Automatic Transcription of Ornamented Irish Flute Music

Yihao Wang
Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
yihaow@andrew.cmu.edu

Yuwen Chen
Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
yuwenc2@andrew.cmu.edu

Tianyi Peng
Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
tianyip@andrew.cmu.edu

Abstract

Many traditional music pieces are taught and passed down by hand and do not have written descriptions such as sheet music. Irish flute music is one such example and would potentially be lost in the near future if there are no actions being made on transcription. Manually transcribing music pieces is inefficient and inaccurate at times, thus motivating research into using novel machine-learning models to automate the process. Limited research has been conducted in this field and this project aims to contribute to this effort so that they can be prolonged for future generations to study and enjoy. The project has focused on implementing traditional methods as well as using deep learning models (CNN) to conduct onset detection and musical segment type classification which are both crucial factors in automatically transcribing a piece of audio music.

1 Introduction

Transcribing music into music notations represents the uniqueness and advance of human intelligence. This process requires humans to perceive, recognize, acknowledge and infer the music pieces. The goal of an automatic musical transcription (AMT) model is to convert acoustic signals into high-level information, making signals more accessible and comprehensive. Normally, the high-level information is in the format of piano-roll representations[5], or score formats[4][18]. The AMT task is challenging in the fields of both signal processing and artificial intelligence. It includes subtasks such as multi-pitch estimation, onset and offset detection, instrument recognition, beat and rhythm tracking, interpretation of expressive things and dynamics, and score typesetting. In this report, we implemented an AMT system with the function of onset and segment type identification.

For our current automatic transcription music model with a focus on traditional Irish flute music, we adopted both traditional and deep learning-based methods. Our goal is to detect onsets for each note

of the annotated flute music, and the type of segment they belong to. In section 4, we discussed the implementation of the spectral-flux method which performs onset detection in the spectral domain[15]. We also implemented Convolutional Neural Networks (CNNs) and treated both detection problems as computer vision tasks. The architecture of the base model is inspired by J.Schluter and S.Bock[19]. The dataset is a collection of traditional Irish flute music recordings which are embedded with characteristics of traditional Irish music, such as Irish tunes and ornamentation[14]. We explored the performance of conventional detection methods (spectral-flux method), base CNN models, and transfer learning to enhance the model's capabilities. We calculated the accuracy for the conventional approach and evaluated popular metrics for the deep learning methods (e.g. precision, recall, f1-score, and so on). Using transfer learning improved the baseline CNN model to an AUROC score of 0.838 while the newly implemented music segment prediction model achieved an F1-score of 0.70.

2 Literature Review

2.1 Types of automatic transcription models

Based on different types of model inputs, we classified the automatic music transcription(AMT) model into two categories. If the input to the model is a frame-level spectrogram, and each frame matches with one output, then the AMT model is considered as **frame-based model**. If the model takes a segment of or the whole spectrogram as the input, and each segment matches with multiple notes, then the AMT model is considered as **note-based model**[6]. Compared with the note-based model, the frame-based model is less complicated as it hasn't considered the relations between each frame. However, some musical work is the aggregation of long-term structures. To consider the integrity of musical transcriptions, several models based on a hidden Markov model (HMM) [10] and a recurrent neural network (RNN) [12][6] were proposed recently. We applied the frame-level model to our onset detection problem as we wanted to explore how onsets are represented in the frequency spectrum and use the advantages of CNNs to perceive edges in a picture.

2.2 Irish Traditional Music

Irish traditional music is a creative and lively musical form. Tunes, as the basic structure of Irish music, are usually uncomplicated and common. Ornamentation is the critical feature determining the style of Irish traditional music, expressed through improvisation in various musicians' performances. However, even though there are few previous research targeting at Irish traditional music styles, ornaments are normally ignored and insufficiently recorded[11][3]. To answer the questions concerning the differences in individualistic and regional styles in Irish traditional music, a variety of recordings containing diverse performances by musicians should be collected, analyzed, and compared. This goal inspired the birth of this manually annotated dataset and also encouraged us to build an automatic music model specialized for Irish traditional music.

Tunes and Ornaments The term "tune" is regarded as a melody that is usually composed of parts that may be repeated several times. The tunes normally include two segments and tunes with more than two segments are not common[14]. Ornaments can be seen as the embellishments of the melodic lines in Irish traditional music[17]. Performers can use the particular fingered expression to create ornaments on the flute. However, the ornaments are usually not marked in the script[16], and the option of using ornaments is improvised. Single-note and multi-note ornaments are two common forms of ornaments.

Mannual Annotation The traditional Irish music data set is annotated by Köküer, M., Ali-MacLachlan, et al.[14] using a software tool called Tony. Tony is designed for the interactive annotation of melodies from monophonic audio recordings. The information on each segmentation in the flute music pieces includes the time of onset, time of offset, duration of the note (or ornament), type of segment, note identity (if applicable), and note frequency (if applicable). The type of segment consists of note, one of the kinds of single-note or multi-note ornaments, and breath.

2.3 Onset Detection

There is a wide range of methods for onset detection in musical signal analysis. Traditional methods focused on the variations in the energy of the signal and analyzed them in the temporal or spectral

domain. J. P. Bello, G. Monti, et al.[1] used fundamental frequency (F0) for onset detection, but in their work, the F0 estimation method still suffers from octave errors. A. Klapuri[13] utilized multiple-F0 estimation to build an automatic music transcription system to extract pitches, onset times, etc. His algorithm is effective in rich polyphonies. For example, the proposed system achieved an error rate below 10% in six-note polyphonies. There is also another method for monophonic F0 estimation called YIN algorithm. It analyses only a single proficient frequency. In A.Klapuri's work, the YIN algorithm achieved 4.1% error rate for isolated notes.

Recently, some machine learning-based algorithms are applied to build an automatic musical transcription system. B. Fuentes, R. Badeau, and G. Richard [7] proposed a Harmonic Adaptive Latent Component (HALC) model based on Probabilistic Latent Component Analysis (PLCA), making it suitable for notes having variations in both pitches and spectral envelopes. G. E. Poliner and D.Ellis[8]designed the detection system using a support vector machine trained on spectral features, which detects frame-level instances. The output of this system is then smoothed by note-level HMM to perform the transcription. Their system achieved an accuracy of 62.3% for note onset transcription with a tolerance of 100 milliseconds. The recordings used are all played by the piano.

Jan. S and Sebastian. B [19] utilized Convolutional Neural Networks (CNN) to detect onset. They regarded this problem as detecting edges on the spectrogram. The dataset is 102 minutes in length and comprises 25927 notes. It achieved an F-score of 90.3%, about three percent above the previous RNN-based model. Our project is inspired by their work, and we will create our own CNN-based model detecting both notes and ornaments in Irish traditional music.

S.Sigtia and E.Benetos[20] employed the architecture which provides a principle way of superposing an RNN to the predictions of an random frame-level classifier and combines two models under a common training objective. It is superior to use RNNs for high-dimensional problems like AMT since the outputs of the RNN form a distributed representation, which makes the parameter estimation problem more tractable compared to an HMM.

As ornaments are short in duration, making them hard to detect and annotate, there is only a little work related to ornaments detection. Boenn, et al.[2]focused on the automatic quantization and rhythmic transcription of syncopated rhythms and baroque ornaments. comprise any machine learning-based techniques. In their work, there are 65% of ornaments were transcribed based on prior knowledge of downbeat locations, and 95% of ornaments were transcribed based on prior knowledge of single-beat locations. The work done by M. Gainza, E. Coyle, et al.[8] is the only research about ornaments detection in the Irish flute. However, the data set used in their work is on a relatively small scale. They built their onset detection system based on comb filters (ODCF). The database of flute signals consists of 290 notes. 36 notes are single-ornamented and the extra 5 multi-note ornaments are included. For single-note ornaments, the accuracy is 58.33%, while the accuracy for detecting multi-note ornaments is 80%.

2.4 Transfer Learning

In general, machine learning and deep learning are based on the assumption that training data are drawn from the same distribution as the testing data. However, this may not always be true in practical datasets. Also, a common difficulty is that some types of data are hard and expensive to collect. Hence, developing a training mode that can utilize the learning experience on more easily obtained data is in demand, which promotes the emergence of transfer learning [22].

3 Data

The pretraining progress is performed on MusicNet, a large public musical dataset containing 330 freely-licensed classical music recordings [21]. Mel-frequency cepstral coefficients (MFCCs) are coefficients that collectively compose a Mel-frequency cepstrum (MFC). MFCCs are commonly used to characterize speakers. The number of MFCCs is set as 20. It is necessary to match each frame with each onset accurately. For instance, as the onset information is recorded every 0.01 second, if the length of each frame of audio is set as 160, then the frame contains the information lasting for 0.01 second. We computed the single magnitude spectrograms with a hop size of 10 ms and window size of 20 ms, 30 ms, and 40 ms. Also, we extracted mfcc-delta features and mfcc-delta-delta features

with a hop size of 10 ms and a window size of 10 ms.

$$period\_contained\_in\_the\_frame = \frac{1}{sampling\_rate} \times frame\_length \tag{1}$$

After converting to spectral features and stacking the spectrograms generated with different window sizes but the same hop length, and different delta features but the same window and hop length, the data is in the format $channel \times frames \times band$. The number of samples and positive labels (positive if the given note is onset) in the training, validation, and testing set are shown in the table below:

Table 1: The number of samples and positive labels in training, validation and testing set

Set	# of samples	# of positive labels
Train	126,430	19,845
Validation	17,865	2,592
Test	17,977	2,500

3.1 Labels

Annotations include onset, offset, duration of the segment, type of segment, note identity, and note frequency. The manual annotations are exemplified in table2.

Table 2: An example of manual annotation from the first of the May's Fishers performed by H.Bradley

Onset(s)	Offset(s)	Duration(s)	Type of Segment	Note Identity	Note Frequency
0.0348	0.13932	0.10449	NOTE	A4	463.523
0.13932	0.39474	0.25542	NOTE	C5	518.509
0.39474	0.44989	0.055147	shRoll_ct	D5	610.991
0.44989	0.51664	0.066757	shRoll_nt	C5	518.658
0.51664	0.55293	0.036281	shRoll_str	B4	504.825
0.55293	0.66177	0.10884	shRoll_nt	C5	519.887
0.66177	0.82141	0.15964	NOTE	D5	576.006
0.82141	0.93751	0.1161	lngRoll_nt	A4	463.358
0.93751	1.0144	0.076916	lngRoll_ct	C5	515.857
1.0144	1.0725	0.05805	lngRoll_nt	A4	462.43
1.0725	1.1059	0.033379	lngRoll_str	A4	443.255
1.1059	1.2118	0.10594	lngRoll_nt	A4	463.699
1.2118	1.2785	0.066757	ct	C5	523.07
1.2785	1.354	0.075465	NOTE	C5	510.808
1.354	1.4832	0.12916	NOTE	A4	462.412
1.4832	1.6298	0.14658	NOTE	D5	585.44
1.6298	1.7241	0.094331	NOTE	E5	696.36
1.7241	1.8489	0.12481	NOTE	A5	923.938
1.8489	2.0143	0.16544	br	BREATH	0

3.1.1 Onsets

In Table2, it is noticed that the units of our onset are 0.1ms and all the onsets are truncated to 2 decimal places, representing 10ms. For all the MFCC frames we extracted from audio files if the window length is 160 and hop length is 160, each frame contains the spectral information for one consecutive 0.01 second. If this frame is marked as an onset, the label of this frame is 1. Otherwise, its label is zero. As the amount of positive labels is small, we add fussy labels when we encode the onsets. If the current frame is annotated as an onset, the next frame after the current frame is also annotated as an onset.

3.2 Type of segments

The type of segments is categorized into notes, ornaments, and others. In ornaments, there are cut, strike, crann, roll, and shake. In others, there are breath and triplet. "Notes" is labeled as 1, "Ornaments" is labeled as 2, and "Others" is labeled as zero. The number of samples and each category is summarized in Table3.

Table 3: The number of samples and each category in training, validation, and testing set

Set	# of samples	# of notes	# of ornaments	# of others
Train	126,430	93,869	15,067	17,494
Validation	13,412	2,592	1,999	2,454
Test	17,977	13,078	2,074	2,825

4 Experiments

4.1 Signal energy: spectral domain

The signal energy-based method (spectral-flux) analyzes the signal in the spectral domain and measures how fast the power spectrum is changing. The variations of signal energy are calculated by comparing the power spectrum for one frame against the power spectrum from the previous spectrum.[9] We summarized the pseudocode for the detailed spectral-flux algorithm in Figure1.

Algorithm 1 Spectral-flux algorithm for onset detection

for overlapped signal_frame in signal **do**
    $signal\_frame = hamming\_window(signal\_frame)$
    $signal\_frame = zero\_padded(signal\_frame)$
    $frequency\_bin = STFT(signal\_frame)$
end for
for each frequency_bin **do**
    $magnitude\_difference = frequency\_bin(i) - frequency\_bin(i+1)$
    $magnitude\_difference = half\_waved(frequency\_bin)$
    $detection\_value = L2\_Norm(magnitude\_difference)$
    if $detection\_value > threshold$ and *not two consecutive peaks are found within a given time instance* **then**
        $The\ frame\ is\ regarded\ as\ onset.$
    end if
end for

4.2 Convolutional Neural Networks

We borrowed the architecture and training details from work by Jan. S and Sebastian. B [19] to construct our baseline framework shown in 1. As reported by this baseline paper, they achieve around 90 percent accuracy utilizing the CNN model to detect onset notes, which outperforms our conventional methods. Therefore, it is worthy to implement such CNNs and evaluate their capability on our datasets. The input to this network is a 3-channel spectrogram, where each channel includes 21 frames by 20 bands. In the first convolutional layer, 10 rectangle filters of size 5×3 are adopted. The next max-pooling layer downsamples each frame by selecting the maximum number of values of 3 adjacent bands without overlap, which reduces the map size to 6 bands. This pooling layer is then followed by another convolutional layer of 3×3 filters and a 2-band max-pooling layer. To prevent information loss, we increased the number of filters to 20. Eventually, this results in 20 feature maps which are all of the sizes 15×2. Before feeding into the fully-connected layer, all the activation values in these feature maps are flattened. After processing by a fully-connected layer of 256 units, the final fully-connect layer gives a binary output, indicating if the input notes are onset or not.

Following,. the baseline paper, we train this network for 100 epochs using a fixed learning rate of 0.05. The optimization in each epoch is performed by SGD with a momentum of 0.9. The batch size

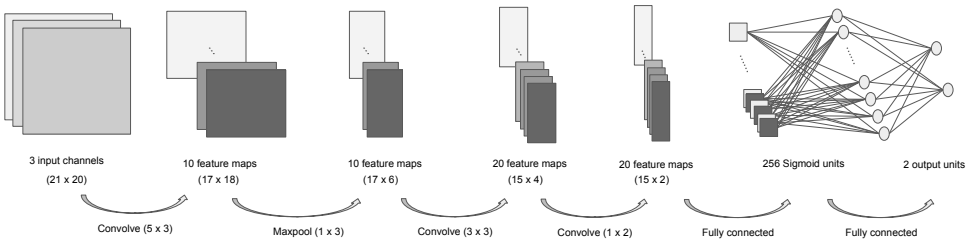

Figure 1: Baseline CNN architecture used in our work. This network includes convolution and max-pooling operations. All activation values in the last-level feature maps are input into a fully-connected layer. To predict onset notes, the final output is binary.

is set to 256 and the backpropagation is based on cross-entropy loss. The final weights are selected associated with the best validation accuracy. As not clearly stated in the baseline paper, we follow the Youden index [23] to select the threshold for label prediction.

4.3 Grid Search

In order to evaluate different training settings, we applied grid search on several significant hyper-parameters and thus, find the optimal combinations. The hyper-parameters we have explored are dropout rates, context values, learning rates, and learning schedulers.

4.4 Transfer Learning

To overcome the pitfalls on small training datasets, we adopt the transfer learning technique, which is inspired by human learning. Human is able to transfer knowledge between different tasks inherently. Similarly, it is expected that the model would better perform onsets detection on Irish flute music after learning multiple types of musical information. In our experiments, we trained the model on MusicNet using the optimal combination of the hyper-parameters we found during baseline model training.

4.5 Window features vs. Delta features

Window feature is constructed by the Mel-frequency cepstral coefficients (MFCCs) extracted from each frame. We concatenated the extracted MFCCs for all frames. As long as the hop length remains constant, when we extend the window length (which means more frames are involved when calculating MFCCs), the output MFCCs has the same length. We stacked three MFCC features with the fixed hop length and various window lengths as a 3-channel feature. Delta feature is constructed by MFCC features, MFCC delta features, and MFCC delta-delta features. MFCC delta feature and MFCC delta-delta are built by the local estimate of the derivative and the second derivative of the input MFCC data along the selected axis. Again, We stacked MFCC, MFCC-delta, and MFCC delta-delta features with fixed hop length and fixed window length as a 3-channel feature.

To further search for another type of feature that may better represent onsets, we also built models based on delta features and compared the outcomes with that based on window features.

4.6 Type of segments

To further evaluate if our baseline model is able to perform more complicated tasks, we designed multilabel tasks based on different types of segments.

4.7 Metrics

Similarly, we adopted precision, recall, and F-score, as described in the baseline paper citeschluter2014improved as well as accuracy. In our cases, onset notes are assigned to a posi-

tive (1) label, and the calculations of these 4 metrics are shown below:

$$TP = P(pred = Onset|label = Onset) \tag{2}$$

$$TN = P(pred = Not\ onset|label = Not\ onset) \tag{3}$$

$$FP = P(pred = Onset|label = Not\ onset) \tag{4}$$

$$FN = P(pred = Not\ onset|label = Onset) \tag{5}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F1 - score = 2 \times \frac{Precision * Recall}{Precision + Recall} \tag{9}$$

Apart from these 4 metrics, we also calculate the AUROC value, which is the area under the ROC curve. AUROC can reflect the model's ability to discriminate different classes and is widely used in model evaluation.

In the task of predicting the type of segments, as it is a multi-class classification task, we calculated metrics for each label, and find their average weighted by support (the number of true instances for each label).

5 Results

5.1 Signal energy: spectral domain

Table 4: Result comparison between spectral-flux and CNN baseline model

Model	Accuracy	Precision	Recall	F1-score	AUROC
Spectral-Flux	0.730	0.170	0.130	0.150	0.500
CNN baseline	**0.748**	**0.328**	**0.776**	**0.464**	**0.832**

From Table 4, it is observed that even though the spectral-flux method reaches comparable accuracy, whether the such method is able to classify onsets given mfcc features still requires further validation as the AUROC is around 0.5. Additionally, the CNN-based model seems to be more reliable considering the improvement of not only accuracy but also other evaluation metrics. The AUROC value over 0.8 can also demonstrate the generalization and capability of the CNN model to classify onsets.

5.2 Transfer learning

Table 5: Result comparison between baseline model and pre-trained model

Model	Accuracy	Precision	Recall	F1-score	AUROC
CNN baseline	**0.748**	**0.328**	0.776	**0.464**	0.832
CNN (pretrained on MusicNet)	0.711	0.307	**0.855**	0.451	**0.838**

According to Table 5, there is a slight boost in AUROC after pretraining on the MusicNet. Moreover, the higher recall indicates the better ability of such a model on predicting onsets. Although there the model suffers a drop in accuracy, it could be claimed that the model generalization has been improved due to higher AUROC. From another perspective, one of the possible reasons that the model does not boost obviously is that MusicNet contained a different type of musical data which may lead to different representations in the same feature space.

Table 6: Result comparison between window features and delta features

Model	Accuracy	Precision	Recall	F1-score	AUROC
CNN (pretrained & Window features)	**0.711**	**0.307**	**0.855**	**0.451**	**0.838**
CNN (pretrained & Delta features)	0.671	0.270	0.803	0.405	0.786

5.3 Delta feature

From Table 6, it could be seen that the model trained on delta features under-performs that trained on window features. This may be due to that the extracted window features are more distinguishable between onsets and non-onsets while the extracted delta features are relatively more similar.

5.4 Prediction of type of segments

Table 7: Note identity prediction results based on the model with the best parameter set

Model	Accuracy	Precision	Recall	F1-score
Baseline Model(Window features)	0.72	0.70	0.72	0.70

Based on Table 7, it could be observed that our baseline model is able to perform more complicated tasks apart from simple binary prediction.

6 Future work

As the project is focused on a niche subcategory of music, the main difficulty was finding enough resources and data to support the development of better models. For example, the original dataset for Irish flute music contained a total of 162,272 samples which is minuscule compared to other popular datasets such as images or text which have practically an unlimited amount of well-prepared data to use. Further work could be done to put more emphasis on this area of research and develop more accessible datasets for researchers to work with. As mentioned in this report, the lack of data and overfitting is tackled by extending to additional goals as well as implementing transfer learning models that are able to take advantage of larger music datasets that contain other instruments. Although an improvement was obtained, future work could be exploring alternative learning frameworks that have similar properties such as few-shot learning. Alternatively, overfitting can be mediated by reducing model complexity. Although out of scope for this course, traditional machine learning methods such as decision trees, SVMs, logistic regression, etc. may produce better results and are worth putting effort into.

7 Conclusion

To summarize, the project has been successful on three fronts, altering the baseline model to fit Irish flute music data, extending the model to predict additional musical properties, and improving the baseline model by implementing a transfer learning framework. To the best of our knowledge, this project is the first to use deep-learning neural networks to predict aspects of Irish flute music. Since there is a scarce amount of resources in this field, the majority of this project was focused on selecting the right model and altering it to fit our needs. However, we have made progress in producing good results for the onset detection baseline model and improving it further by incorporating transfer learning into our model. Extensive experimentation was also conducted on feature selections and the results suggest that using delta features perform worse than window features due to the level of similarity between the extracted features. Additionally, an effort was made in extending the CNN model to predict the type of musical segment feature of the dataset and achieved similar levels of testing metrics as the onset detection model. This suggests that the model can be generalized and extended to classify multiple features of Irish flute music. Lastly, We hope this project will help promote further research combining deep learning techniques and the music field, which is significant to preserve musical heritage.

References

[1] Juan Pablo Bello, Giuliano Monti, Mark B Sandler, et al. Techniques for automatic music transcription. In *ISMIR*, 2000.

[2] Georg Boenn. Automated quantisation and transcription of ornaments from audio recordings. 2007.

[3] Breandán Breathnach. *Folk music and dances of Ireland*. Mercier Press, 1977.

[4] Ralf Gunter Correa Carvalho and Paris Smaragdis. Towards end-to-end polyphonic music transcription: Transforming music audio directly to a score. In *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 151–155, 2017.

[5] Ali Taylan Cemgil. *Bayesian music transcription*. PhD thesis, s.n., 2004.

[6] Kin Wai Cheuk, Yin-Jvun Luo, Emmanouil Benetos, and Dorien Herremans. The effect of spectrogram reconstruction on automatic music transcription: An alternative approach to improve transcription accuracy. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9091–9098. IEEE, 2021.

[7] Benoit Fuentes, Roland Badeau, and Gaël Richard. Harmonic adaptive latent component analysis of audio and application to music transcription. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(9):1854–1866, 2013.

[8] Mikel Gainza, Dan Barry, and Eugene Coyle. Automatic bar line segmentation. In *Audio Engineering Society Convention 123*. Audio Engineering Society, 2007.

[9] Dimitrios Giannoulis, Michael Massberg, and Joshua D Reiss. Automating dynamic range compression. *J. Audio Eng. Soc*, 61, 2013.

[10] Peter Jancovic, M Kokuer, and Wrena Baptiste. Automatic transcription of ornamented irish traditional flute music using hidden markov models. In *Int. Society for Music Information Retrieval Conference (ISMIR), 2015*, 2015.

[11] Niall Keegan. The art of juncture–transformations of irish traditional music. 2012.

[12] Jong Wook Kim and Juan Pablo Bello. Adversarial learning for improved onsets and frames music transcription. *arXiv preprint arXiv:1906.08512*, 2019.

[13] Anssi P Klapuri. Multiple fundamental frequency estimation based on harmonicity and spectral smoothness. *IEEE Transactions on speech and audio processing*, 11(6):804–816, 2003.

[14] Münevver Köküer, Islah Ali-MacLachlan, Daithí Kearney, and Peter Jančovič. Curating and annotating a collection of traditional irish flute recordings to facilitate stylistic analysis. *International Journal on Digital Libraries*, 20(1):107–121, 2019.

[15] Munevver Kokuer, Peter Jancovic, Islah Ali-MacLachlan, and Cham Athwal. Automatied detection of single-and multi-note ornaments in irish traditional flute playing. ISMIR, 2014.

[16] Münevver Köküer, Daithí Kearney, Islah Ali-MacLachlan, Peter Jančovič, and Cham Athwal. Towards the creation of digital library content to study aspects of style in irish traditional music. In *Proceedings of the 1st International Workshop on Digital Libraries for Musicology*, pages 1–3, 2014.

[17] Grey Larsen. *Essential Guide to Irish Flute and Tin Whistle*. Mel Bay Publications, 2011.

[18] Miguel A Román, Antonio Pertusa, and Jorge Calvo-Zaragoza. An end-to-end framework for audio-to-score music transcription on monophonic excerpts. In *ISMIR*, pages 34–41, 2018.

[19] Jan Schlüter and Sebastian Böck. Improved musical onset detection with convolutional neural networks. In *2014 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 6979–6983. IEEE, 2014.

[20] Siddharth Sigtia, Emmanouil Benetos, Nicolas Boulanger-Lewandowski, Tillman Weyde, Artur S d'Avila Garcez, and Simon Dixon. A hybrid recurrent neural network for music transcription. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2061–2065. IEEE, 2015.

[21] John Thickstun, Zaid Harchaoui, and Sham M. Kakade. Musicnet, Nov 2016.

[22] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016.

[23] W. J. Youden. Index for rating diagnostic tests. *Cancer*, 3(1):32–35, 1950.

A Github Codebase

https://github.com/Brauntt/Automatic-Transcription-of-Ornamented-Irish-Flute-Music

B Grid Search - Baseline model

Table 8: Grid search on different dropout rate (baseline model)

Model	Accuracy	Precision	Recall	F1-score	AUROC
Our baseline (no dropout)	0.742	0.323	0.777	0.456	**0.825**
Model with dropout=0.1	**0.745**	**0.325**	0.773	**0.458**	0.818
Model with dropout=0.2	0.716	0.305	0.814	0.444	0.825
Model with dropout=0.3	0.714	0.303	0.811	0.441	0.824
Model with dropout=0.5	0.696	0.293	**0.836**	0.433	0.823

Table 9: Grid search on different context values (baseline model)

Model	Accuracy	Precision	Recall	F1-score	AUROC
Our baseline (context=10)	0.742	0.323	0.777	0.456	0.825
Model with context=5	0.714	0.297	0.772	0.429	0.801
Model with context=15	**0.748**	**0.328**	0.776	**0.464**	**0.832**
Model with context=20	0.704	0.298	**0.834**	0.439	0.825
Model with context=32	0.715	0.306	0.832	0.448	0.831

Table 10: Grid search on different initial learning rates (baseline model)

Model	Accuracy	Precision	Recall	F1-score	AUROC
Our baseline (LR=0.05, context=15)	**0.748**	**0.328**	0.776	**0.464**	**0.832**
Model with LR=0.1	0.720	0.302	0.772	0.434	0.808
Model with LR=0.01	0.699	0.291	0.812	0.429	0.811
Model with LR=0.005	0.710	0.301	0.817	0.440	0.821
Model with LR=0.001	0.709	0.301	**0.827**	0.442	0.826

Table 11: Grid search on different learning rate schedulers (baseline model)

Model	Accuracy	Precision	Recall	F1-score	AUROC
Our baseline (Constant LR)	**0.748**	**0.328**	0.776	**0.464**	0.832
Model with Cosine Annealing	0.715	0.303	0.809	0.441	0.824
Model with Exponential LR	0.734	0.319	0.800	0.456	0.831
Model with ReduceLROnPlateau	0.715	0.308	**0.839**	0.450	**0.833**

C Grid search - Transfer learning

Table 12: Grid search on different dropout rate (transfer learning)

Model	Accuracy	Precision	Recall	F1-score	AUROC
Baseline (no dropout)	**0.742**	**0.323**	0.777	0.456	**0.825**
Model pretrained (no dropout)	0.703	0.300	0.853	0.444	0.833
Model pretrained (dropout=0.2)	0.700	0.301	**0.874**	**0.448**	0.834
Model pretrained (dropout=0.3)	0.711	0.307	0.855	0.451	**0.838**
Model pretrained (dropout=0.5)	0.706	0.305	0.70	0.452	**0.838**

Table 13: Grid search on different initial learning rates (transfer learning)

Model	Accuracy	Precision	Recall	F1-score	AUROC
Baseline (Dropout=0.3 & LR=0.05)	0.711	0.307	**0.855**	0.451	**0.838**
Model pretrained (LR=0.1)	0.715	0.307	0.832	0.448	0.832
Model pretrained (LR=0.01)	**0.728**	**0.317**	0.828	**0.458**	0.836
Model pretrained (LR=0.005)	0.719	0.310	0.836	0.453	0.834
Model pretrained (LR=0.001)	0.714	0.307	0.840	0.450	0.832

D Grid search - Delta features

Table 14: Grid search on different dropout rate (delta features)

Model	Accuracy	Precision	Recall	F1-score	AUROC
Baseline (no dropout)	0.671	0.270	0.803	0.405	0.786
Model pretrained (no dropout)	0.668	0.247	0.676	0.361	0.713
Model pretrained (dropout=0.2)	0.675	0.246	0.647	0.357	0.713
Model pretrained (dropout=0.3)	0.678	0.257	0.694	0.375	0.739
Model pretrained (dropout=0.5)	0.621	0.230	0.737	0.351	0.727

Table 15: Grid search on different initial learning rates (delta features)

Model	Accuracy	Precision	Recall	F1-score	AUROC
Baseline (Dropout=0.3 & LR=0.05)	0.678	0.257	0.694	0.375	0.739
Model pretrained (LR=0.1)	0.673	0.253	0.690	0.370	0.733
Model pretrained (LR=0.01)	0.590	0.203	0.667	0.312	0.664
Model pretrained (LR=0.005)	0.623	0.225	0.700	0.341	0.702
Model pretrained (LR=0.001)	0.614	0.219	0.689	0.332	0.685