Microphone-based Data Augmentation for Automatic Recognition of Instrumental Playing Techniques

Nicolas Brochec, Tsubasa Tanaka, Will Howie

Microphone-based Data Augmentation for Automatic Recognition of Instrumental Playing Techniques

Nicolas Brochec
Tokyo University of the Arts
nicolas.brochec@pm.me

Tsubasa Tanaka
Tokyo University of the Arts
tanaka.tsubasa@ms.geidai.ac.jp

Will Howie
Japan Society for the Promotion of Science International Research Fellow,
Tokyo University of the Arts
wghowie@gmail.com

ABSTRACT

Within existing research on the automatic classification of musical instrument playing techniques, few available datasets include enough playing techniques to cover the full range of a given musical instrument's expressive ability. However, creating a new large dataset requires recording many samples for many performance techniques, which is costly and time-consuming. Therefore, in this study, we attempt to augment data by increasing the number of recording microphones without increasing the recording duration and verify the effectiveness of this data augmentation method. As a result of recording flute playing techniques using multiple microphones, the accuracy and macro F1-Score of a convolutional neural network-based classifier improved when using a combination of the five most close-to-source microphones. The classifier's performance further improved when data were combined with a data augmentation method based on pitch shifting.

1. INTRODUCTION

Throughout the history of contemporary art music, composers have diversified Instrumental Playing Techniques (IPTs) to foster musical innovation. In contemporary mixed music, real-time digital audio effects are often applied to the live performance of acoustic instruments. The performer, a foot pedal, or a computer operator usually triggers these effects. However, automated triggering could reduce the burden on the operator and expand creative possibilities. This burden originated the development of score follower systems such as Antescofo [1]. Antescofo tracks the performer's temporal position on the score and automatically switches effects, reducing the burden on the operator. However, using Antescofo requires the score input information for each piece of music which adds an extra burden. Moreover, Antescofo's machine listening is based on pitch and rhythm detection, which does not allow for the recognition of IPTs. This research aims to develop a system that automatically recognizes IPT and switches audio effects according to it, thereby reducing the burden on the operator and the cost of music score input.

Several existing studies demonstrate that the development of automatic recognition of IPT systems suffers from a lack of large common sound banks [2, 3], with proposed systems often relying on commercial sound banks. Commercial sound banks are primarily recorded for industrial music production that does not require advanced contemporary instrumental techniques. These banks typically only include one sample per pitch per instrumental playing technique, severely limiting the available audio samples to train a classification algorithm.

One way to improve classification accuracy is to increase the number of audio samples by using data augmentation methods. Existing research [3, 4, 5, 6] proposes generating new audio samples by applying various digital audio transformations to the original data such as pitch shifting. However, these experiments have shown that the accuracy is still too low to put automatic recognition of IPT into practice.

Another possible data augmentation method is to record audio samples at different distances from the sound source. It would augment the number of samples without increasing the recording duration. In this study, we propose a novel method of data augmentation using multiple microphone positions, and evaluate its efficacy for automatic recognition of IPTs.

2. PROPOSED METHOD

To evaluate the efficacy of the microphone-based data augmentation method, we first simultaneously captured audio samples of flute playing techniques from different source-to-microphone distances. We then made several datasets, trained a neural network-based classifier, specifically Convolutional Neural Network (CNN), and tested it on a dataset originating from a different sound bank. Finally, we assessed the performance of the classifier with different metrics. We selected the flute as our instrument of focus because the first author of this study is familiar with this instrument. Furthermore, this study continues our previous study on the automatic recognition of flute IPTs [4].

2.1 Microphone Set

Diverse flute playing techniques were captured with seven microphones placed at different distances from the source, as described in Table 1. The microphones **A**, **B**, **C**, **D** were oriented toward the source, focusing on direct sound capture. Microphones **E** were oriented toward the ceiling to capture the room's natural reverberation. The "main

Microphone	Distance	Height	Nbr of Mics	Letter
DPA 4099	33 cm	/	1	**A**
DPA 4011	80 cm	/	1	**B**
DPA 4011	100 cm	/	1	**C**
DPA 4006	140 cm	179 cm	2	**D**
DPA 4006	458 cm	277 cm	2	**E**

Table 1. Microphones used to record the sound bank.

aeolian	flatterzunge	key click	multiphonics
ordinario	pizzicato	play and sing	staccato
tongue ram	trill	whistle tone	

Table 2. Flute playing techniques in 11 categories.

stereo" pair **D** microphones were 38 cm apart, while the microphones within the "ambient" pair **E** were 300 cm apart. Microphone positions **A**, **B**, **D**, and **E** are typical of contemporary studio and live recording/amplification microphone techniques. We placed microphone **C** at a distance of 1m from the source to the microphone because this distance is typically used for acoustic and electroacoustic measurements. The DPA 4099 is frequently used for live acoustic instrument amplification, while DPA 4011 and 4006 microphones were chosen for their flat frequency response, allowing for more accurate acoustic information capture. The height of **A**, **B**, and **C** is equal to the distance from the floor to the flute head hole.

2.2 Selection of Flute Playing Techniques

The flute can produce a chromatic scale from medium (C3≈261 Hz) to treble pitch (C6≈2093 Hz). The sound of the flute is produced by the friction of the air on the mouthpiece, while different changes in the velocity and characteristics of the blown air allow for playing diverse techniques [7, 8, 9]. In our previous study [4] we selected 19 different flute IPTs from the FullSOL library [10], and we showed that some IPTs are unsuitable for our investigation. Indeed, classifiers get confused when classifying the *harmonics*, *discolored fingering*, and *aeolian and ordinario* techniques because the *ordinario* technique is very close to them. We also concluded that similar techniques should be grouped together to reduce the number of classes and increase the number of samples per class. For instance, a minor second trill and a major second trill fall under the *trill* category. Playing a whistle tone with or without glissando is essentially performing a *whistle tone*. Playing and singing at the same pitch or not is performing a *play-and-sing* technique. As a result, we use in this study 11 flute IPTs (see Table 2).

All seven microphones simultaneously captured each flute playing technique played pitch by pitch chromatically within their respective register at *mezzoforte*. Some techniques have a small range of pitches (*aeolian*) or produce a short duration sound (*key click*, *staccato*, *pizzicato*, *tongue ram*). This can cause an imbalance in the training datasets because of the difference in sample duration for each class. We therefore recorded each short-duration playing technique twice more to increase available data. Our sound bank includes 2.85GB of audio files recorded at 96 kHz / 24-bit resolution [11].

2.3 Datasets

We designed two experiments to investigate which microphone combinations produce the best results for automatic recognition of flute playing techniques. For this purpose, we created multiple training datasets based on different microphone combinations and tested them on a separate test dataset (heterogeneous datasets).

2.3.1 Training Datasets

We first made a dataset for each monophonic recording (**A**, **B** and **C**), and for each stereo recording (**D** and **E**). We then combined these monophonic and stereophonic recordings to make new training datasets. First, we combined **A** and **B**, **B** and **C**, and **C** and **A**. Then, we combined **A**, **B** and **C**. After, we combined **A**, **B**, **C** and **D**, and **A**, **B**, **C**, and **E**. A final dataset was also created using all the datasets (**ABCDE**). We created a total of 12 different datasets. For stereo recordings **D** and **E** we use both channels as two different monophonic signals.

2.3.2 Test Dataset

For a comprehensive evaluation of the classifier, we tested it on a separate test dataset. The test dataset is made from the FullSOL sound bank [10], and includes flute playing techniques similar to those found within our datasets. To match the flute playing techniques from FullSOL with those from our training datasets, we deleted any unused techniques, and combined similar techniques into the same categories, as we did for our own datasets.

2.3.3 Validation Dataset

Our experiments used a validation dataset to measure the accuracy at each step of the training. This dataset is made from 20% of the test dataset. The selection of samples is stratified, which means that the proportion of samples in each class is proportional to the original test dataset.

2.4 Audio File Pre-processing

To prepare our data, we followed an existing methodology [4]. We downsampled the audio file sample rate to 24 kHz, as 12 kHz (the Nyquist frequency) is sufficient to cover most flute harmonics. We removed any silence within the audio file, as it is irrelevant. We edited the audio file into 15-frame-long sequences (≈ 320ms). We analyzed the sequences with a Log-Mel-Spectrogram (LMS) analysis. We computed the LMS on 128 bins, and the FFT window is fixed at 2048 samples with a hop size of 512 samples (≈ 21.3ms). The minimum frequency is set to 150 Hz because bass frequencies are irrelevant for the flute. Each sound file was edited into a maximum number of data samples according to the number of frames used. When the length of a given chunk was less than 15 frames, we padded the audio sample with zeros. We then normalized the data.

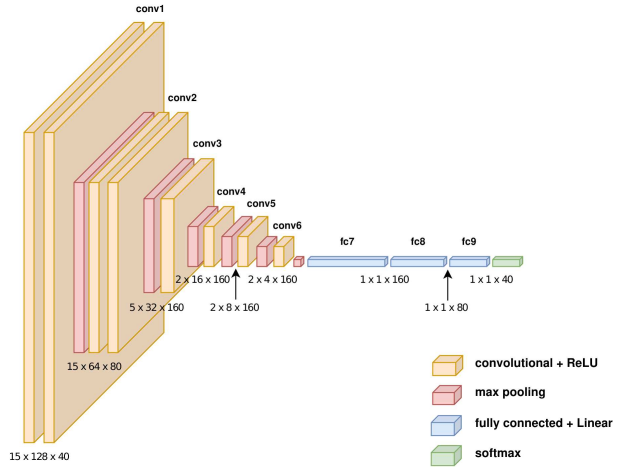

Figure 1. Schematic representation of the proposed architecture.

2.5 Methods of Classification

2.5.1 Classifier Architecture

To perform our experiments, we chose to implement a deep Convolutional Neural Network (CNN) architecture, which has been shown in several previous studies to have high efficiency for instrument-related audio classification tasks [3, 5, 6, 12, 13, 14, 15, 16]. In the domain of automatic recognition of IPT, existing research proposes design strategies to create CNN architectures [3]. We propose augmenting the capacity of these CNN architectures by adding several layers and by selecting the hyper-parameters that favored a rise in accuracy. A schematic representation of our architecture can be seen in Figure 1. We found that a kernel size of *2x3* works better for conv1, conv2, and conv3 modules than square-like sizes. Conv4, conv5, and conv6 use a *2x2* kernel size. We use batch normalization [17] and dropout layers [18] after each convolutional layer to speed up the training process and reduce overfitting. After the sixth convolutional layer, we connect three fully connected layers (fc7, fc8, and fc9) with a linear activation function.

2.5.2 Training

The neural network weights are initialized using Xavier Normal Initialization [19]. Training minimizes cross-entropy loss through mini-batch gradient descent with ADAM optimization. The learning rate begins at $lr = 0.001$, decaying exponentially, with validation accuracy monitored on the last 10 epochs. Training lasts for a maximum of 100 epochs on an A100 GPU machine, halting if validation accuracy has not improved in the last 20 epochs.

3. EXPERIMENTS

We performed two experiments to test whether microphone-based data augmentation can be effective. We first trained our classifier on the original data with different combinations of microphones. Then, in a second experiment, we augmented the number data by applying pitch shifting to the original audio data.

3.1 Transformation-based Data Augmentation

For audio classification tasks, digital audio transformations are often applied to the original data to augment the number of samples within the training datasets [20]. In the case of IPT classification, audio transformations are commonly utilized to represent the real-world environment(s) of musical performances [3, 4, 5, 6]. The aim of adding transformed audio data samples to our original data is to verify whether combining microphone-based data augmentation with transformation-based data augmentation would increase the accuracy. We only chose pitch shifting for that purpose because it increases accuracy [20]. The tuning of each of our audio samples is randomly modified in a range of 200 Hz around the tuning frequency (440 Hz).

3.2 Metrics

We evaluated our classifier by conducting two types of measurements. Firstly, we measured the accuracy of the model state achieving the highest accuracy on the validation dataset. Due to variations in sample durations per class, our datasets are unbalanced. To address this, we opted for the macro F1-Score, providing a more representative measure of classifier performance considering no difference between highly and poorly populated classes [21]. We repeated the training and testing five times per dataset and calculated the average accuracy and macro F1-Score on the five tests.

We computed confusion matrices to understand how well our classifier identified the playing techniques. For each of the five tests, a confusion matrix is generated. The final confusion matrix is the average of five individual matrices from the tests. We provided the average confusion matrices for the best-performing models based on average accuracy scores.

4. RESULTS

We performed two experiments, with and without the addition of pitch-shifted audio samples.

4.1 Accuracy and Macro F1 Score

Dataset	Original		Pitch-shifted	
	Accuracy	Macro F1	Accuracy	Macro F1
A	84.83	66.38	84.78	69.12
B	86.00	65.28	85.42	64.02
C	80.96	60.06	84.16	65.62
D	81.40	59.98	85.45	63.79
E	71.74	51.74	82.53	54.99
AB	86.19	68.05	89.18	73.18
BC	86.66	66.92	88.71	71.55
CA	85.78	69.19	87.07	71.03
ABC	**87.86**	**70.76**	89.72	72.82
ABCD	84.47	68.95	**91.32**	**75.56**
ABCE	85.40	68.89	90.35	72.36
ABCDE	84.91	67.98	90.84	74.05

Table 3. Comparison of accuracy and F1-Score results with and without adding pitch-shifted audio samples. Results averaged on five tests (%).

Figure 2. Original audio samples. **ABC** microphones. Confusion matrix averaged on five tests (%).

Figure 3. Pitch-shifted audio samples added to original audio samples. **ABCD** microphones. Confusion matrix averaged on five tests (%).

With original audio samples, we measured the highest accuracy and macro F1-Score of 87.86% and 70.76% with the combination of **ABC** microphones. When adding the pitch-shifted audio samples, we measured accuracy and macro F1-Score of 91.32% and 75.56% with the combination of **ABCD** microphones.

4.2 Confusion Matrices

The confusion matrix Figure 2 shows that the majority of playing techniques are well-identified when the classifier is trained with original audio samples. However, the *key click* and the *tongue ram* techniques are misidentified. The confusion matrix Figure 3 shows that the majority of playing techniques are better identified when the classifier is trained with the addition of pitch shifted audio samples.

5. DISCUSSION

We found that the combination of **ABC** microphones gave the best accuracy (87.86%) and macro F1-Score (70.76%) when using original audio data samples. The ambient sound microphones **E**, placed farthest from the source and oriented towards the ceiling, yielded the lowest accuracy and macro F1-Score. This indicates that for the classifier, the acoustic information of the room's natural reverberation is less important than the direct signal. Adding more microphones increased the accuracy by 1.86% and the macro F1-Score by 5.48% compared to the best-performing single microphone **B**. However, using more than three microphones caused drops in accuracy and F1-Score.

Adding pitch-shifted audio samples to the original samples improved accuracy and F1-Score. The best scores were achieved with microphone combination **ABCD**, with an accuracy of 91.32% and macro F1-Score of 75.56%. The lowest accuracy and macro F1-Score were measured with the microphones **E**. Adding microphones improved accuracy by 5.9% and macro F1-Score by 11.54% compared to microphone **B**. Using more than five microphones decreased accuracy and F1-Score.

For a single microphone, the microphone **B** yielded the highest score in both experiments. When included in other datasets, it also yielded high accuracy scores. We think this is likely because the microphone used to record the FullSOL sound bank was placed at a similar distance.

Using more than three microphones in both experiments improved the classifier's generalization ability. However, using more than five microphones degraded performance. We think using more than seven microphones would not enhance the classifier's performance.

The confusion matrices show that *key click* and *tongue ram* techniques are poorly identified when using original audio samples. Using pitch-shifted audio samples improved the accuracy of *key click*, but *tongue ram* remains misidentified. We think leveraging self-supervised learning for general-purpose audio representation systems such as BYOL-A [22, 23] is an approach to consider. The model learns meaningful representations from massive unlabeled audio data using self-supervised learning during pre-training. The variety of internal representations would improve the model's robustness when fine-tuned for a specific task. Further research is required to assess BYOL-A's efficiency in automatically recognizing IPT.

6. CONCLUSION

In this study, we proposed a microphone-based data augmentation method for automatically recognizing instrumental playing techniques (IPTs). We created datasets of audio samples recorded at different distances from the sound source. Training a classifier with these datasets improved its performance, especially with close-to-source microphones and the addition of pitch-shifted audio samples. This method achieved state-of-the-art results with an accuracy score of 91.32% and a macro F1 Score of 75.56% on heterogeneous datasets. Although our study focused on flute IPTs, our method can be applied to other musical instruments as well.

The proposed classifier has shown high performance in most flute playing techniques included in the two experiments. However, the accuracy of short-duration IPTs is low, and adding pitch-shifted audio samples to the original audio samples did not enhance the accuracy. We will address this particular concern in a future study.

Acknowledgments

We want to thank Kanami Koga, the flutist whom we recorded the sound bank. This research is related to ERC Reach Project (GA #883313).

7. REFERENCES

[1] A. Cont, "ANTESCOFO: Anticipatory Synchronization and Control of Interactive Parameters in Computer Music." in *International Computer Music Conference (ICMC)*, 2008, pp. 33–40.

[2] V. Lostanlen, J. Andén, and M. Lagrange, "Extended playing techniques: the next milestone in musical instrument recognition," in *Proceedings of the 5th international conference on digital libraries for musicology*, 2018, pp. 1–10.

[3] J.-F. Ducher and P. Esling, "Folded CQT RCNN for real-time recognition of instrument playing techniques," in *International Society for Music Information Retrieval*, 2019.

[4] N. Brochec and T. Tanaka, "Toward Real-Time Recognition of Instrumental Playing Techniques for Mixed Music: A Preliminary Analysis," in *International Computer Music Conference (ICMC 2023)*, 2023.

[5] J.-F. Ducher and P. Esling, "Apprentissage profond pour la reconnaissance en temps réel des modes de jeu instrumentaux," in *Journées d'Informatique Musicale*, 2019.

[6] A. Martelloni, A. P. McPherson, and M. Barthet, "Real-time Percussive Technique Recognition and Embedding Learning for the Acoustic Guitar," *arXiv preprint arXiv:2307.07426*, 2023.

[7] N. H. Fletcher and T. D. Rossing, *The physics of musical instruments*. Springer Science & Business Media, 2012.

[8] P.-Y. Artaud and G. Geay, *Flûtes au présent: traité des techniques contemporaines sur les flûtes traversières à l'usage des compositeurs et des flutistes*. Editions Jobert & Editions musicales transatlantiques, 1980.

[9] C. Levine and C. Mitropoulos-Bott, *The Techniques of Flute Playing I/Die Spieltechnik der Flöte I*. Bärenreiter-Verlag, 2019.

[10] C. E. Cella, D. Ghisi, V. Lostanlen, F. Lévy, J. Fineberg, and Y. Maresz, "OrchideaSOL: a dataset of extended instrumental techniques for computer-aided orchestration," *arXiv preprint arXiv:2007.00763*, 2020.

[11] N. Brochec and W. Howie, "GFDatabase: A Database of Flute Playing Techniques," Apr. 2024. [Online]. Available: https://doi.org/10.5281/zenodo.10932398

[12] V. Lostanlen and C.-E. Cella, "Deep convolutional networks on the pitch spiral for musical instrument recognition," *arXiv preprint arXiv:1605.06644*, 2016.

[13] 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, 2016.

[14] T. Park and T. Lee, "Musical instrument sound classification with deep convolutional neural network using feature fusion approach," *arXiv preprint arXiv:1512.07370*, 2015.

[15] P. Li, J. Qian, and T. Wang, "Automatic instrument recognition in polyphonic music using convolutional neural networks," *arXiv preprint arXiv:1511.05520*, 2015.

[16] M. Blaszke and B. Kostek, "Musical instrument identification using deep learning approach," *Sensors*, vol. 22, no. 8, p. 3033, 2022.

[17] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. pmlr, 2015, pp. 448–456.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[19] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[20] S. Wei, S. Zou, F. Liao *et al.*, "A comparison on data augmentation methods based on deep learning for audio classification," in *Journal of physics: Conference series*, vol. 1453, no. 1. IOP Publishing, 2020, p. 012085.

[21] M. Grandini, E. Bagli, and G. Visani, "Metrics for multi-class classification: an overview," *arXiv preprint arXiv:2008.05756*, 2020.

[22] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, "Byol for audio: Self-supervised learning for general-purpose audio representation," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[23] ——, "BYOL for audio: Exploring pre-trained general-purpose audio representations," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 137–151, 2022.