CNN-based Note Onset Detection using Synthetic Data Augmentation

Mina Mounir
Dept. of Electrical Engineering
(ESAT-STADIUS) KU Leuven
Leuven, Belgium
mina.mounir@esat.kuleuven.be

Peter Karsmakers
Dept. of Computer Science
(DTAI-ADVISE) KU Leuven
Geel campus, Belgium
peter.karsmakers@kuleuven.be

Toon van Waterschoot
Dept. of Electrical Engineering
(ESAT-STADIUS) KU Leuven
Leuven, Belgium
toon.vanwaterschoot@esat.kuleuven.be

***Abstract*—Detecting the onset of notes in music excerpts is a fundamental problem in many music signal processing tasks, including analysis, synthesis, and information retrieval. When addressing the note onset detection (NOD) problem using a data-driven methodology, a major challenge is the availability and quality of labeled datasets used for both model training/tuning and evaluation. As most of the available datasets are manually annotated, the amount of annotated music excerpts is limited and the annotation strategy and quality varies across data sets. To counter both problems, in this paper we propose to use semi-synthetic datasets where the music excerpts are mixes of isolated note recordings. The advantage resides in the annotations being automatically generated while mixing the notes, as isolated note onsets are straightforward to detect using a simple energy measure. A semi-synthetic dataset is used in this work for augmenting a real piano dataset when training a convolutional Neural Network (CNN) with three novel model training strategies. Training the CNN on a semi-synthetic dataset and retraining only the CNN classification layers on a real dataset results in higher average $\mathbf{F_1}$-score ($\mathbf{F_1}$) scores with lower variance.**

***Index Terms*—CNN, data augmentation, note onset detection**

I. Introduction

Many music processing and music information retrieval (MIR) applications rely on an elementary music analysis task known as NOD. In this task the detection algorithm is expected to produce an ordered list of time instants marking the start of the successive notes constituting a music excerpt. The majority of NOD solutions follows a three-step processing methodology: pre-processing, reduction function estimation, and peak-picking. Most of the recent and ongoing work focuses on the middle step, the objective of which is to yield an onset detection function (ODF): a highly sub-sampled version of the original signal, highlighting the note onsets while suppressing irrelevant signal details. Existing algorithms for ODF estimation fall into two main categories: data-driven and non-data-driven. The latter represents the traditional solutions exploiting the transient nature of a musical note onset in contrast to its subsequent steady-state portion or some other onset properties. These algorithms are completely independent from the data to be processed, as opposed to the data-driven algorithms. Checking the results reported in the Music Information Retrieval Evaluation eXchange (MIREX) NOD challenge [1], the CNN suggested by [2] is currently the best performing data-driven algorithm in terms of F_1.

A crucial problem in the design of data-driven NOD algorithms is the amount and quality of the available development datasets. For instance, the previously mentioned CNN model [2] is fitting around 290K parameters using one of the largest NOD datasets, made available by the authors upon request. This dataset was obtained by combining different datasets, resulting in a total size of 102 minutes (321 files, 28K onsets). A processing rate of 100 frames-per-second (fps) was used and the results reported in [2] were based on an 8-fold cross-validation, which implies the model was trained on 7/8 of the entire dataset for each of the folds. This translates to 89.25 minutes or 535K training points, which is hardly twice the number of model parameters. Despite the number of training points being far from the rule-of-thumb of 10 times the number of model parameters, a well performing model was obtained in [2], presumably thanks to training the CNN over many epochs, hence effectively increasing the number of training points. Nevertheless, as for any data-driven model, it would be beneficial to have more training data. which is difficult to achieve for the NOD task because most of the available NOD datasets are manually labeled. Manual labeling is typically performed in two steps, listening to the music excerpt and then visually assessing its time-frequency representation, and is usually carried out independently by two or more expert annotators. For these reasons it is a costly and time-consuming operation. Moreover, as the resulting annotations are subjective and context-dependent, these should be used with care. This is usually handled by employing an evaluation window of about 50 ms around the onset ground truth [3].

One way to tackle the data availability problem, in particular for deep-learning-based NOD algorithms, is data augmentation. To our knowledge, the only work done for NOD dataset augmentation is reported in [4], where it was suggested to increase the model complexity by two modifications to the model proposed in [2]. These two modified models were

This research work was carried out at the ESAT Laboratory of KU Leuven. The research leading to these results has received funding from the KU Leuven Internal Funds C2-16-00449, IMP/14/037, and VES/19/004, and the European Research Council under the European Union's Horizon 2020 research and innovation program / ERC Consolidator Grant: SONORA (no. 773268). This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information.

trained by using data augmentation strategies common in music applications: transposition, time stretching, spectral envelope transposition, and remixing of sinusoidal and noise components. Unfortunately the modified models did not yield an enhanced NOD performance, and sometimes even resulted in a performance degradation.

The objective of the current paper is to propose a different strategy for tackling the data availability problem, based on the generation of automatically annotated semi-synthetic music excerpts. We assess the effect of using datasets comprising such excerpts on the performance of deep-learning-based NOD algorithms. More specifically, three model training strategies are proposed and evaluated. In the first strategy, the model is simultaneously trained on a semi-synthetic and a real dataset. In the second and third strategy, a Universal Onset model (UOM) is trained on a semi-synthetic dataset, which is then retrained (in two different ways resulting in two different strategies) on a real dataset.

The paper is organized as follows. The data-driven approach to NOD is introduced in Section II, focusing on CNN modeling and data annotation. The proposed model training strategies are described in Section III. Section IV introduces the different datasets used, and shows initial results regarding the use of single vs. multiple instrument types in the semi-synthetic dataset. Details of the CNN model [2] are provided in Section V, after which the main results are presented in Section VI and conclusions are formulated in Section VII.

II. Data-driven modeling and annotation

A. Data-driven NOD with CNN model

In the data-driven approach to NOD, a model is trained based on input data consisting of music excerpts each containing multiple note onsets, and a corresponding annotation consisting of the ground truth note onset times. In this paper, we will consider a CNN model (detailed in Section V) for NOD which can indeed be considered the state-of-the-art data-driven solution [1]. A CNN is usually composed of interleaved convolutional and pooling layers, constituting the feature-learning layers, followed by a few fully connected layers forming the classification layers. Alternatively, these classification layers can be replaced by a Support Vector Machine (SVM) or a K-Nearest Neighbours (KNN) classifier.

Considering the fact that NOD is a binary classification problem with an extremely imbalanced distribution of classes (i.e., there are many more time instants without note onsets than with note onsets), the F_1 score is an appropriate evaluation metric. It is defined as the harmonic mean of the precision (P) and recall (R), i.e.,

$$F_1 = \frac{2PR}{P+R}. \tag{1}$$

A crucial question when assessing the CNN performance on small datasets is how well the reported scores will generalize to other evaluation datasets. One way to address this question is to report the scores after a K-fold cross-validation instead of using a straight train-validate-test workflow which would yield a good generalization only when working with large datasets. In a K-fold cross-validation, the dataset is first divided into K folds, after which K train-test rounds are performed. In each of the K rounds, it is common to use $K-1$ folds for training and 1 fold for testing. The final score is the average of the K test scores. In this work, we adopt a different data split for the K-fold cross-validation, using 1 fold for training and $K-1$ folds for testing. Such data split has been shown useful to assess the performance of data-driven models in applications where a limited amount of training data is available [5], as is typically the case in data-driven NOD.

B. Manually vs. automatically annotated datasets

As mentioned above, the manual annotation of note onsets is a time-consuming and cumbersome procedure. Automatic annotation is therefore an attractive alternative, which could yield larger and more consistently annotated datasets. A first approach to automatic annotation consists in building datasets using musical instruments that have both an audio and MIDI output (e.g., the MAPS dataset [6]). This is however only possible for a limited range of instruments, i.e. (semi-)electronic instruments such as the Yamaha Disklavier used in [6].

A second approach, proposed in this paper, is to generate semi-synthetic datasets in which the music excerpts are composed by mixing recordings of isolated notes for which the onsets are automatically annotated *prior to mixing*. Adopting the definition of a note onset being "the first detectable part of the note in an isolated recording" [7], it can be understood that a simple energy measure suffices to automatically annotate note onsets in isolated note recordings. In this way, we can generate large datasets with consistent annotations obtained at minimal cost. Even if the proposed approach strongly depends on the availability of datasets with isolated note recordings, a large amount of semi-synthetic NOD datasets can be generated from just a limited amount of isolated note recordings, by varying the mixing parameters. Also note that we apply a random mixing strategy (i.e., random note selection, random ordering and spacing of notes), hence resulting in excerpts with somehow no musical structure. This can be justified by observing that data-driven NOD algorithms generally do not rely on such structure.

III. Proposed model training strategies

When using semi-synthetic datasets for NOD, one should be cautious not to overfit the model to the way the semi-synthetic data were generated. To this end, we propose three strategies to train a CNN model for NOD based on both real and semi-synthetic data. As a benchmark, we will compare the models resulting from these three strategies with models trained on only real or only semi-synthetic data, resulting in the following five model training strategies, the performance of which will be compared in Section VI:

- **Real (R):** the model is trained only on real music excerpts. The excerpts could be manually annotated as in, e.g., [2] or alternatively, automatically annotated when using MIDI-supporting instruments as in, e.g., [8].

- **Synthetic (S):** the model is trained only on an automatically annotated semi-synthetic dataset.
- **Synthetic + Real (S+R):** the first proposed model training strategy consists in simultaneously training the model on excerpts from both datasets, by randomly interleaving real and semi-synthetic excerpts in the training set.
- **Synthetic pre-train followed by Real retrain ($\boldsymbol{S \triangleright R}$):** in the second and third proposed model training strategy, the model is first pre-trained on a semi-synthetic dataset then retrained on a real dataset. This strategy resembles transfer learning, adapting a UOM trained on semi-synthetic data to real music excerpts. Given the CNN structure composed of one or more feature layers (convolution, pooling) followed by a number of classification layers (fully connected), see Section II-A, the retraining in this strategy is run only on the classification layers as it can be assumed that the feature layers learned from one dataset can be used to generate useful features for another dataset as well. Moreover, this approach to retraining also protects the model from overfitting to a specific dataset. Two flavors of this strategy are proposed and evaluated in this paper for a CNN having two feature layers, see Section V:
  - **Partial freeze ($\boldsymbol{S \triangleright R_1}$):** in this strategy only the first feature layer is frozen, leaving the second layer open to learn additional features that couldn't be learned only from the semi-synthetic dataset. Hence we will use the symbol $S \triangleright R_1$ to denote this strategy of freezing the first feature layer only.
  - **Full freeze ($\boldsymbol{S \triangleright R_F}$):** here, the model is only relearning the classification layers, assuming the semi-synthetic dataset suffices to learn a generic feature model for note onsets. This full freezing strategy is denoted by the symbol $S \triangleright R_F$.

When training a data-driven model on datasets from different origins, it is important to consider how to deal with their respective annotations. Specifically for NOD evaluation, we suggested in our recent work [9] to take into account the time shift in annotations between the different datasets especially when training a Deep Neural Network (DNN). For instance, a semi-synthetic dataset may consistently have its onsets annotated earlier or later than a real dataset. Here, we adopt the approach of [9], by treating the *annotations time shift* as a hyperparameter determining the optimal temporal alignment of dataset annotations. Practically, given two datasets A and B from different origins, this hyperparameter is tuned by using the model learned on dataset A and estimate the annotations time shift δ that, when applied to the whole dataset, maximizes the performance of this model on dataset B. When training a model on dataset B, the annotations B are then time-shifted by $-\delta$.

IV. Datasets

Two semi-synthetic datasets were generated and tested in the frame of this paper. First, a large semi-synthetic dataset was generated for a wide range of instruments and it was termed All Instruments Semi-Synthetic (AISS) dataset. It was created using the isolated recordings for 138 different combinations of instrument and playing style from the McGill University Master Samples (MUMS) library [10]. The available notes per instrument and playing style were divided into 3 groups: 60% for training, 20% for validation and 20% for testing. Each of these groups were then used to produce a number of semi-synthetic music excerpts by mixing the isolated notes: 4 training mixes, 1 validation mix and 1 testing mix per instrument and playing style. Every mix contained 150 randomly selected notes with random inter-onset distances resulting in a tempo uniformly distributed in the range $[2, 2646]$ beats per minute (bpm). The note amplitudes were also randomized and the mixing was polyphonic, i.e., notes could (partially) overlap in time.

Despite the large amount of annotated examples, the optimization of the CNN model parameters using the AISS dataset stalled rapidly – after 10 epochs – for both the training and validation. In other words, the model was not able to continue learning which we believe may relate to the strong diversity of onset characteristics over the different instruments in the AISS dataset.

A second semi-synthetic dataset was therefore created, consisting of excerpts with only one type of instrument, namely the piano. This so-called Piano Semi-Synthetic (PSS) dataset was built from the piano recordings in the MUMS library, corresponding to 8 different pianos and playing styles or conditions: Concert Hall Steinway Soft, Hamburg Steinway Loud, Harmonics, Mpp Loud, Mpp Medium, Mpp Soft, Right Pedal Vol9 and Steinway Plucked. The PSS dataset generation process and parameters were exactly as for the AISS dataset, except for the amount of training mixes which was increased to 8 mixes to have a sufficiently large training set (see below). Figure 1 compares the validation and testing F_1 scores for the PSS and AISS datasets after training for 100 epochs, selecting the hyperparameters (annotations time shift and detection threshold) yielding the best validation F_1, and using the resulting model and hyperparameters to obtain the test F_1 scores. The box plots in the figure show the F_1 distribution over the different instruments/playing styles. It is clear that there is a higher average performance and less variance when restricting the NOD task to a single instrument type. Therefore, in the remainder of the paper we will focus on datasets of a single instrument type, and will use the PSS dataset as the semi-synthetic dataset when assessing the proposed model training strategies.

Further, two real datasets were selected from the MAPS dataset [6], more specifically the ENSTDkCl and ENSTDkAm datasets which will be referred to here as MAPS_CL and MAPS_AM. Also these two datasets are of the single instrument type (piano) and were recorded using a Yamaha Disklavier with annotations generated automatically from the output MIDI file. The suffixes "CL" and "AM" refer to *close* and *ambient* recording, hence specifying the distance between the microphone and the instrument.

Table I provides a summary of the datasets introduced here,

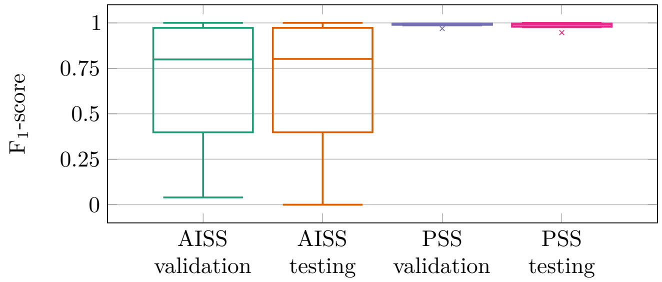

Fig. 1. Comparison of validation and testing F$_1$ scores for CNN model trained on synthetic datasets (piano and all instruments).

TABLE I
DATASET PROPERTIES

Name	Files	Time (min.)	Points (K)	Onsets (K)	
				B.C.	A.C.
AISS	828	597.3	3583	124.2	122.4
PSS	88	125.4	752	13.2	13.1
MAPS_CL	30	136.1	753	76.4	41.8
MAPS_AM	30	125.5	817	78.0	37.2

TABLE II
CNN STRUCTURE [2]

Layer	C_{in}	C_{out}	S_K	S_M
convolution	3	10	7x3	640
max-pool	10	10	1x3	0
convolution	10	20	3x3	1820
max-pool	20	20	1x3	0
fully-connected	1120	256	1	286976
fully-connected	256	1	1	257

in terms of number of music excerpts (Files), total duration (Time), number of input points fed to the CNN with 100 fps processing, and amount of onsets per dataset. As in most other NOD research, onsets less than 30 ms apart are combined into a single onset for evaluation, therefore the last column in Table I differentiates between number of onsets before combination (B.C.) and after combination (A.C.).

Even though from Table I it may seem that the PSS dataset is smaller than the MAPS datasets, that is not the case when considering the data subsets used for training. The PSS dataset is composed of 80% training, 10% validation and 10% testing excerpts. The training subset of length 113 min (676K points, 11.8K onsets) will be the set "S" used in the model training strategies proposed in Section III. On the other hand, each of the MAPS datasets is divided for every training/validation/test round in the K-fold cross-validation (explained in Section II), into 20% training, 10% validation and 70% testing. Hence, the MAPS training subset length amounts to about 25–27 min, which is 4 times less than the PSS training subset length.

V. CNN MODEL

In the following we summarize the characterization of the CNN proposed by [2] and point out the minor differences to the CNN used in our experiments. First, an input point is a 3D-tensor ($3 \times 15 \times 80$) containing 3 magnitude spectrograms with different processing window sizes (23 ms, 46 ms and 93 ms) but same frame rate of 100 fps. The number of logarithmically scaled MEL bands, from 27.5 Hz to 16 kHz, per frame is 80 while the number of frames per data point is 15. A data point is given a label "1" if the middle frame, i.e. the 8th of the 15 frames per data point, is matching an onset. Moreover, fuzziness in training is introduced by labeling the two neighboring data points with "0.25" to account for annotation ambiguity. Note that neighboring data points are 10 ms apart and jointly occur in 14 out of 15 frames. As labeling starts by checking the middle frame, the complete feature sequence per music excerpt is padded with 7 frames at both the start and end, consisting of repetitions of the first and last frames. No feature normalization is applied as it seemed unnecessary for our experiments.

We used the exact network structure as in [2] and as detailed in Table II, where for each layer, C_{in} denotes the number of input channels, C_{out} is the number of output channels, S_K is the kernel size representing the (time × frequency) dimensions, and $S_M = (C_{\text{in}} S_K + 1)\, C_{\text{out}}$ is the number of model parameters per layer. For the convolutional and max-pooling layers, channels refer to feature maps while in case of fully-connected layers, channels refer to neurons. A 50 % dropout is applied to the input of each of the fully-connected layers. Convolutional layers use the ReLU activation functions, and the fully-connected layers use the logistic sigmoid function. The training is done on mini-batches of 256 data points for 100 epochs using the adaptive moment estimation (Adam) optimizer [11] with its default parameters and minimizing the binary cross-entropy error. The order of training excerpts is shuffled after each epoch. This was found to result in a performance comparable to what has been reported in [2] when applied on their datasets with 8-fold cross validation and using the same folds.

VI. RESULTS

Here we compare the performance of the different model training strategies proposed in Section III. Two experiments are run, i.e., separately testing with each of two the real piano datasets MAPS_CL and MAPS_AM, and the resulting F$_1$ scores are shown in Fig. 2 and Fig. 3, respectively. Each boxplot shows the F$_1$ distribution among the different folds used per model training strategy (see Section IV for details on the size of the folds for the different datasets).

For both close and ambient recordings, we see that the average performance is better when training on only real data (R) than only semi-synthetic data (S), despite having many more semi-synthetic than real training points. This is not surprising as in the (R) setup, training/validation/testing excerpts originate from the same dataset and are hence much more similar in terms of note onset characteristics. Still,

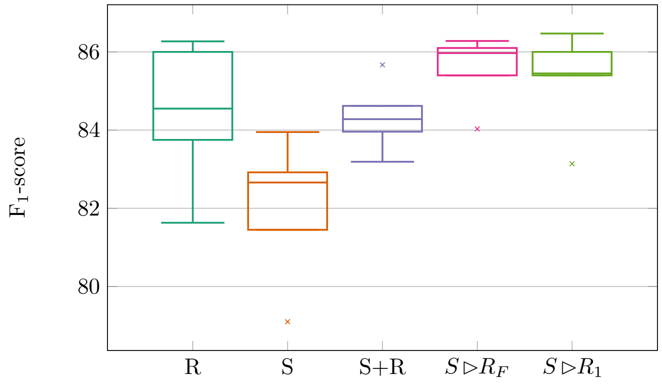

Fig. 2. F$_1$ scores for the different model training strategies tested on the MAPS_CL real dataset. Every boxplot is summarizing the 5-fold cross-validation test results.

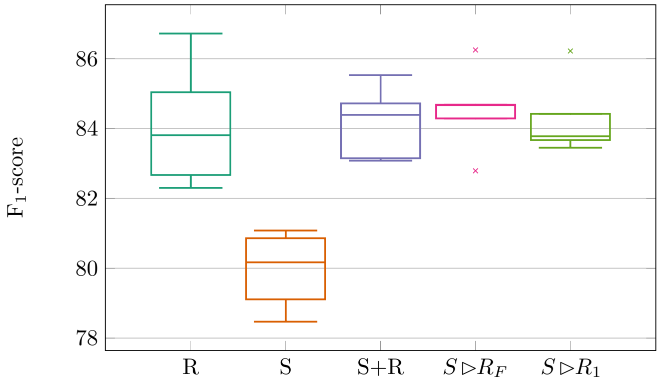

Fig. 3. F$_1$ scores for the different model training strategies tested on the MAPS_AM real dataset. Every boxplot is summarizing the 5-fold cross-validation test results.

training on the semi-synthetic dataset results in a lower F$_1$ variance compared to training on the real data, which is an observation that can be extended to all model training strategies involving the semi-synthetic data (i.e., S+R, $S \triangleright R_F$ and $S \triangleright R_1$) in both experiments. Moreover, a better average performance is obtained for almost all strategies in which real and semi-synthetic are combined. The $S \triangleright R_F$ strategy, in which the model was trained to learn note onset features from semi-synthetic excerpts and then retrained on real data, consistently outperforms the other model training strategies in terms of both mean and variance of F$_1$. This confirms our rationale that a more generic NOD model can be learned from a large semi-synthetic dataset, and that such model can serve as a UOM for onset detection. Comparing the results in Fig. 2 and Fig. 3, we notice that the onset detection becomes more challenging with ambient microphone recordings, presumably due to the presence of reverberation.

VII. Conclusion and Future Work

We proposed three new model training strategies that allow to augment the scarce datasets for NOD with automatically annotated semi-synthetic data. The best performing strategy consists in training a CNN model on semi-synthetic data for a single instrument type, effectively obtaining a UOM of which the classification layers can then be retrained on a real dataset. We suggest three directions for the further development of this work:

- **Single instruments:** To investigate the effect of further increasing the size and variability of the semi-synthetic dataset on the resulting CNN model performance. Dataset size can be increased by collecting more isolated note recordings and generating more mixes, whereas the variability can be increased by more strongly varying the density of onsets in semi-synthetic mixes, and adding background noise and reverberation.
- **Multiple instruments:** To study the effect of combining different instrument types into a joint semi-synthetic dataset for multi-instrument NOD. In this work, only two extreme cases (single instrument type vs. all possible instruments types) were considered, while a natural way of grouping instrument types would be to consider datasets of instrument families, e.g., winds, pianos, brass and strings.
- **Model complexity:** Due to the automated procedure for creating and annotating semi-synthetic mixes, the dataset size that can be generated in this way is practically unlimited. Having an unlimited amount of training examples opens perspectives to increase the CNN model complexity and enhancing the NOD performance by adding more feature maps or layers.

References

[1] MIREX, "Onset detection results 2018," https://nema.lis.illinois.edu/nema_out/mirex2018/results/aod/summary.html, 2018.

[2] J. Schluter and S. Bock, "Improved musical onset detection with Convolutional Neural Networks," in *Proc. 2014 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '14)*, May 2014, pp. 6979–6983.

[3] A. Lerch and I. Klich, "On The Evaluation Of Automatic Onset Tracking Systems," zplane.development, Techreport, Jan. 2005.

[4] A. Roebel, C. Jacques, and A. Aknin, "MIREX 2018: Training CNN onset detectors with artificially augmented datasets," UMR STMS - IRCAM, CNRS, Paris Sorbonne University, Tech. Rep., 2018.

[5] Y. Mu, W. Ding, and D. Tao, "Local discriminative distance metrics ensemble learning," *Pattern Recognition*, vol. 46, no. 8, pp. 2337 – 2349, 2013.

[6] V. Emiya, "MAPS database – a piano database for multipitch estimation and automatic transcription of music," http://www.tsi.telecom-paristech.fr/aao/en/2010/07/08/maps-database-a-piano-database-for-multipitch-estimation-and-automatic-transcription-of-music/, 2008.

[7] P. Leveau and L. Daudet, "Methodology and Tools for the evaluation of automatic onset detection algorithms in music," in *Proc. 5th Int. Symp. on Music Information Retrieval (ISMIR '04)*, 2004, pp. 72–75.

[8] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," *IEEE Trans. Audio Speech Lang. Process.*, vol. 18, no. 6, pp. 1643–1654, Aug. 2010.

[9] M. Mounir, P. Karsmakers, and T. van Waterschoot, "Annotations Time Shift: A Key Parameter in Evaluating Musical Note Onset Detection Algorithms," in *Proc. 2019 IEEE Workshop Appls. Signal Process. Audio Acoust. (WASPAA '19)*, New Paltz, NY, USA, Oct. 2019, pp. 21–25.

[10] F. Opolko and J. Wapnick, *McGill University Master Samples*. Montreal, QC, Canada: McGill University, Oct 2006, DVD edition.

[11] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. 3rd Int. Conf. on Learning Representations (ICLR '15)*, San Diego, CA, USA, May 2015.