Neuro-Ear

Alim Bukharaev

Optimization Class Project. MIPT

Introduction

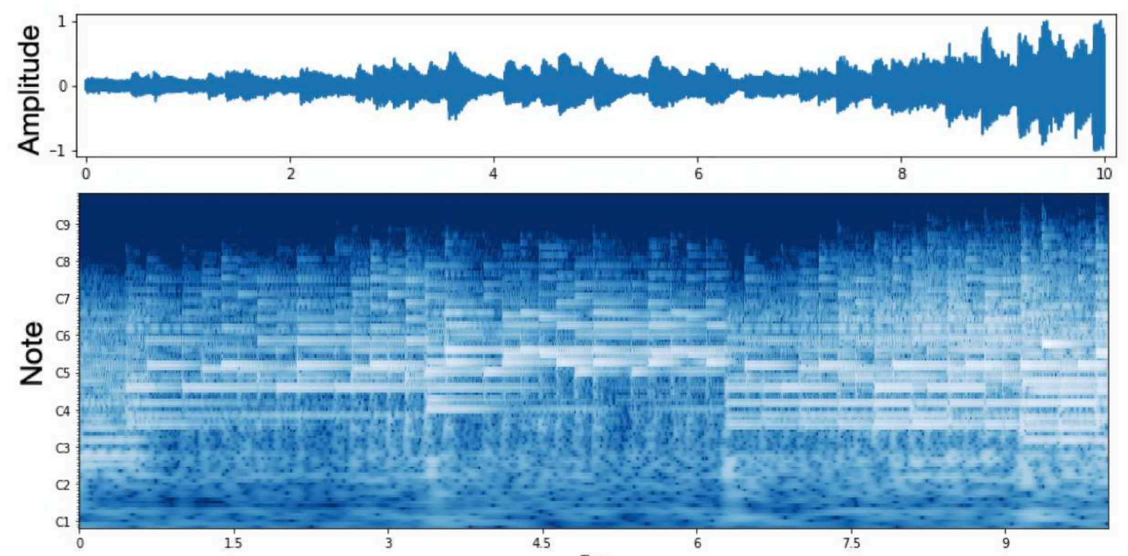
The goal of the Neuro-Ear is to design the best machine learning algorithm for the problem of classification of musical instruments. Many approaches to this problem exist today, for instance, in this paper [1] Attention was used and in this one [2] LSTM-based approach is described.

Like [3] we chose the most straightforward approach – a convolutional neural network. At **www.neuro-ear.appspot.com** the best model that has been developed so far is demonstrated. You can upload your own files and see the results. Your feedback will be highly appreciated

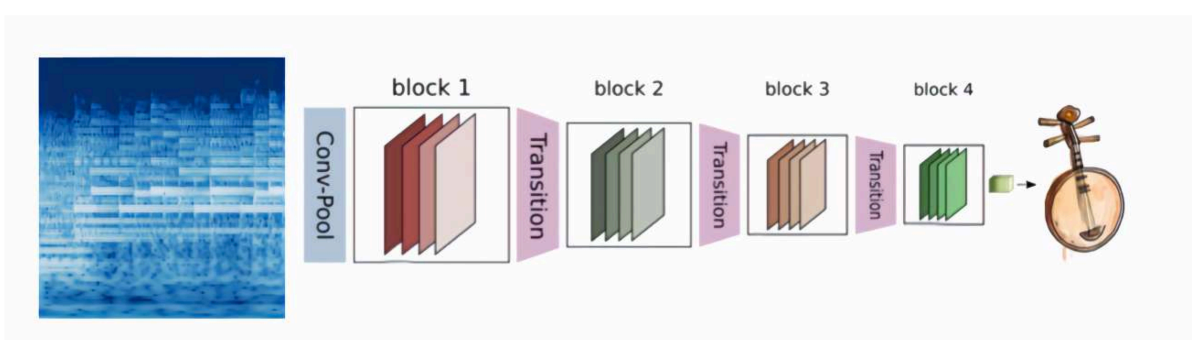
Source: **www.github.com/alimbfromlimb/neuro-ear_public**

Pipeline

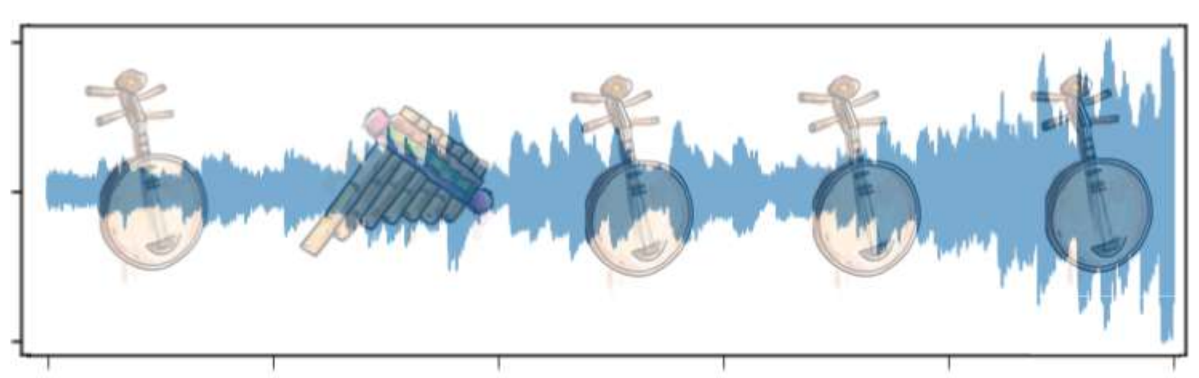
- A given .wav file is first converted to a chromargam using the Constant-Q Transform, which is a variation of the Fourier Transformation



- Then each one-second piece of the chromagram goes through a deep convolutional neural network, which returns the probability of each instrument to sound on this one-second piece. You can find the model diagram **here**. We also tried to substitute plain (conv $->$ batch norm $->$ ReLU $->$ conv $->$ batch norm $->$ ReLU) blocks with the corresponding ResBlocks. However, this did not enhance the quality of predictions at all. We believe it was due to the fact that the model we used was simple enough and there would have been no use of residual connections



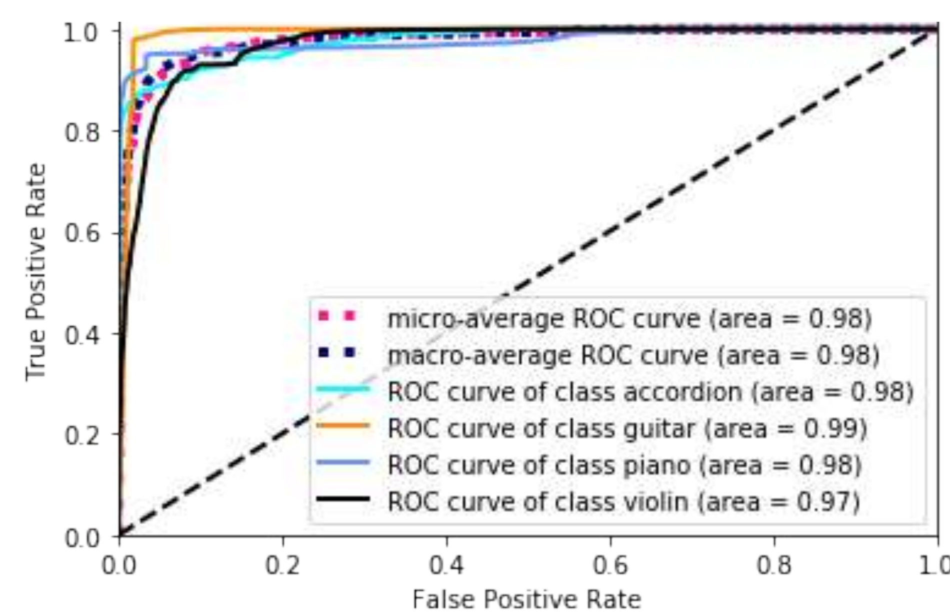
- Finally, you can see the probability distribution for each instrument (currently an accordion, guitar, piano and violin are available) over time



Dataset

The model was trained on a dataset consisting of 124 tracks (studio records, amateurs playing home and street musicians) found on the Internet. You can find a full list of the videos used for training and testing **here**

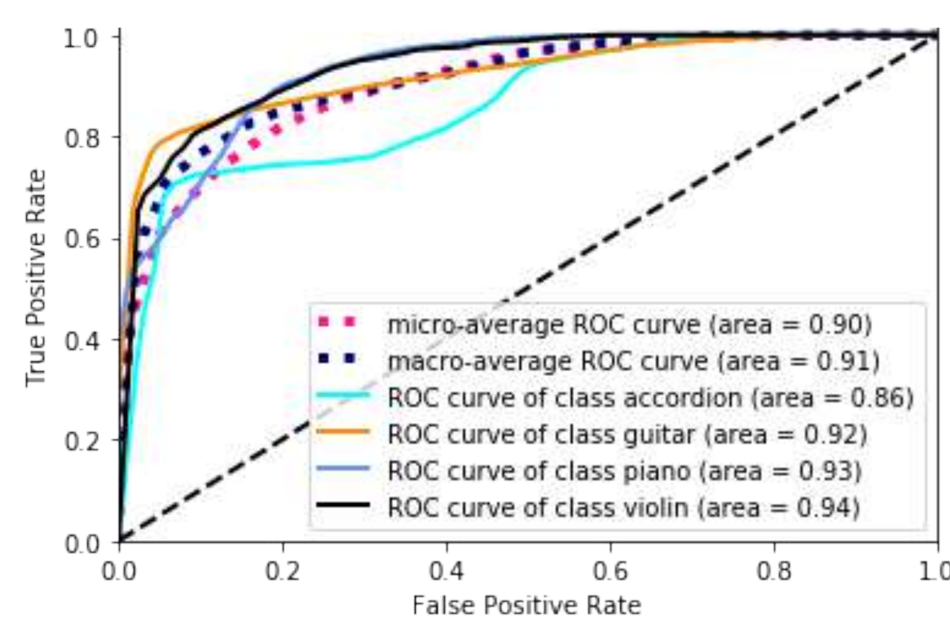
Results. Good Quality Audios



The model performed fine when tested on the audios of a relatively good quality (the dataset of 24 well-recorded audios). The BCE loss (with logits) and the Adam optimizer were used for training. Here are **the metrics:**

Instrument	F1 score	Precision	Recall
macro	88%	89%	88%
micro	88%	88%	88%
weighted	88%	89%	88%
accordion	84%	79%	90%
guitar	94%	89%	89%
piano	92%	98%	87%
violin	82%	88%	76%

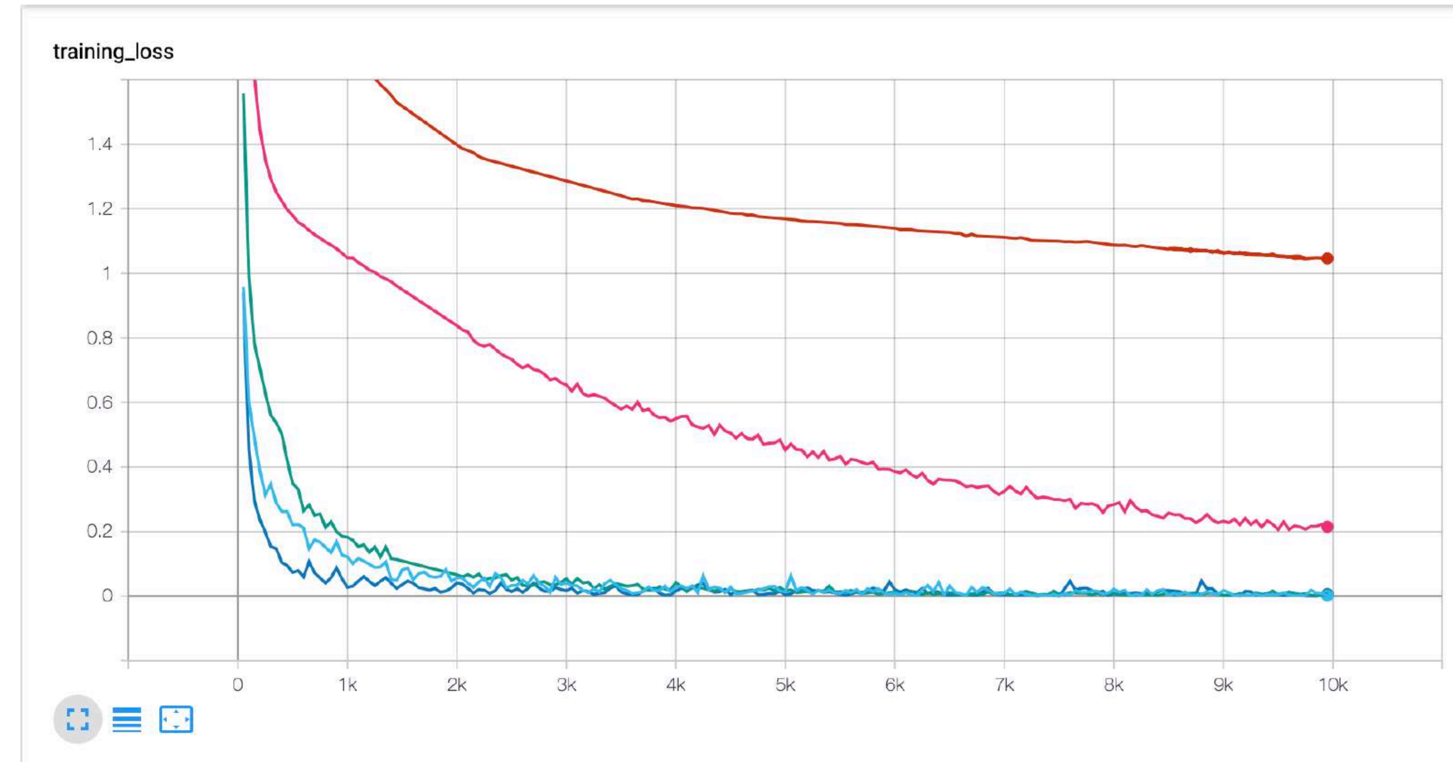
Results. Noisy Audios



When tested on noisy audios the model performed significantly worse. We are planning to perform some data augmentation to enhance the results

Testing Various Optimizers

Different gradient descent optimization techniques were compared during the training process. Much has been said about them in lectures on Optimization Methods. So why not to try them all and see which one is really the best?



Here they are, from top to bottom: SGD, SGD with Nesterov momentum (0.9), Adagrad, Adadelta and Adam (with default torch.optim parameters)

Conclusion

A neural net capable of accurate musical instruments detection on well recorded audios was written. Data augmentation is to be performed in the future to produce robust results on noisy inputs. It was shown by example that Nesterov Momentum [4] really helps to accelerate gradient descent and that Adam is one of the best modern methods of accelerating GD

Acknowledgements

This material is based upon work supervised by my seminarian Daniil Merkulov

References

[1] Siddharth Gururani, Mohit Sharma, Alexander Lerch. *An Attention Mechanism for Musical Instrument Recognition*. Center for Music Technology, Georgia Institute of Technology, USA.

[2] M. Yun *Deep Learning for Musical Instrument Recognition*

[3] G. Jawaherlalnehru, S. Jothilakshmi. *Music Instrument Recognition from Spectrogram Images Using Convolution Neural Network* ISSN: 2278-3075, Volume-8 Issue-9, July 2019

[4] Yu. E. Nesterov. *A method of solving a convex programming problem with convergence rate* $O\left(\frac{1}{k^2}\right)$. (Russian) Dokl. Akad. Nauk SSSR, 269:3 (1983), 543–547.