Attention Augmented CNNs for Musical Instrument Identification

Andrew Wise, Anthony S. Maida, Ashok Kumar
Center for Advanced Computer Studies
University of Louisiana at Lafayette
Lafayette, Louisiana 70504
Email: andrew.wise1@louisiana.edu

***Abstract*—We study the effectiveness of attention augmented convolutional neural networks for musical instrument identification in audio, which is an unsolved problem. Attention augmentation has not previously been applied to this task. The proposed architecture augments the final convolution modules from a baseline convolutional template with attention mechanisms. The network contains five total convolutional modules followed by five dense layers. The final layer uses softmax output to categorize 19 different orchestral instruments. Attention is introduced to enhance the network's ability to extract the casual structure underlying the formation of the spectrograms. We manipulate the ratio of attention augmentation to convolution in order to assess the efficacy of adding attention in this particular task. Input to the network is a 2D sound spectrogram of a 1s duration audio file taken from the London Philharmonic Orchestra and the University of Iowa Musical Instrument datasets. Experiments use two different spectrogram types, CQT and STFT, to assess their relative merits. Results show that the networks augmented with 25% of their filters for attention are able to outperform their only-convolutional counterparts and achieve 95.09% and 92.40% overall accuracy for STFT and CQT input spectrograms, respectively. The convolution only models achieve 84.94% and 91.43% accuracy, respectively.**

I. Introduction

Recently, Convolutional Neural Networks (CNNs) have been applied to the task of musical instrument identification [1]–[4]. When an audio file contains multiple instruments, the mixture creates a new, unique timbre that can cause individual instruments to sound completely different from playing solo. This creates two distinct research problems for identifying instruments within a piece: classifying an isolated instrument, and detecting all or only the predominant instrument in mixtures. An ideal solution will satisfy each of these problems, and can be used in applications like Automatic Music Transcription [5] and dataset purging [6].

When processing an audio file for instrument recognition, it is typically converted (preprocessed) to a spectrogram which takes the form of an image representation, and the spectrogram is then used as input to the network. Because classification is then applied to the spectrogram image, CNNs are an obvious choice to perform this task.

It is believed that CNNs are successful because they recover the casual model that maps 3D objects in the visual world to an image projection [7], [8]. However, recovering a casual model of sound generation in a musical piece from a visual spectrogram presents more complications. One assumption of CNNs is that the statistical properties across an input image are largely uniform and hierarchically local. This may not be true for the casual structure represented within a spectrogram. Moving a filter across the y-axis visits image regions whose properties reflect different aspects of the causal model relating to changes in the fundamental frequency and harmonics. These changes appear to deviate from the statistical assumptions of using a kernel in the first place. Additionally, the spectral properties of sound are non-local, but rather move according to a constant relationship. CNN layers, however, impose locality due to a limited receptive field that makes them unable to grasp these global contexts.

Although in principle, a CNN could learn to identify musical instruments from visual correlations in a spectrogram with their associated instrument, completely ignoring any casual properties, we hypothesize recovering the casual model in the internal representations of the network may improve performance. However, this requires that more representational power be provided to the network. Attention augmented CNNs [9] are believed to have more representational capacity, particularly with respect to non-local relationships in an image. Consequently, we hypothesize a CNN augmented with attention may offer improved performance over existing CNN models. Thus, we propose to train attention augmented CNNs to perform musical instrument identification and evaluate them on isolated instruments.

Previous musical instrument identification methods have been trending towards the use of deep learning methods, specifically CNNs. Attention mechanisms have been shown to outperform recurrence when tracking long-range dependencies in natural language processing tasks. Despite these dependencies occurring naturally in music, attention has yet to be applied to music-related tasks with the only known occurrence being the Google Brain team's application to music generation [10]. Since then, [9] demonstrated that CNNs augmented with attention outperformed non-augmented variants for image classification tasks. Given the recent success in using CNNs for musical instrument identification, a natural next step is to test the impact of an attention augmentation in this context but it has yet to be done. We propose just this and report on how attention impacts the task in a single-source setting by training and comparing multiple networks with varying levels of attention. Results show augmenting CNNs

with attention mechanisms aids in the process of identifying musical instruments with potential to extend to other music related tasks, laying the groundwork for future research.

The rest of this paper is organized as follows. Section II briefly discusses recent methods of musical instrument identification that use CNNs to familiarize the reader with a baseline performance level and other background information. Section III describes the created model in detail. In Section IV, we describe the experiments run on the model and report the results. Finally, Section V concludes the paper.

II. Background and Related Works

Convolutional Neural Networks (CNN) are among the most recent deep learning methods applied to the musical instrument identification task. For instance, [3] created a Deep CNN (DCNN) to determine an arbitrary number of predominant instruments from a one-second mixture that outperformed previous methods proposed by [11] and [12]. Another DCNN proposed in [4] utilized auxiliary classification based on onset groups and instrument families to achieve micro and macro F1 scores of 0.685 and 0.597 respectively, an increase of 10.7% and 16.4% over those achieved in [3].

In contrast to our proposal, some methods circumvent the limitations of processing spectrograms with CNNs by changing the input. Park et al. combined spectrograms with multiresolution recurrence plots (MRP) as input for a CNN to classify 20 instruments [1]. The MRP addition preserved the phase information that is typically lost by a spectrogram, and helped them achieve an error rate of 6.35%. Li et al. created an end-to-end DCNN utilizing raw audio as input to classify 11 instruments [2]. Their end-to-end approach lets the network run feature extraction and determine which ones are helpful for classification. This approach achieved an accuracy of 82.74% beating their baseline methods. Most recently, [13] augmented their dataset by applying a variety of mixing methods to make it more polyphonic and increase its size. They were able to achieve just above 80% label ranking precision on the IRMAS dataset, a 2% increase over its predecessors.

Recently, attention mechanisms were introduced [14] and then popularized [15] to capture long-range dependencies in sequence modeling tasks like natural language processing. In [15], they achieved state-of-the-art results in machine translation without the need for recurrence. They now perform better than recurrence and have replaced such approaches [16]. Most recently, attention augmented CNNs were introduced in [9] to capture and represent long-term dependencies in images. To current knowledge, only a Google Brain team has applied attention to a music task where the authors created a music transformer with attention to generate symbolic music sequences that contained long-term structure [10]. Accordingly, attention has yet to be applied to musical instrument identification, making our work novel.

The Short Time Fourier Transform (STFT) is a common audio processing technique which converts a time-domain signal into the corresponding time-frequency distribution [17]. It uses a constant separation between components, providing a fixed time-frequency resolution for all types of signals [18], [19]. However, the fundamental frequencies of music notes are non-constantly spaced, making some researchers believe this causes inefficient results in a music setting. The Constant-Q-Transform (CQT) extends STFT by ensuring the center frequencies of the bins are geometrically spaced, creating a constant ratio of frequency to resolution and ensuring the relative spacing between harmonics remains constant [20]. Depending on the task, one or the other method can prove better. In practice, few works directly compare the two methods under a specific task. [21] compared them in a note onset detection task, though CQT performed better, the results were not significant enough to qualify it as superior. Since there is no consensus on which spectrogram type provides better input for music tasks, we compare both methods in all of our simulations.

III. Proposed Model Architecture

The proposed architecture is shown in Fig. 1. We augment a baseline convolutional architecture with attention mechanisms. The network contains five 2D convolution/attention blocks, followed by flattening, and then five dense layers using the ReLu nonlinearity, except for the last dense layer, which uses softmax for non-exclusively classifying 19 musical instruments. A thorough search of different architectures found that five convolutional blocks performed the best. To maintain reasonable memory requirements [9], the attention augmentation is limited to convolution blocks 4 and 5, which have smaller image representations. The methods in [9] are followed to implement the attention augmentation. In the figure, Convolution Block Four is expanded to reveal parallel convolution and attention pathways with the operations labeled 'Conv Out' and '2D Attn' respectively. The blocks also use batch normalization and max pooling. Except when 1x1 kernels are used to interface with 2D attention, all convolution filters use 5x5 kernels. The number of filters starts with 32 in the first convolution block and is doubled for each thereafter, while 2x2 max pooling is simultaneously applied. The first dense layer has 1024 units and the subsequent layers are consecutively reduced by two, until the last layer is reached which uses 19 output units.

Input to the network is either a CQT or STFT 2D spectrogram whose dimension is (87, 252, 1) or (87, 1025, 1) respectively. In addition to varying the type of spectrogram input, the attention-convolution ratio is varied in blocks 4 and 5. This is done by varying the percent of filters allocated to attention versus those allocated to pure convolution. Accordingly, for a convolution operation augmented with 25% attention and having 100 total filters, there are 25 filters for attention and 75 for convolution. All augmented convolutions in a given network use the same attention percentage, and we vary the amount over multiple networks to assess the impact of the attention augmentation on the task. The different attention percentages cause the number of trainable parameters to vary from 55 million to 65 million.

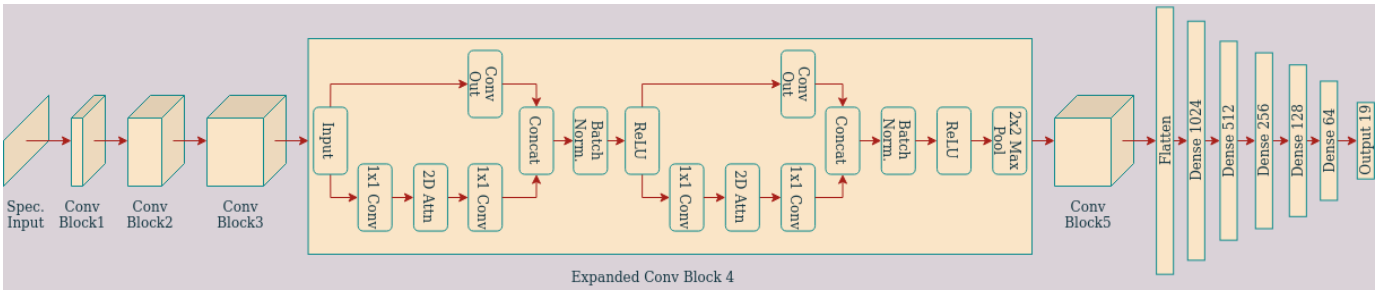

Fig. 1. Proposed Attention Augmented Deep CNN. The fourth convolution block is expanded, showing split paths for the convolution and attention augmentation.

Source code for the proposed model is available on github at https://github.com/Adurnis/Attention-Augmented-CNN-For-INS-ID.

IV. Experiments and Methods

A. Datasets and Preprocessing

The audio datasets were obtained from the London Philharmonic Orchestra Dataset [22] and the University of Iowa Musical Instrument Samples [23]. The audio files cover a variety of playing techniques from 19 orchestral instruments. Their duration ranges from 0.5 to 5+ seconds. As the goal of this work is to study and report the efficacy of attention on identifying musical instruments with CNNs, we chose the datasets from [22] and [23]. The datasets contain samples for multiple different percussion instruments, but there are very few for each individual one. Additionally, they are not consistent in sound/type to where they can properly be grouped and contain sufficient samples for classification. Accordingly, they were excluded from the dataset.

Following the window-size results of [3], we use 1s duration audio samples as input to the studied networks. Consequently, shorter files are discarded. Longer files are partitioned into files of 1s duration. CQT and STFT representations were created for the audio files using the Python Librosa Library [24]. The training/test split was randomly sampled at a ratio of 90/10. Table I shows the sample distributions across the 19 instruments for both the training and testing data. The training data is further split 90/10 via random sampling for training/validation, though the distribution is not included for brevity. These splits ensure the networks are tested on data that has not previously been seen.

TABLE I
Test Data Instrument Counts

Instrument	Training	Testing	Total
banjo	199	18	217
bass-clarinet	765	78	843
bassoon	970	118	1088
cello	1438	150	1588
clarinet	1510	163	1673
contra-bassoon	1443	132	1575
cor-anglais	1096	132	1228
double-bass	1388	173	1561
flute	1426	148	1574
french-horn	1166	147	1313
guitar	525	59	584
mandolin	188	25	213
oboe	614	80	694
saxophone	1086	128	1214
trombone	885	91	976
trumpet	1020	124	1144
tuba	520	60	580
viola	1251	112	1363
violin	1225	141	1366
Total	18715	2079	20794

B. Training and other Methods

All models trained are instances of the template shown in Fig. 1. The attention-to-convolution ratio for the filters was manipulated, yielding five models having attention percentages of: 0, 25, 50, 75, and 100 percent. All five were tested with both CQT and STFT input, yielding ten total models.

The models were implemented in Keras provided by Tensorflow 2.3 and run on a Linux server equipped with a 24Gb Titan RTX graphics card. Weights and biases were initialized with the Keras "random normal" initializer using default settings. A training epoch consisted of all the training data with a minibatch size of 16. Training used the Keras Adagrad optimizer with default settings. A cross-entropy loss function is used. Training ran for 400 epochs with early stopping when validation did not improve after 30 epochs. In practice, only CQT networks surpassed 200 epochs during training, and those that did showed very little improvement beyond 250 epochs.

Figure 2 shows a sample of how the training and validation accuracies change as training progresses on STFT and CQT networks with 25% attention. Both networks in this instance stopped early after the validation loss did not increase for 30 epochs; the STFT network ran for 72 epochs while the CQT ran for 257. The figure shows that, for both STFT and CQT input types, generalization from the training to the validation sets was robust. This is true at all percentages of attention tested.

C. Performance Measures

Five performance measures were calculated for each simulation. These were accuracy, precision (P_{macro}), recall (R_{macro}), and F1 (both micro and macro). Since instrument samples were not uniformly represented in the dataset, micro and

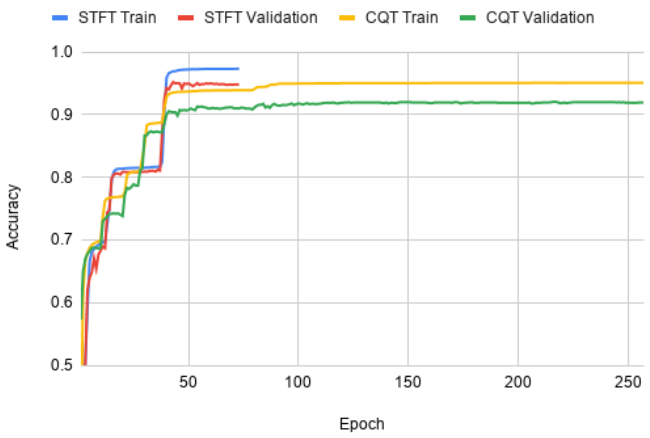

Fig. 2. Training and validation accuracy vs epoch for selected STFT and CQT models with 25% attention.

TABLE II
RESULTS OVER VARYING ATTENTION AMOUNTS FOR DIFFERENT INPUT SPECTROGRAMS.

Attention Ratio	Accuracy	P_{macro}	R_{macro}	$F1_{micro}$	$F1_{macro}$
STFT					
0%	84.94%	0.7353	0.8504	0.8494	0.7887
25%	**95.09%**	**0.8499**	**0.8675**	**0.9509**	**0.8586**
50%	89.71%	0.7535	0.8220	0.8971	0.7863
75%	87.30%	0.6914	0.7583	0.8730	0.7233
100%	87.69%	0.7063	0.7826	0.8769	0.7425
CQT					
0%	91.43%	0.7863	0.8388	0.9143	0.8117
25%	**92.40%**	**0.8042**	**0.8611**	**0.9240**	**0.8317**
50%	85.26%	0.6483	0.7411	0.8526	0.6916
75%	81.96%	0.6035	0.6996	0.8196	0.6480
100%	81.60%	0.6046	0.6885	0.8160	0.6438

macro measures were used for F1. Micro averages are globally computed to give more weight to more frequently sampled classes. Macro averages are calculated on instrument class averages, thereby removing the frequency bias.

D. Results

The results can be found in Table II. Each network variation was tested three times, with values reported as averages across the runs. We find that networks augmented with 25% attention outperform all other amounts of attention for both input representations. As the attention level is further increased, the overall performance of the network diminishes, however, the results still appear competitive to pure convolution at all but 75% and 100% attention for CQT input. The attention augmentation seems to have improved the performance of the STFT network significantly more than the CQT variant. This is demonstrated by the significant jump in performance seen between 0 and 25% attention for the STFT network while the CQT network only showed a minor boost. Specifically, STFT with 25% attention saw increases of 15.59%, 2.01%, 11.95%, and 8.86% for P_{macro}, R_{macro}, $F1_{micro}$, and $F1_{macro}$ respectively over the fully convolutional architecture; while the CQT with 25% attention saw increases of 2.28%, 2.66%, 1.06%, and 1.71%. Additionally, at all amounts of attention the STFT network outperformed its fully convolutional version, while the CQT network saw a performance degradation at any attention level above 25%. We attribute this to the CNN having the learning capacity to do a good job of capturing most of the casual properties contained in the underlying spectrogram. The 25% attention augmentation captures some of the remaining properties with diminishing returns for larger proportions of attention. Though full attention outperformed full convolution on STFT input, the best result coming from a mixture shows attention cannot fully replace convolution. Overall, STFT input representations consistently outperformed CQT on all metrics at multiple attention levels, but the difference is not significant enough to deem it superior. Interestingly, the only variant where CQT showed better performance is the fully convolutional version. This is being attributed to the CNN being able to better capture and classify the features by itself on the CQT spectrogram than the STFT version. When attention augmentation is used, it is believed the long-distance relationships in the harmonics are better captured when they are not consistently spaced. Further investigations are required to confirm this hypothesis, but current results suggest STFT spectrograms are better than CQT ones as input for convolutional networks augmented with attention mechanisms. However, this may prove task-dependent.

V. CONCLUSION

In this work, we proposed and studied the application of attention augmented CNNs to a musical instrument identification task. The networks were trained and evaluated on 1-second sound clips of isolated notes and phrases from 19 unique orchestral instruments from [22] and [23]. Our experimental results found networks augmented with 25% attention outperformed their fully convolutional counterparts, and this may open a new frontier in musical instrument identification. This shows that for our task, attention mechanisms can augment convolutional mechanisms but they cannot fully replace them. Specifically, the STFT network saw 11.95% and 8.86% increases in the $F1_{micro}$ and $F1_{macro}$ values while the CQT network achieved 1.06% and 2.46%, respectively. Increasing the attention percentage further proved competitive, but overall the performance degraded as attention was increased past 25%. A direct comparison between STFT and CQT input spectrograms was run. We found that STFT outperformed CQT across multiple levels of attention in all studied cases. Through this work, we have shown the potential for using attention augmented CNNs in music-related tasks. Given our results, we have laid the groundwork for future research to explore why attention differentially improves STFT more than CQT, what attention-based learning captures and extend the model to include testing on the IRMAS dataset.

REFERENCES

[1] T. Park and T. Lee, "Musical instrument sound classification with deep convolutional neural network using feature fusion approach," *CoRR*, vol. abs/1512.07370, 2015. [Online]. Available: http://arxiv.org/abs/1512.07370

[2] P. Li, J. Qian, and T. Wang, "Automatic instrument recognition in polyphonic music using convolutional neural networks," *CoRR*, vol. abs/1511.05520, 2015. [Online]. Available: http://arxiv.org/abs/1511.05520

[3] Y. Han, J. Kim, and K. Lee, "Deep convolutional neural networks for predominant instrument recognition in polyphonic music," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 1, pp. 208–221, Jan 2017.

[4] D. Yu, H. Duan, J. Fang, and B. Zeng, "Predominant instrument recognition based on deep neural network with auxiliary classification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 852–861, 2020.

[5] S. Sigtia, E. Benetos, N. Boulanger-Lewandowski, T. Weyde, A. S. d'Avila Garcez, and S. Dixon, "A hybrid recurrent neural network for music transcription," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 04 2015, pp. 2061–2065.

[6] A. Livshin and X. Rodet, "Purging musical instrument sample databases using automatic musical instrument recognition methods," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 5, pp. 1046–1051, 2009.

[7] S. Huang, Y. Chen, T. Yuan, S. Qi, Y. Zhu, and S.-C. Zhu, "Perspectivenet: 3d object detection from a single rgb image via perspective points," 2019.

[8] A. Rabinovich, E. Wiewiora, A. Vedaldi, S. Belongie, and C. Galleguillos, "Objects in context," in *In ICCV*, 2007.

[9] I. Bello, B. Zoph, Q. Le, A. Vaswani, and J. Shlens, "Attention augmented convolutional networks," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 3285–3294.

[10] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, I. Simon, C. Hawthorne, N. Shazeer, A. M. Dai, M. D. Hoffman, M. Dinculescu, and D. Eck, "Music transformer: Generating music with long-term structure," in *International Conference on Learning Representations*, 2018.

[11] F. Fuhrmann and P. Herrera, "Polyphonic instrument recognition for exploring semantic similarities in music," *13th International Conference on Digital Audio Effects, DAFx 2010 Proceedings*, pp. 1–8, 10 0002.

[12] J. Bosch, J. Janer, F. Fuhrmann, and P. Herrera, "A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals," in *13th International Society for Music Information Retrieval Conference (ISMIR 2012)*, Porto, Portugal, 08/10/2012 2012, pp. 559–564. [Online]. Available: http://mtg.upf.edu/system/files/publications/Bosch-ISMIR2012.pdf

[13] A. Kratimenos, K. Avramidis, C. Garoufis, A. Zlatintsi, and P. Maragos, "Augmentation methods on monophonic audio for instrument classification in polyphonic music," in *2020 28th European Signal Processing Conference (EUSIPCO)*, 2021, pp. 156–160.

[14] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1409.0473

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6000–6010.

[16] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805

[17] J. O. Smith, *Spectral Audio Signal Processing*. http://ccrma.stanford.edu/ jos/sasp/-ccrma.stanford.edu/- jos/-sasp/, accessed 30/08/2019, online book, 2011 edition.

[18] J. C. Brown, "Calculation of a constant q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991. [Online]. Available: https://doi.org/10.1121/1.400476

[19] S. Nisar, O. Khan, and M. Tariq, "An efficient adaptive window size selection method for improving spectrogram visualization," *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–13, 01 2016.

[20] C. Schörkhuber and A. Klapuri, "Constant-q transform toolbox for music processing," *Proc. 7th Sound and Music Computing Conf.*, 01 2010.

[21] A. Lacoste and D. Eck, "A supervised classification algorithm for note onset detection," *EURASIP J. Adv. Signal Process*, vol. 2007, no. 1, pp. 153–153, 01 2007. [Online]. Available: https://doi.org/10.1155/2007/43745

[22] L. P. Orchestra, "Sound samples."

[23] Matt Hallaron et al., "University of iowa musical instrument samples," 1997, accessed: 2020-05-07. [Online]. Available: http://theremin.music.uiowa.edu/MIS.html

[24] Brian McFee, Colin Raffel, Dawen Liang, Daniel P.W. Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto, "librosa: Audio and Music Signal Analysis in Python," in *Proceedings of the 14th Python in Science Conference*, Kathryn Huff and James Bergstra, Eds., 2015, pp. 18 – 24.