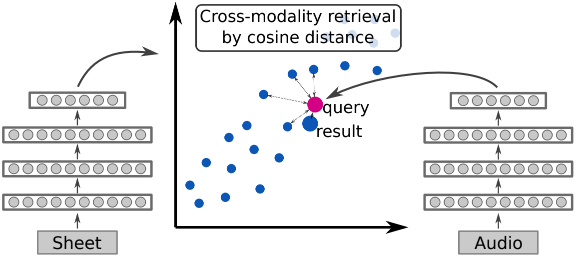
Model Details and Optimization



- Uses CCA Embedding Layer
- Trained with Pairwise Ranking Loss
- 32-dimensional embedding

Encourage an embedding space where
the distance between matching samples is lower than the
distance between mismatching samples.

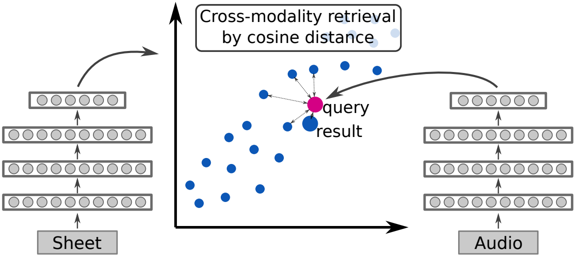
Cross-Modality Retrieval



Audio query point of view:

- blue dots: embedded candidate sheet music snippets
- red dot: embedding of an audio query.

Cross-Modality Retrieval



Audio query point of view:

- blue dots: embedded candidate sheet music snippets
- red dot: embedding of an audio query.

→ **Retrieval by nearest neighbor search**