Improving Polyphonic Piano Transcription using Deep Residual Learning

Arne Corvin Jaedicke

Audio Communication Group
Technische Univerität Berlin

This thesis is submitted for the degree of
Master of Science

First Supervisor:
Prof. Dr. Stefan Weinzierl
Second Supervisor:
Athanasios Lykartsis

June 2019

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt gegenüber der Fakultät I der Technischen Universität Berlin, dass die vorliegende, dieser Erklärung angefügte Arbeit selbstständig und nur unter Zuhilfenahme der im Literaturverzeichnis genannten Quellen und Hilfsmittel angefertigt wurde. Alle Stellen der Arbeit, die anderen Werken dem Wortlaut oder dem Sinn nach entnommen wurden, sind kenntlich gemacht. Ich reiche die Arbeit erstmals als Prüfungsleistung ein. Ich versichere, dass diese Arbeit oder wesentliche Teile dieser Arbeit nicht bereits dem Leistungserwerb in einer anderen Lehrveranstaltung zugrunde lagen.

Titel der schriftlichen Arbeit

Improving Polyphonic Piano Transcription using Deep Residual Learning

Verfasser

Jaedicke, Arne Corvin, Matrikel-Nr.: 325662

Betreunde Dozenten

Prof. Dr. Stefan Weinzierl,
Athanasios Lykartsis

Mit meiner Unterschrift bestätige ich, dass ich über fachübliche Zitierregeln unterrichtet worden bin und verstanden habe. Die im betroffenen Fachgebiet üblichen Zitiervorschriften sind eingehalten worden. Eine Überprüfung der Arbeit auf Plagiate mithilfe elektronischer Hilfsmittel darf vorgenommen werden.

Berlin, den 7.6.2019

.....

Abstract

In this thesis a new deep learning method is adapted for frame-wise polyphonic piano note transcription. It is based on the idea of Residual Learning which is then extended with Bidirectional Long Short-Term Memory units and a Multitask Learning strategy. Furthermore, the final transcription system applies an aggregation function to simultaneously detect the onset, pitch and offset of the notes. The use of complementary methods combined into one model enables the transcription system to significantly improve the note-level detection performance and allows it to produce perceptually rich transcriptions. The evaluation is performed on frame-level and note-level metrics and utilizes a common test set on the publicity available MAPS dataset. Thus, the proposed transcription system is recommended as the new state-of-the-art for this dataset.

Zusammenfassung

In dieser Abschlussarbeit wird eine neue Deep-Learning-Methode für die polyphone Transkription von Klaviernoten adaptiert. Der Ansatz basiert auf der Idee des Residual Learning, welches anschließend um Bidirectional Long Short-Term Memory Einheiten und eine Multitask-Lernstrategie erweitert wird. Darüber hinaus wendet das finale Transkriptionssystem eine Aggregationsfunktion an, um gleichzeitig den Onset, die Tonhöhe und den Offset der Noten zu erfassen. Die Verwendung komplementärer Methoden, welche in einem Modell kombiniert werden, ermöglicht es dem Transkriptionssystem, die korrekte Klassifikation auf Notenebene signifikant zu verbessern und eine perzeptuell überzeugende Transkriptionen zu erzeugen. Die Auswertung erfolgt auf Frame- und Notenebene und verwendet ein verbreitetes Testset auf dem öffentlich zugänglichen MAPS-Datensatz. Das untersuchte Transkriptionssystem wird als neuer Stand der Technik für diesen Datensatz empfohlen.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

The general purpose of Music Information Retrieval (MIR) includes the extraction and aggregation of information from audio data [1]. A sub area of MIR is the so-called Automatic Music Transcription (AMT). The transcription of music is the process of transforming audio recordings into a musical score or similar symbolic representation (e.g. MIDI). Three sub tasks can be identified as a necessary condition to this process: multi-pitch estimation, onset and offset detection [2]. These must be established as meticulously as possible to enable an accurate transcription of the music. More complex tasks, such as determining the intensity or velocity of a note, are subject to a greater scope of interpretation. Nevertheless, these basic tasks are a crucial first step in solving various more abstract MIR problems, such as instrument identification, source separation and music structure analysis [2]. First attempts in AMT have been already explored in the 1970s by Moorer [3] and it is only now that the first multi-pitch piano transcription system has reached a quality where perceptually relevant transcriptions are possible [4].

Due to the simultaneity of events in music, its transcription is a challenging task, even for humans and even if only one instrument is considered [5]. It takes a trained ear and a comparatively long time to translate a piece of music into a symbolic form. This is even further complicated by the different characteristics and the timbre of an instrument, introducing ever changing note attack and decay. Therefore, research on AMT has often focused on a solo instrument with well known properties and a high representative value - the piano.

In recent years, classical signal processing methods have been replaced by machine learning techniques, and today the majority of piano transcription systems is based on this approach [4, 6–9]. Different machine learning strategies now compete for a few percentage points of accuracy in the respective task. With deep learning, a more

contemporary branch of machine learning, there is a new candidate for improved results. Compared to other approaches, the function to be optimized is based upon very long chains of nonlinear operations called deep neural networks [10]. In particular, the success of deep convolutional networks in the field of image classification has resulted in many new innovations within the AMT community, mainly because of the similarities between images and spectrograms as a representation of audio data.

Up to this point, spectrograms still have a considerable advantage over raw audio data as a feature input to neural networks [11]. This is mostly due to the considerable information density of audio data, rather than that a neural network would not be capable of learning Fourier transformed features from raw audio. Therefore, machine learning on audio data still requires a certain amount of preprocessing. This step still demands domain knowledge and is often subject to uncertainty as to which method is best suited for a task.

Described in more general terms, the task of polyphonic piano transcription is one of mapping time-ordered frame-wise spectrogram features $\mathbf{x}^{(t)} \in \mathbb{R}^{\mathrm{F}}$ to a time-ordered frame-wise symbolic note representation $\mathbf{y}^{(t)} \in \{0,1\}^{\mathrm{K}}$: with F being the number of frequency bins in a time frame t and $\mathrm{K} = 88$ the tonal range of a piano. This output can then be converted to a set of tuples describing the pitch, onset and offset of a piano note.

The following thesis will demonstrate how the application of residual learning in combination with different deep learning methods achieves a relative improvement of the note-level F1-Score by 10 % compared to current state-of-the-art piano transcription.

1.1 Current State of Research

Early work on onset and pitch detection mostly relied on signal processing with spectrograms, using changes in a pitch detector to find note onsets [12] or by simply applying a magnitude threshold to semitone-filtered spectrograms [13]. A method that is still popular with AMT today is based upon Nonnegative Matrix Factorization and has also been successfully applied to polyphonic piano transcription [14]. Current research on onset and pitch classification tasks extend over a wide range of machine learning methods. Early experiments use Neural Networks (NNs) to improve a manually developed onset detection by learning peak picking on piano music [15]. Lacoste and Eck [16] use NNs with Short-Time Fourier Transforms (STFT) and constant-Q transform as input to detect onsets, but conclude that CNNs are more promising.

As mentioned above, many concepts applied by deep learning on spectrograms have their origin in image classification tasks, due to the similar input data. Particularly noteworthy are the findings from the work on LeNet [17], which contains the essence of todays CNNs, and AlexNet [18], a deeper version of LeNet. In 2014 the VGGNet (19 layer CNN) [19] performed very well on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) and illustrates how an extremely deep, yet simple network architecture improves image classification.

However, compared to images, the two dimensions in a spectrogram have a fundamentally different modality from each other. This is especially acknowledged by 1D convolutional layers, but recent work by Lostanlen and Cella [20] indicate that 2D convolution is superior to using only 1D in MIR tasks.

If one considers common audio preprocessing methods, these could also be replaced by a neural network, as in the end-to-end network developed by Sigtia et al. [8] for piano music transcription. Although good results can be achieved, this method is still inferior to state-of-the-art CNNs based on spectrograms as input data [11].

Böck and Schedl [6] applied semitone filterbanks to spectrograms which were then fed into Recurrent Neural Networks (RNNs) to achieve polyphonic piano note transcription. In 2014 Schlüter and Böck [21, 22] further improved general music onset detection with a CNN trained on mel-scaled spectrograms. A different approach by Thome and Ahlbäck [23] utilized a Convolutional Recurrent Neural Network (CRNN) trained on constant-Q transform (CQT) excerpts, intended as an online polyphonic pitch detection system. Similarly, Li [7] adopted CQT to a feed-forward Neural Network (NN) with two hidden layers as a polyphonic piano onset transcription system.

In an important work, Kelz et al. [9] were able to demonstrate the importance of choosing a proper representation of the input data and tuning of the learning rate for transcription systems based on deep neural networks. This was done by analyzing the impact of spectrograms with linearly spaced bins, spectrograms with logarithmically spaced bins, spectrograms with logarithmically spaced bins and logarithmically scaled magnitude, and the CQT, on the performance of a shallow net. Their investigation was carried out on the widely used MIDI aligned piano sounds dataset (MAPS) [24].

Kelz and Widmer [25] further investigate frame-wise transcription and were able to identify errors that common deep neural transcription systems make, which lead to the so-called glass ceiling effect [25] - where improvements have only marginal impact on the overall performance of the system. Their analysis of a CNN shows that, when the training data incorporates examples with concurrent notes being played, neural networks suffer from the entanglement problem.

The newest and most promising development in polyphonic piano transcription directly builds upon the findings of Kelz et al. and was suggested by Hawthorne et al., with a model on the brink of enabling downstream applications such as symbolic MIR and automatic music generation [4]. The team of Google Brain concludes that future work on piano transcription systems will need to adopt to more strict evaluation metrics in order to compare the musically relevant performance.

1.2 Objective and Research Questions

Recent research has shown promising results on using simple frame-wise approaches for piano transcription [9, 25], which have then been adapted to be used as a feature extraction stage in a more complex transcription system [4]. However, the underlying CNN architecture has not been changed and the application of new system design strategies leaves room for improvement.

Therefore, this thesis will investigate frame-wise polyphonic piano transcription with deep learning methods by designing a convolutional neural network using deep residual learning. This architecture is then extended by employing bidirectional long-short-term memory in a multitask learning strategy. It is expected that the application of this architecture will further improve state-of-the-art piano transcription, possibly enabling more abstract MIR applications downstream. Comparability of the results is ensured through the usage of the MAPS dataset, which is well established in AMT research.

By also evaluating two different input preprocessing techniques, one of which is used for the first time in the context of piano transcription, this thesis also aims at providing a best practice guideline for the preprocessing step.
The stated objective leads to the following questions:

1. *Does the adoption of deep residual learning to frame-wise piano transcription further improve frame-level and note-level metrics?*

2. *What effect do the associated complementary methods have on creating perceptually relevant piano transcriptions from audio recordings?*

This thesis will address the questions by conducting a series of experiments which iteratively add the aforementioned methods. Additionally, the reimplemented CNN by Kelz et al. [9] will be evaluated and extended to note-level metrics. The different performance metrics are used to compare the developed transcription systems against each other and the related systems of recent publications. By further visualizing individual transcription examples the perceptual quality of the best performing system is

examined and compared to the state-of-the-art Onset and Frames system by Hawthorne et al. [4].

The following chapters are divided into three parts. Chapter 2 highlights the methods used in conducting the experiments and the treatment of the MAPS dataset. After motivating and documenting the experimental tools, Chapter 3 presents the specific setup of each experiment and the corresponding results. The discussion in Chapter 4 compares the findings with other studies, assesses how well the transcription system will generalize and reviews the strengths and weaknesses of the chosen approach, before the questions raised in the introduction are answered.

Chapter 2

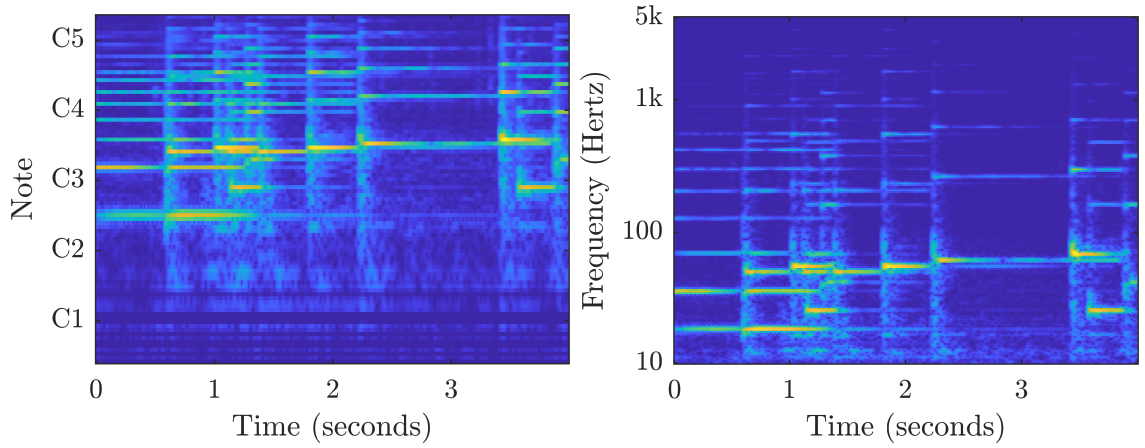
Method

The aim of this chapter is to outline the steps taken in arriving at the results presented in Chapter 3. The findings of this thesis heavily depend on methods developed over the past few years, therefore, it is important to showcase the validity and reliability of those insights. More so than in other deep learning tasks, input representation matters (see Section 2.1), since the network architecture operates on filtered spectrograms and not on raw audio data. In Section 2.2 a brief introduction to the MAPS dataset, followed by a description of the commonly used performance measures, will help to rank the results in the context of ongoing research. The key deep learning methods used in this thesis are presented in Section 2.3, highlighting the residual network architecture as a new adoption to the task of frame-wise piano transcription.

2.1 Input Features

Two different input representations are used in this thesis, log-magnitude semitone filtered spectrograms and a self-developed extension to Harmonic Pitch Class Profiles (HPCPs) [26] dubbed Octave-wise Harmonic Pitch Class Profile (OHPCP). HPCPs are a form of enhanced chroma features and often used in the context of chord recognition systems [26]. Both input features are calculated using the *madmom* library [27] and are now described in more detail.

The *madmom* library allows to develop efficient processing pipelines which can be serialized, saved and re-run, in order to ensure the reproducibility of the results [27]. Furthermore, the library provides a convenient way to directly load audio data and slice it into (overlapping) frames. Therefore, the spectrogram preprocessing can be summarized as follows: the discretely sampled audio input signal is sliced into frames and transformed to the frequency domain by STFT. Hereafter, the logarithmic



(a) OHPCP features **(b)** spectrogram features

Fig. 2.1 The two different input representations used to preprocess the audio data. Fig. 2.1a shows the OHPCP with 88 frequency bins and Fig. 2.1b depicts the log-magnitude semitone filtered spectrogram with 199 frequency bins.

magnitude spectrograms are filtered with a semitone filterbank using triangular unit area filters. The same frame resolution of 100 fps was used during the preprocessing stage for the spectrogram and OHPCP computation.

To the best knowledge of the author, HPCPs have not yet been used in the context of polyphonic piano note transcription or similar topics. This may be partly because HPCPs by definition reduce audio information to the twelve pitch classes of the diatonic scale. The reduction is accomplished by mapping each frequency bin to a pitch class in addition to applying a frequency weighting and considering the presence of harmonics [26]. However, by using bandpass filtered spectrograms to calculate HPCPs, an octave-wise representation can be generated. Stacking HPCPs calculated per octave on top of each other leads to the OHPCP feature, which is presented against a spectrogram in Fig. 2.1. Interestingly, OHPCPs closely resemble the note activation function outputted by the piano transcription model, since each bin of the OHPCP already corresponds to one of the 88 notes. However, due to the poor resolution in the lower frequencies it is likely that some notes are only represented by their higher order harmonics. In some cases the lower keys even produce nearly empty bins as can be observed in Fig. 2.1a.

Reducing spectrograms to the standard 88 piano keys or semitone spacing may have several advantages over a more granular frequency representation. In particular, the lower resolution of the frequency dimension can result in fewer parameters used in the deep learning model, hence leading to a reduced model complexity. Furthermore,

it desensitizes the system against learning the timbre or tuning variations of different piano instruments, hence leading to a better generalization.

2.2 Dataset and Metrics

This section clarifies the origin and composition of the data set used in this thesis, followed by a short introduction to the generation of frame-wise pitch occurrences, which motivates the performance metrics used to evaluate the deep learning models.

2.2.1 MAPS

All experiments are conducted using the MAPS dataset created by Emiya et al. [24]. The dataset consists of 270 MIDI aligned classical piano pieces, 30 of which are real musical piano pieces recorded from a Yamaha Disklavier upright piano. The remaining pieces are software synthesized from high quality piano sample patches. The Yamaha Disklavier piano is controlled by a MIDI signal and therefore self-playing during the audio recording. Arguably, it would improve the validity of the results if the recordings where performed by a human pianist. On the contrary, since no human pianist was involved in the creation of this dataset, a high level of accuracy can be guaranteed in the alignment of audio and MIDI annotations. However, as [4] points out, the Disklavier fails to correctly play a note if its MIDI velocity drops beneath a certain threshold. Moreover, Ewert et al. [28] found some cases where audio-midi alignment errors were up to 100 ms. For the sake of comparability with other publications, this label-noise will not be addressed any further.

In recent publications [4, 8, 9], a common train-test division has been established on the MAPS dataset. The exact four train-test folds are taken from [9], which were published online[1] and were designed in turn using the methods established by [8], commonly referred to as *configuration II*. Since training takes place only on synthesized audio and performance comparison solely on recorded audio, this configuration is presumed to be the most realistic setting. Furthermore, the reproducibility and comparability of the results is improved by adapting to this widely used training scheme.

[1] https://github.com/rainerkelz/ICASSP18/tree/master/splits/sigtia-conf2-splits, last visited: Tuesday 14th May, 2019

2.2.2 Background: Pitch Occurrences

The output layer of most frame-wise piano transcription models is governed by the sigmoid nonlinearity, mapped to the 88 notes of a typical modern piano, as showcased in Fig. 2.2. The logistic sigmoid function $\sigma(z)$ is defined as

$$\sigma(z) = \frac{1}{1 + \exp(-z)} \tag{2.1}$$

and its main property is to map the input to the open interval $(0, 1)$. The usage of the sigmoid function is justified by the binary nature of pitch occurrences, where class 1 describes an active pitch and class 0 an inactive one. It is therefore only necessary to determine the probability of one of the classes since the sum over the two probabilities has to add up to one. Furthermore, the mean of a distribution over a binary variable has to be between 0 and 1 [10], which again fits the property of the sigmoid nonlinearity.

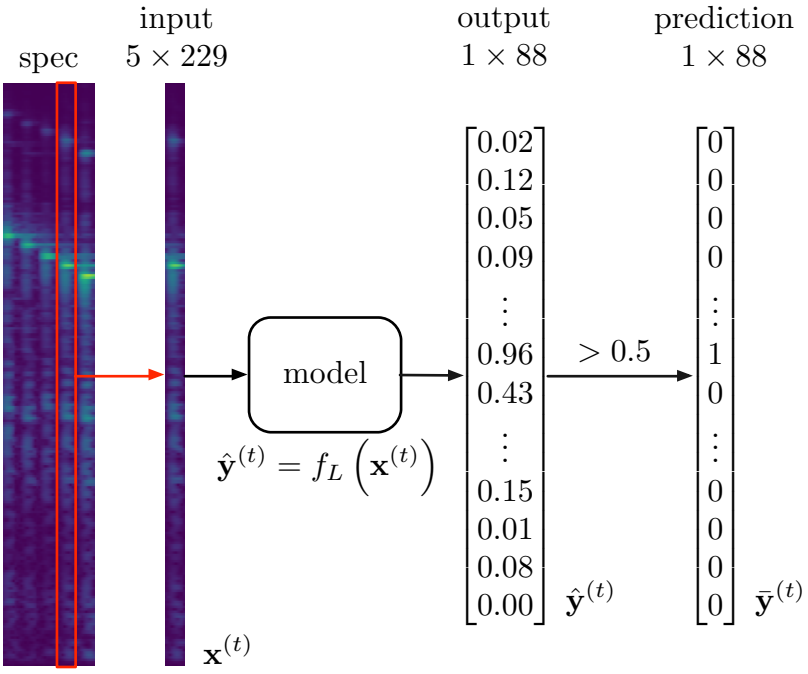


Fig. 2.2 A generic frame-wise piano transcription model. The spectrogram is sliced into overlapping excerpts $\mathbf{x}^{(t)}$ centered on the frame of interest and applied to the model $f_L(\mathbf{x}^{(t)})$. The sigmoid nonlinearity of the model produces the output $\hat{\mathbf{y}}^{(t)}$, which is then thresholded to obtain a binary prediction $\bar{\mathbf{y}}^{(t)} = \hat{\mathbf{y}}^{(t)} > 0.5$.

Hence, the output layer is able to present concurrently played notes as independent, sigmoid activated occurrences, which are then interpreted as probability values in

the closed interval $[0, 1]$. However, this kind of models are heavily impacted by the entanglement problem and therefore independence is not guaranteed [25]. In other words, the note C might be classified correctly when played concurrently with the notes of Cmaj, whereas it could be missed when played alone. In general it is desirable to build a model with perfect disentanglement, since this model would immediately generalize to all piano genres and would not need to implicitly learn the music theoretical patterns of the training data. However, since music theory is build on an extensive set of rules which translate to most music genres, even a model which suffers from entanglement may generalize well enough.

The general process of frame-wise piano transcription is depicted in Fig. 2.2. During training each spectrogram excerpt is centered on the frame to be classified, surrounded by an even number of context frames. The model, once trained, can be exposed to a succession of maximally overlapping spectrogram excerpts of a recording to obtain a piano-roll like representation from the sigmoid activation function. This is true for models which use convolutional layers with a neural network as a final layer, since CNNs are not explicitly designed to handle temporal information. Therefore, it has to be added artificially by giving a fixed temporal context of frames surrounding the frame being classified. In contrast, recurrent neural networks are able to operate on non-overlapping spectrogram excerpts of several seconds of audio data.

In the presented generic model all pitch occurrences are assumed to be of equal importance. However, in reality some pitch occurrences have more value for a successful transcription than others, as will be discussed later on. The two most important pitch occurrences are the onset and offset, when considering a note sliced into frames. In the context of this thesis, a note onset is the pitch occurrence which has no preceding occurrence of the same pitch, unless it is an offset and the offset has no following pitch occurrence unless it is an onset. This simplified definition of the onset and offset is similar to the one used by [4]. However, their findings suggest that a note onset should stretch over the first two frames of a note.

2.2.3 Metrics

Three different metrics will be applied to the predictions of the frame-wise piano transcription model, which in each case are expressed through Precision (P), Recall (R) and F1-Score (F1).

A simple frame-wise measure, as adopted from [9], is described in equations (2.2) through (2.4). The occurrences of true positives (TP), false positives (FP) and false negatives (FN) are counted on a per frame level t over all 88 MIDI pitches. True

negative (TN) occurrences are neglected for the obvious reason that the total number of TNs would shadow the important TP/TN count, therefore accuracy will not be applied as a performance measure.

$$\mathrm{P} = \sum_{t=0}^{T-1} \frac{\mathrm{TP}[t]}{\mathrm{TP}[t] + \mathrm{FP}[t]} \tag{2.2}$$

$$\mathrm{R} = \sum_{t=0}^{T-1} \frac{\mathrm{TP}[t]}{\mathrm{TP}[t] + \mathrm{FN}[t]} \tag{2.3}$$

$$\mathrm{F1} = 2 \cdot \frac{\mathrm{P} \cdot \mathrm{R}}{\mathrm{P} + \mathrm{R}} \tag{2.4}$$

The following two metrics are implemented using the *mir_eval* library [29], which employs a note-wise performance measure. The measure, that only considers the correctly predicted onset of a note, will be referred to as *note*, while the measure considering onset and offset of a note, will be referred to as *note with offset*.

The *mir_eval* library uses a bipartite graph to find note matches between predictions and ground truth. Since the graph operates on a list of notes described in Hertz with a start and stop time in seconds, it is necessary to transform the frame-wise predictions of the transcription model back to this domain.

The onset of a note is considered correct if it is within a ± 50 ms tolerance of the reference note with a pitch within ± 50 cent (quarter tone) of the corresponding reference note. The note with offset measure defines on top of the above criteria, that the note offset needs to be within 20 % of the reference notes duration around the reference notes offset, or within 50 ms (whichever is larger) [29]. Thus, a reference note exceeding a duration of 250 ms will always apply the upper bound of ± 20 % of the reference notes duration. These settings are common ground for many MIR transcription tasks and are adopted from the 2015 MIREX multiple fundamental frequency estimation and tracking, note tracking subtask.

The note-wise metric can also be used to incorporate a measure for note velocity estimation, as has been done by Hawthorne et al. [4]. However, in this thesis the focus is on note detection with and without offset. Again, Hawthorne et al. also illustrates how the note-level metrics are perceptually more relevant than frame-level metrics and encourages to emphasize on these when comparing transcription models.

2.3 Neural Network Architecture

This section describes the advantages and complementary capabilities of the three architecture types: residual network, bidirectional long-short-term memory and neural networks, as well as the method of combining these approaches into a unified architecture for piano transcription.

2.3.1 Deep Residual Learning

The foundation to ResNets are convolutional neural networks [30], which are a form of neural networks designed to operate on grid-like input data. As mentioned in Section 1.1, the most prominent application of CNNs is within the area of image processing, but since spectrograms can be thought of as a 2-D grid of pixels, many approaches have been transferred to the domain of audio processing.

As the name suggests, the central aspect of a convolutional network is the mathematical operation called *convolution*, which is denoted with an asterisk:

$$f(t) = (x * w)(t). \tag{2.5}$$

In the terminology of CNNs the function x is referred to as the *input*, and the second function w as the *kernel*. The output f is known as the resulting *feature map* at index t. The equation above describes the convolutional operation in 1-D. However, in machine learning applications the input and kernel are usually a multidimensional array of which the kernel is iteratively adapted by the learning algorithm.

In comparison to a CNN, a simple neural network learns a matrix of parameters potentially connecting each input value to each output value through the usage of a matrix multiplication. On the contrary, convolutional layers have a property called *sparse interactions*, which emphasizes on local connectivity by making the kernel smaller than the input data. This aspect is illustrated in Fig. 2.3 and is one of the important ideas behind CNNs, next to *parameter sharing* and *equivariant representations*, which are explained in more detail by Goodfellow [10]. However, since CNNs may have several convolutional layers, deeper layers will have a larger receptive field than the kernel size would suggest, which allows for complex interactions between local features even if they are far apart in the input data.

Very deep CNNs are difficult to train because of vanishing or exploding gradients, due to gradient values less than or greater than one being multiplied many times over through the layers of the network. This opposes the theoretically possible reduction of

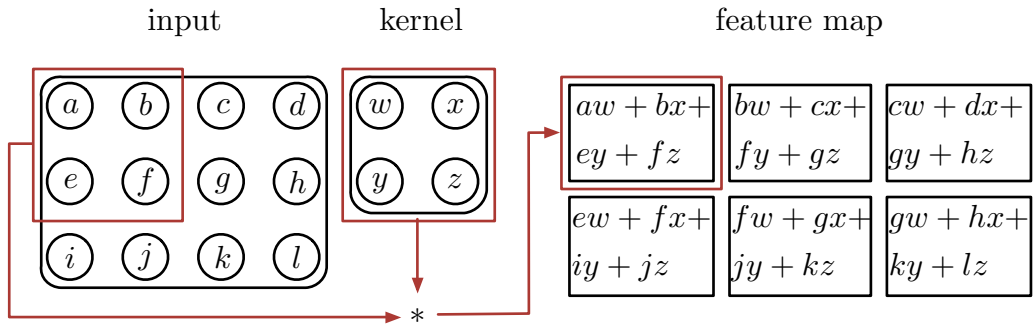


Fig. 2.3 A 2-D convolution performed without kernel flipping [10]. The feature map is computed by taking an equally sized subset of the input and convolving it with the kernel. This process is repeated by shifting the red window over the input data from left to right and top to bottom. In this example, the convolution produces a *valid* feature map, since the subset always lies entirely within the input data. In contrast to a *same* sized feature map, which is accomplished by zero-padding the boundary of the input.

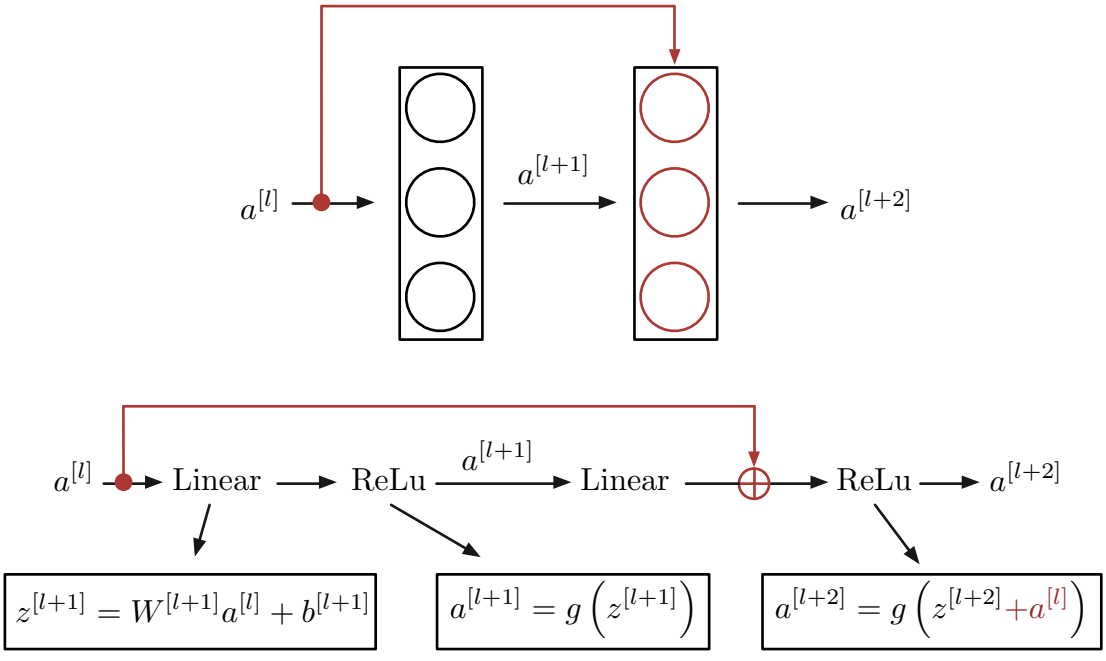


Fig. 2.4 Composition of a residual block.

the networks error rate with ever increasing depth. One way to address this problem is to use skip-connections, which allows to take the activation from one layer and feed it into a layer much deeper in the network. These ResNets make it possible to train very deep neural networks, sometimes with up to 1000 layers [31].

ResNets are build from so-called residual blocks, as depicted in Fig. 2.4. Like in a normal neural network the activation $a^{[l]}$ is passed through the layers on a main path till it becomes $a^{[l+2]}$. The steps of this computation are governed by a linear operator $z^{[l+1]}$, composed of the weights $W^{[l+1]}$ and bias $b^{[l+2]}$, followed by the rectifier linear unit $g(z^{[l+1]})$. In residual networks an additional skip-connection takes the activation $a^{[l]}$ and injects it in the second layer, right before the rectifier linear unit. Thus, skipping the first layer and retaining the information of the first activation in $a^{[l+2]}$.

Stacking multiple residual blocks on top of each other leads to the architecture of a deep residual network. The modular design of a residual block makes it easy to use them as a drop-in replacement for an ordinary convolutional layer. Therefore, it is an obvious step to use an already successful CNN architecture on the task of frame-wise piano transcription and replace its convolutional layers with residual blocks. At the time of this writing, the most effective convolutional architecture was proposed by Kelz et al. [9], a model with three convolutional layers followed by a dense neural network. By replacing the second and third convolutional layer with residual blocks, one obtains the ResNet architecture used throughout this thesis.

A more precise comparison of the composition of this two architectures can be found in Table 2.1. Note that despite the increased complexity of the ResNet, the total number of parameters only increases by five percent. Since this measure is proportional to the computational cost, it is reasonable to assume that the new model will benchmark similarly in this regard. Both models integrate commonly used best-practice techniques, such as max-pooling, dropout and batch normalization, the latter also being used inside a residual block. The following description summarizes the techniques mentioned above and points to the literature for further studies:

max-pooling Similar to the convolutional layer this operation applies a window with a stride to the input, with the difference that only the maximum value is passed to the output. The objective is to make the input units invariant to small translations and to down-sample the data representation [10].

dropout This technique randomly mutes units by multiplying them with zero and presents a computationally inexpensive regularization method. The effect is similar to training a bagged ensemble of exponentially many neural networks [32].

Table 2.1 Side-by-side comparison of convolutional and residual architecture. Note, instead of using the convolutional layers to reduce the frame dimension, the ResNet architecture uses the pooling layers for this task.

(a) The CNN architecture by Kelz et al. [9].

Layer Type	Output Dimensions	No. of Parameters
Input	1x5x229	
Conv	**32x5x229@3x3**	288
BatchNorm	32x5x229	128
ReLu	32x5x229	
Conv	**32x3x227@3x3**	9216
BatchNorm	32x3x227	128
ReLu	32x3x227	
MaxPool	**32x3x113@1x2**	
Dropout, $p = 0.25$	32x3x113	
Conv	**64x1x111@3x3**	18432
BatchNorm	64x1x111	256
ReLu	64x1x111	
MaxPool	**64x1x55@1x2**	
Dropout, $p = 0.25$	64x1x55	
Dense	512	1802240
BatchNorm	512	2048
ReLu	512	
Dropout, $p = 0.5$	512	
Dense	88	45144
Sigmoid	88	
		$\sum$ 1877880

(b) The ResNet architecture.

Layer Type	Output Dimensions	No. of Parameters
Input	1x5x229	
Conv	**32x5x229@3x3**	288
ReLu	32x5x229	
ResidualBlock	**32x5x229@3x3**	18688
ReLu	32x5x229	
MaxPool	**32x3x114@3x2**	
Dropout, $p = 0.25$	32x3x114	
ResidualBlock	**64x3x114@3x3**	37376
ReLu	64x3x114	
MaxPool	**64x1x57@3x2**	
Dropout, $p = 0.25$	64x1x57	
Dense	512	1867776
BatchNorm	512	2048
ReLu	512	
Dropout, $p = 0.5$	512	
Dense	88	45144
Sigmoid	88	
		$\sum$ 1971320

batch normalization This method provides a way of adaptive re-parametrization during the back-propagation step, preventing the gradients from proposing an operation which would only increase the standard deviation or mean of the input units [33].

ReLu The rectified linear unit applies a piecewise linear operation $f(x) = \max\{0, x\}$ to the output of the convolutional layer. The usage of this activation function is motivated from biological neurons and has shown better results than functions with a two-sided saturation when applied to intermediate network layers [34].

2.3.2 Bidirectional Long-Short-Term Memory

While CNNs are designed to operate on grid-like input data, recurrent neural networks were invented to process sequential data. Hence, this property is particularly interesting for audio processing applications, since audio data is a paradigmatically sequential signal. And although CNNs are able to process a small temporal context, as mentioned in Section 2.2.2, many challenges in AMT can be addressed by accessing spectrogram data which lies further in the past than would be practicable for a CNN. RNNs are able to share information across very long sequences by using cyclical connections, thereby memorizing previous inputs in their internal state [35]. Furthermore, this idea can be extended to also incorporate future inputs from a sequence.

The invention of bidirectional recurrent neural networks by Schuster and Paliwal [36] addresses the need for solving ambiguities, which arise from only looking into past values of a sequence. For example, when considering speech recognition, the identification of the present sound as a phoneme could depend on the following or previous phonemes because of co-articulation. Similarly, extended to piano transcription, the correct interpretation of a note onset may depend on how the note reverberates over the following few time steps or may even depend on the next few notes because of music theoretical dependencies between them.

As the name suggests, Bidirectional Long-Short-Term Memory (BiLSTM) combine an LSTM network that moves forward through time and another LSTM network which moves backwards through time. This kind of network architecture is able to concurrently uncover a time series from the start and the end, thus, making it possible to output a prediction of $\mathbf{y}^{(t)}$ that may depend on the whole input sequence $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(t)}$. Nevertheless, the output units are most susceptible to input values near t, but without the necessity to define a fixed-size time window around t. This is one of the key advantage of a BiLSTM network over a regular RNN and especially over a pure residual

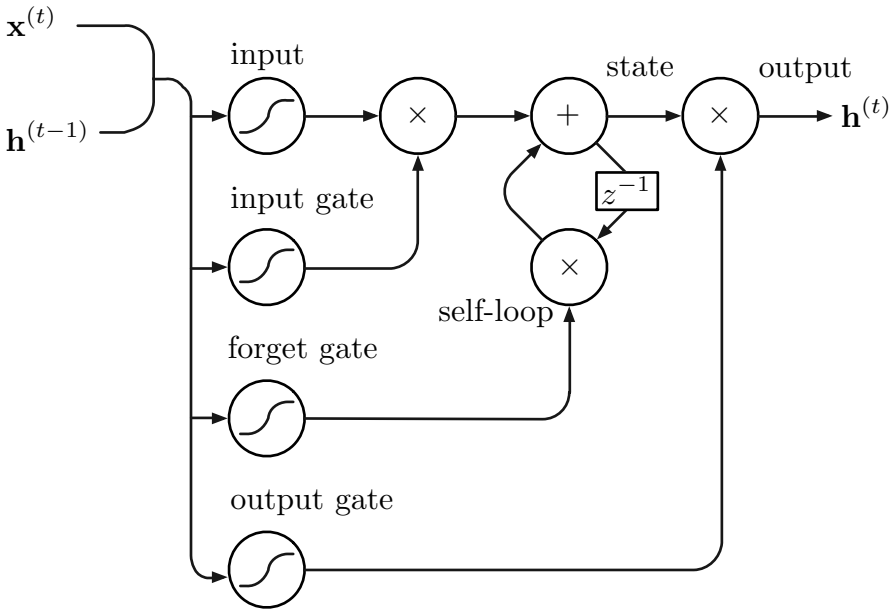


Fig. 2.5 Block diagram of an LSTM cell.

network as described in section 2.3.1, which requires a fixed context window to operate on.

Fig. 2.5 highlights the data flow through one cell of an LSTM recurrent network. The concatenated values of the input $\mathbf{x}^{(t)}$ and the output $\mathbf{h}^{(t-1)}$ of the previous LSTM cell are broadcast to the sigmoid activated input, forget and output gate, which act as kill switches for any elements of the input vector that are not required. The gate units always have a sigmoid nonlinearity while the input unit can be used with any compressing nonlinearity. The state unit and self-loop form a layer of recurrence which is controlled by the forget gate, thus, telling the network which state variables to remember and which to forget. Similar to the idea of ResNets, the delayed state $\mathbf{s}^{(t-1)}$ is added to $\mathbf{s}^{(t)}$ in order to prevent vanishing gradients. Eventually, the output $\mathbf{h}^{(t)}$ of the LSTM cell can be shut off by the output gate.

Instead of standard neural network layers the LSTM network uses a succession of cell blocks to form a recurrent neural network. Subsequently, by stacking two LSTM networks with opposing recurrence directions on top of each other, a network with a single output unit that benefits from a summary of future and past time steps is constructed.

2.3.3 Multitask Learning

The input representations described in section 2.2 would be directly applicable to a BiLSTM network, since the spectrogram of a piano piece is a valid time series. However, recent research has shown that using a convolutional network as a feature extraction stage provides exceedingly better results across tasks like speech recognition [37], language models [38] and piano transcription [4]. The combination of these two network architectures seems to come naturally to audio related tasks, since convolutional networks are very good at reducing frequency variations in spectrograms, while LSTMs have been proven to learn long, time-dependent representations better than simple recurrent networks [10]. Fully connected neural networks are an often used component of deep learning architectures, and well known for their ability to separate the input space into an easy to classify output [37]. Hence, the target architecture of this thesis incorporates all three machine learning architecture types, governed by a multitask learning paradigm.

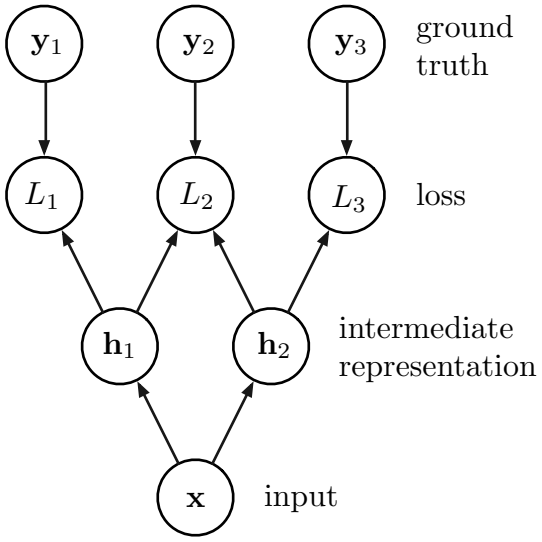


Fig. 2.6 Abstract view of the multitask learning strategy used in the context of this thesis. It is assumed that the intermediate representation of $\mathbf{h}_1$ is more specialized towards the task of predicting $\mathbf{y}_1$, while $\mathbf{h}_2$ is specialized towards $\mathbf{y}_3$. Predicting a combined representation $\mathbf{y}_2$ as well, emphasizes on the common pool of factors extracted from $\mathbf{x}$.

In the context of deep learning, the multitask strategy is one of several regularization methods. It can be described as an prior assumption, in which some of the variance explained by observations made in the data are shared across multiple tasks. This strategy can be imposed onto a deep learning architecture by learning tasks in parallel

on a shared data representation, while exchanging features at a later point in the network [39]. Fig. 2.6 illustrates the strategy used in this thesis. The loss of three different prediction tasks, based upon two independent intermediate representations, is used to train the model for frame-wise piano transcription. The three tasks correspond to predicting the frame-wise note occurrences, the note onset frame and the combination of the previous two. The separation of the underlying global task of piano transcription is well justified, since note onset detection is very similar to detecting any other part of a note (see Section 2.2.2). Therefore, it is reasonable to assume that these two different forms of pitch occurrences share common features.

Multitask learning has a similar effect on deep learning models as increased training data has, insomuch as it helps the model to generalize better to unseen data [10]. The intuition behind this improvement can be understood as follows: Multiple tasks use the same intermediate representation of the model, thus, imposing more constraints onto the representation and therefore leading towards common factors which hold meaning for each of the tasks.

Chapter 3

Experimental Setup and Results

Four different new piano transcription models where trained on the MAPS dataset, as well as a reimplementation of the CNN proposed by Kelz et al. [9]. The models are compared to recent publications and in particular to the Onset and Frames system by Hawthorne et al. [4], forming the state-of-the-art at the time of these experiments. Starting with the initial ResNet system, several modifications are performed to both architecture and training, yielding further improvements. These are reported on in Section 3.2 after explaining the experimental setup (Section 3.1) and the details of the transcription system (Section 3.1.1) and its postprocessing stage.

3.1 Experimental Setup

Different preprocessing settings have been applied on conducting the experiments, the final setups are concisely summarized in Table 3.1. Setup I was taken from [25] and used to train the reimplemented model described in [9], both publications build on very similar CNNs and it was necessary to cross-reference implementation details. This was done to ensure evaluation consistency and to test the reproducibility of the results generated by the CNN architecture and will be discussed further in Chapter 4. Setup II is derived from I and informed by the observation of many empty frequency bins above 5 kHz. Theoretically these bins could be occupied by note harmonics and therefore hold information useful for the transcription model. However, preliminary experiments showed no or very little impact on the overall F1-Score. Therefore, the upper frequency limit was lowered in favor of fewer frequency bins.

Setup II was used for all experiments involving a ResNet based architecture, except where noted differently. Setup III describes the preprocessing of Octave-wise HPCP, the frequency boundaries are taken from the lowest and highest octave. Any intermediate

octave frequency boundaries are calculated by taking the mean of the frequencies of the highest and lowest note in two consecutive octaves.

Table 3.1 The different setups used in the preprocessing stage.
Setup I: Kelz et al., reimplementation
Setup II: ResNet - ResNet & BiLSTM - Multitask ResNet & BiLSTM.
Setup III: ResNet w/ OHPCP.

	Setup I	Setup II	Setup III
input features	log. mag. semitone spec	log. mag. semitone spec	OHPCP
f_s	44.1 kHz	44.1 kHz	44.1 kHz
FFT size	4096	4096	12 288
frame size	4096	4096	4096
fps	100	100	100
freq. bands	48	48	12
$f_{\min}$	30 Hz	10 Hz	27.5 Hz
$f_{\max}$	8000 Hz	5000 Hz	6645 Hz
norm filters	yes	yes	no
unique filters	yes	yes	-
circular shift	no	no	no
freq. bins	229	199	88

Hyper-parameter optimization was conducted by a human expert by applying an informed grid search to the parameter selection of related models [8–10]. The summarized final parameterization is depicted in Table 3.2 for each of the trained models. In case of the convolutional models a linear learning rate schedule was applied by halving the learning rate every three epochs up until epoch 24, while training a total of 30 epochs. The models using a BiLSTM layer employ a exponential decay on the learning rate with a decay rate of 0.98, reducing it every 5000 steps. The choice of a batch size of 128 is within the margin of standard mini-batch stochastic gradient decent, as it was proposed by [25]. The drastic reduction in batch size with BiLSTM architectures is motivated by the results of [40], which state that small batch sizes converge to a flatter minimum due to the noise in the gradient decent. This is desirable since a flatter minimum is associated with better model generalization.

During training, the frame- and element-wise applied binary crossentropy [9]

$$\mathcal{L}^{(t)}\left(\mathbf{y}^{(t)}, \hat{\mathbf{y}}^{(t)}\right) = -\left(\mathbf{y}^{(t)} \cdot \log\left(\hat{\mathbf{y}}^{(t)}\right) + \left(1 - \mathbf{y}^{(t)}\right) \cdot \log\left(1 - \hat{\mathbf{y}}^{(t)}\right)\right) \tag{3.1}$$

is used on the networks output, with $\hat{\mathbf{y}}^{(t)}$ being the output of the network and $\mathbf{y}^{(t)}$ the ground truth of the training data at frame t. The global loss of one training step is calculated by taking the mean over a batch of training data, which is then passed to the optimization algorithm to be minimized. The CNN and ResNet architecture are regularized by clipping the networks output below $1e^{-7}$ and above $1 - 1e^{-7}$. The architectures with a BiLSTM layer employ a L_2-norm penalty term calculated from the connection weights and added to the cost function. Both approaches help to reduce overfitting and regularize the network.

All models where trained using the TensorFlow [41] framework. The architectures without BiLSTM layers were first trained on a GeForce 1060 (6 Gb), but due to the high memory demand of recurrent networks other models had to be trained on a P5000 GPU (16 Gb). Therefore, the simpler models were also retrained on the P5000 GPU in order to compare the computational cost. This hardware configuration allowed to train models involving BiLSTMs with a maximum batch size of 8.

Table 3.2 Final hyper-parameter choice used to train the models.

	Kelz et al., reimp.	ResNet	ResNet using BiLSTM
batch size	128	128	8
clip L2-norm	-	-	3
clip output	$1e^{-7}$	$1e^{-7}$	-
learning rate	0.1	0.1	0.0006
momentum	0.9	0.9	-
input format	channels first	channels first	channels last
optimizer	Nesterov, momentum	Nesterov, momentum	adam

3.1.1 Transcription System

Of the different model architectures developed in this thesis the transcription system depicted in Fig. 3.1 performed the highest on the note-level metrics. The transcription system is build upon the ResNet architecture presented in Fig. 2.1b, followed by an BiLSTM layer and a fully connected neural network with sigmoid activation. The system consists of two main paths which interact with each other through a third fully connected sigmoid layer. The right path represents the onset detector while the left path is trained on predicting general frame activations. The combined loss emphasizes on the similar properties between detecting pitch occurrences and onset occurrences.

The onset predictions are later used to restrict pitch occurrences in the frame-wise predictions. This step should be considered part of the postprocessing rather than part

of the deep learning model, since it does not influence the training process directly. In other words, the restriction by onset-frames is part of an aggregation function with a slightly different goal than the frame-wise calculated loss. While the model is trained on a frame-level logic with a bias for onset-frames, the aggregation function anticipates a note-level logic by assuming general pitch occurrences are only possible after an onset occurred. This means that any pitch occurrences are neglected if not preceded by an onset. Vice versa, if an elsewhere continuous stream of pitch activations, with an onset at the beginning, is interrupted by a single missing frame, all following frames are also neglected. While this approach is prone to produce more false negatives it also alleviates the transcription system of the issue with multiple note reactivations.

Furthermore, the importance of note onsets is emphasized by applying a note activation heuristic to each onset. This approach shows similarities to the findings of [4], which state that the prediction quality was improved by assuming an onset takes up two frames rather than one. However, the proposed heuristic aims at closing gaps of missing frames between the onset and following pitch occurrences, which would result in missing notes or multiple reactivations. The underlying assumption is as follows, if a note onset was detected, it will always be followed by a number of k_{heu} active frames of the same pitch. This heuristic is applied during the postprocessing stage and can be easily changed at a late point in time, since it is not part of the trained model. The value of this heuristic was determined empirically to perform best at $k_{\text{heu}} = 10$. This means, any given onset detection leads to the transcription of a note with a duration of at least 100 ms, independent of the true duration of the note in question. Naturally, this heuristic introduces errors of its own by eradicating any note actually played shorter, or by emphasizing false positive onset detections which otherwise would have been inaudible. On the up-side, this process ensures audible notes where the deep learning model was only able to identify the onset frame. This approach can be justified by the steep attack of piano notes right after the onset, making it difficult to detect the onset and the following three to five pitch occurrences correctly as compared to the frames where the maximum note energy is reached. This observation is illustrated in Fig. 3.2, almost all models reach the highest F1-Score within 30-50 ms after a note onset, followed by an often equally steep descent up to 100 ms. This second observation is most likely due to the energy decay in the piano note right after the initial excitation. The following findings suggest that these two observations explain the performance improvement of all considered models when applying the onset heuristic.

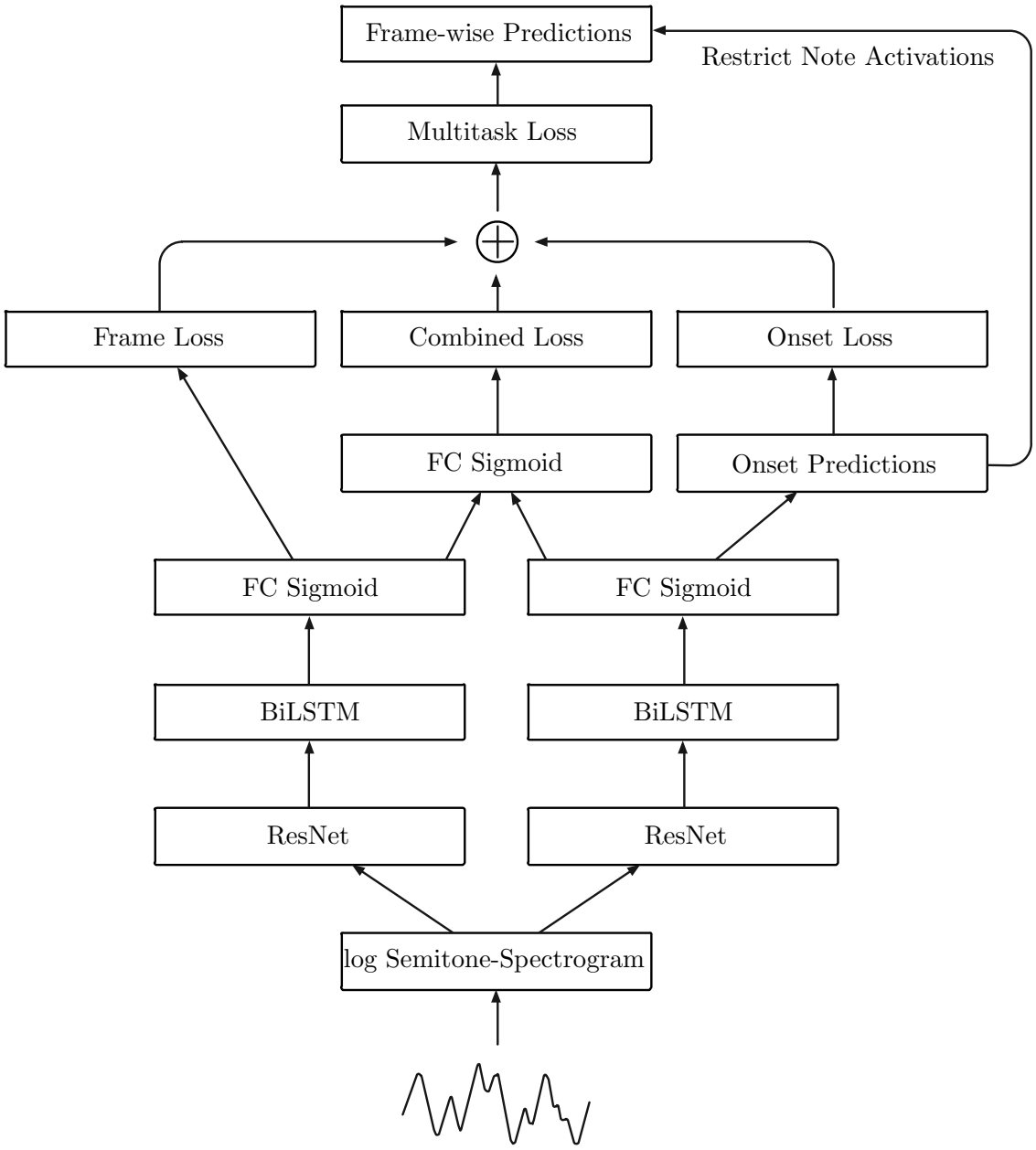


Fig. 3.1 Diagram of the final multitask transcription model architecture.

3.2 Experiments

The following Experiments are observed over different frame- and note-level metrics, divided by the model architecture and its preprocessing. The ResNet, ResNet with OHPCP and ResNet & BiLSTM models do not employ an onset detector themselves, therefore, the piano note onset detector developed by Böck et al. [6] was used to incorporate onset information during the postprocessing stage. The system applies a BiLSTM network and was also trained on the MAPS dataset, which makes it the ideal candidate to compare it against the concurrently trained onset and pitch occurrence detector of the Multitask ResNet & BiLSTM. The implementation of the onset detector by Böck et al. used during the experiments is part of the *madmom* library. However, the onset detection model was not trained according to configuration II of the MAPS dataset, but rather configuration I. This train pattern allows pieces of the recorded piano to be part of the training set and therefore has a data advantage over the Multitask ResNet & BiLSTM and other models which only use configuration II.

Like with the ResNet model, the ResNet & BiLSTM experiment uses Setup II for the preprocessing parametrization. Additionally, a further preprocessing step was applied. The log-magnitude semitone-filtered spectrograms are enhanced by adding their positive frame-wise and frequency-wise differences to the spectrogram, followed by normalization and clipping values $< 10^{-3}$. This step is motivated by the idea of the *superflux* onset detector designed by [42]. The algorithm applies trajectory tracking to the positive frame-wise differences of a spectrogram to detect onsets in music. Similarly, a convolutional architecture could be prompted to extract more relevant features from this input representation. From an image processing point of view this step is similar to enhancing the contrast of a picture. However, preliminary experiments showed a significant increase in frame-level F1-Score of 5 to 6 points and in note-level F1-Score of 4 to 5 points. Therefore, this improvement was applied in this experiment and its extension the Multitask ResNet & BiLSTM.

For the Multitask ResNet & BiLSTM experiment the transcription system depicted in Fig. 3.1 was used. The spectrogram enhancement described above was applied in addition to the preprocessing Setup II. Due to the multitask learning strategie, the architecture of this deep learning network is more complex than in the previous experiments. However, this system is still simpler than the model proposed by Hawthorne et al. as it only employs two parallel acoustic models, compared to three parallel strands used in the Onset and Frames system [4].

Frame-level results in Table 3.3 also depict the results of adding onset predictions and the onset heuristic. The overall effect of adding further postprocessing techniques

Table 3.3 Frame-wise results on the MAPS dataset, configuration II. Frame-wise scores calculated as defined in [8], the final measure is the mean of scores calculated per piece.

	Frame			Frame onset prediction			Frame onset prediction and heuristic		
	P	R	F1	P	R	F1	P	R	F1
Kelz et al., reimp.	75.15	65.08	69.06	74.74	65.41	69.07	72.18	68.99	69.81
ResNet	75.24	68.68	71.19	74.87	68.89	71.13	72.21	71.95	71.38
ResNet w/ OHPCP	74.19	58.17	64.70	73.78	58.47	64.75	71.39	63.35	66.56
ResNet & BiLSTM	75.50	70.25	72.54	75.14	70.39	72.45	72.64	74.55	73.28
Multitask ResNet & BiLSTM	**79.70**	**71.94**	**75.27**	**79.53**	**72.37**	**75.45**	**77.62**	**75.93**	**76.39**

to the transcription systems is modest, the ResNet w/ OHPCP benefits the most followed by the Multitask ResNet & BiLSTM. Altogether, the Multitask ResNet & BiLSTM achieves the highest scores on all frame-level metrics.

Table 3.4 Note with offset results on the MAPS dataset, configuration II. Note scores calculated with the *mir_eval* library [43], the final measure is the mean of scores calculated per piece.

	Note w/ offset			Note w/ offset onset prediction			Note w/ offset onset prediction and heuristic		
	P	R	F1	P	R	F1	P	R	F1
Kelz et al., reimp.	25.60	40.76	30.94	38.97	36.17	37.32	45.43	41.10	42.91
ResNet	22.66	45.58	29.80	42.48	39.54	40.73	47.62	43.44	45.18
ResNet w/ OHPCP	25.59	42.18	31.19	40.30	38.20	39.01	46.76	43.32	44.72
ResNet & BiLSTM	38.67	47.77	42.50	49.88	45.12	47.14	54.16	48.17	50.72
Multitask ResNet & BiLSTM	**41.61**	**50.23**	**45.29**	**55.57**	**49.54**	**52.20**	**59.03**	**51.78**	**55.22**

The note-level results are shown in Table 3.4. In contrast to the frame-level metrics the note-level metrics greatly improve through adding the additional postprocessing methods. The ResNet even improves its F1-Score by 15 points while the other models achieve an improvement of at least 10 points. The incorporation of onset predictions to the output of the transcription systems shows the most significant advance in F1-Scores. Again, the Multitask ResNet & BiLSTM achieves the best performance in all note-level metrics.

The computational cost, in time spent training each of the transcription models, is showcased in Table 3.5. The most efficient model in terms of performance gain to invested training time is the ResNet & BiLSTM, since the improvement of the Multitask ResNet & BiLSTM is facilitated by four times the training time of the baseline CNN architecture. Interestingly, the ResNet architecture only needs half as much training steps compared to the CNN by Kelz et al., while still slightly increasing the training time.

Table 3.5 Comparison of best model F1-Scores, training steps and computational cost. One training step equals one batch of training examples passing through the network. All models were trained using the same P5000 GPU.

	Frame F1-Score	Note w/ offset F1-Score	Train Steps	Train Time
Kelz et al., reimp.	69.81	42.91	1.5 M	6 h
ResNet	71.38	45.18	850 k	7 h 30 m
ResNet w/ OHPCP	66.56	44.72	850 k	7 h 30 m
ResNet & BiLSTM	73.28	50.72	40 k	11 h
Multitask ResNet & BiLSTM	76.39	55.22	45 k	24 h

The Fig. 3.2 illustrates the performance of each model to predict individual pitch occurrences. The longer a model can maintain a high F1-Score, the better its ability to uninterruptedly track the progression of a note. The results show that the ResNet architecture achieves the highest scores in this regard, while it is still able to maintain a high F1-Score over a long period when combined with the BiLSTM. Nevertheless, the ResNet architecture also shows the worst performance when trained on OHPCP instead of spectrograms.

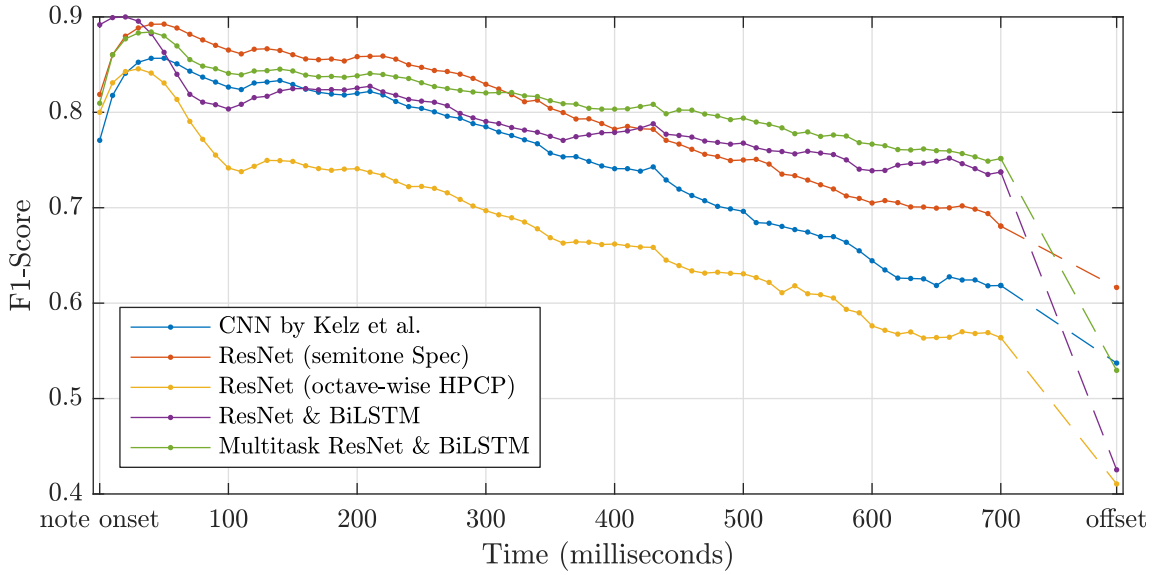


Fig. 3.2 Mean F1-Score of a piano note over time, without using onset predictions or onset heuristic. The score is calculated per frame as the mean over all notes in the test-set and interpolated linearly. Depicted are the onset frame, the first 70 frames (700 ms) and the offset frame. Note, the offset is added for better comparison, obviously an individual note offset might occur before or after 700 ms.

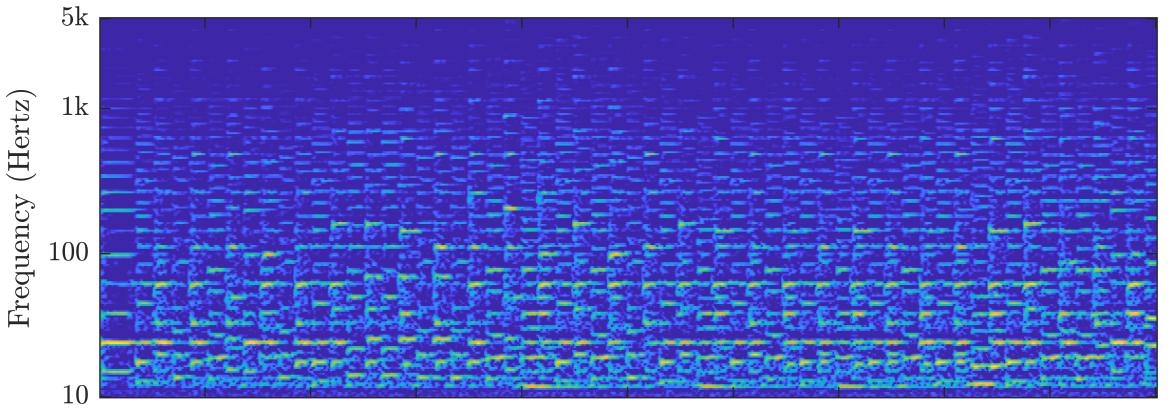
Fig. 3.3 illustrates the input spectrogram 3.3a to the Multitask ResNet & BiLSTM, the resulting note activation function 3.3b and the piano-roll transcription 3.3c generated from the postprocessing stage. The piano-roll is complemented by the corresponding ground truth which is indicated by the color coding: missed reference frames in yellow, falsely predicted frames in red and correctly estimated frames in green. Fig. 3.4 presents an exemplary comparison of the transcription system proposed by Hawthorne et al. [4] and the Multitask ResNet & BiLSTM. In case of the Onsets and Frames model the transcribed piano excerpt is generated by the online transcription service *piano scribe*[1], provided by Google Brain. Both piano-roll representations are quantized by the frame-wise temporal resolution of 10 ms, used by the transcription systems.

3.3 Extensions and Dead Ends

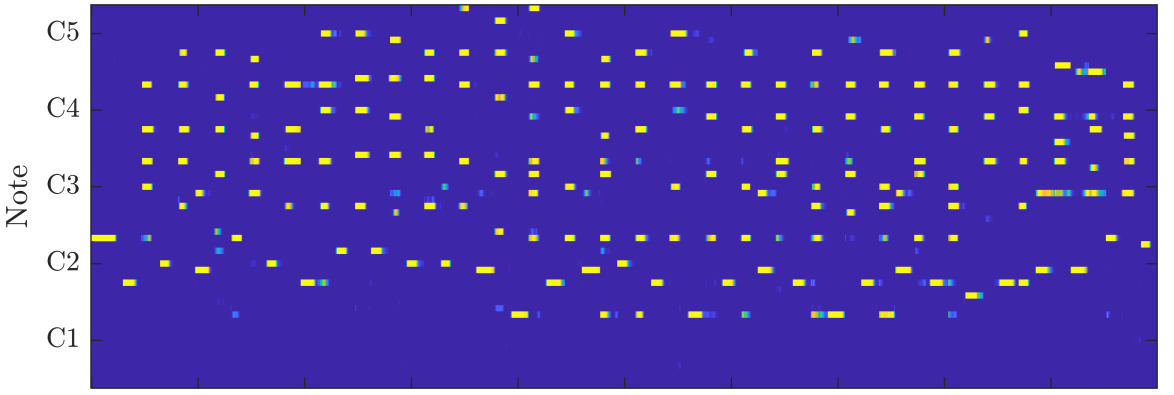
Motivated by the idea of the musical onset detector proposed by Schlüter et al. [22] an attempt was made to compute three semitone-filtered spectrograms using different frame lengths (23 ms, 46 ms and 93 ms, respectively). The spectrograms would be passed to the network as three channels, similar to the RGB channels of an image. This approach has shown good results with onset detection and works around the trade-off between frequency resolution and temporal resolution. The anticipated improvement for frame-wise piano transcription would be a clear localization of onsets in the short windowed spectrogram and a sharp distinction between frequency bins in the spectrogram with a long window. However, this did not improve results, but even deteriorated them. Possibly, the later adopted multitask learning could provide a new way to incorporate spectrograms with different frame lengths, for example by using the short windowed spectrogram only for the onset detector.

The Multitask ResNet & BiLSTM was used in an attempt to investigate how well the transcription system can generalize to other instruments. For this task, a dataset for guitar transcription was used called *GuitarSet* [44]. Surprisingly, the transcription system performed beyond expectations, even without any retraining of the networks layers. On the frame-level F1-Score the system scored 56.89 points while it still achieved 10.41 points on the note with offset F1-Score. This result gives hope that a future transcription system will also be applicable to different instruments and indicates possible approaches for transfer learning.

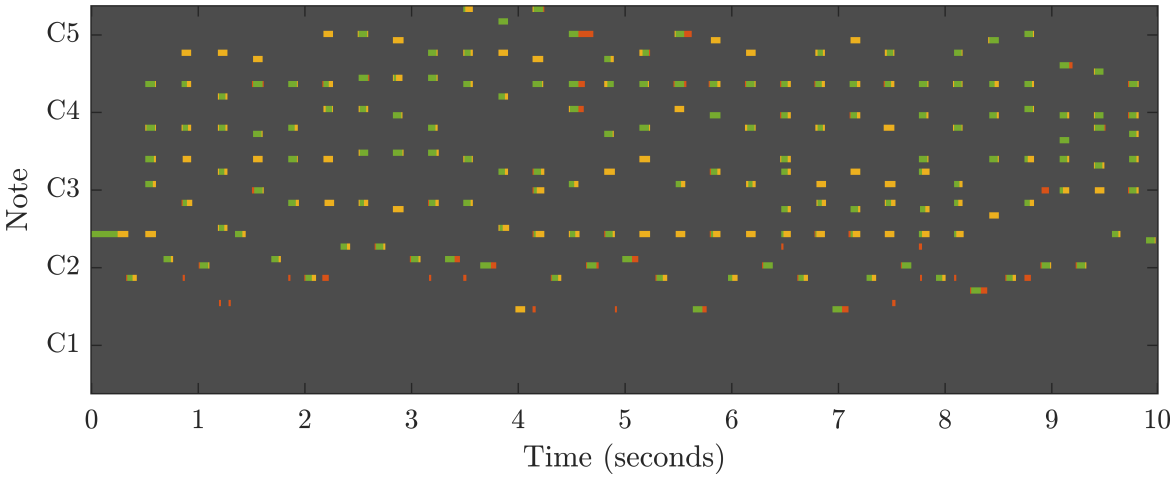
[1] https://piano-scribe.glitch.me, last visited: Tuesday 21st May, 2019



(a) Input Semitone Spectrogram

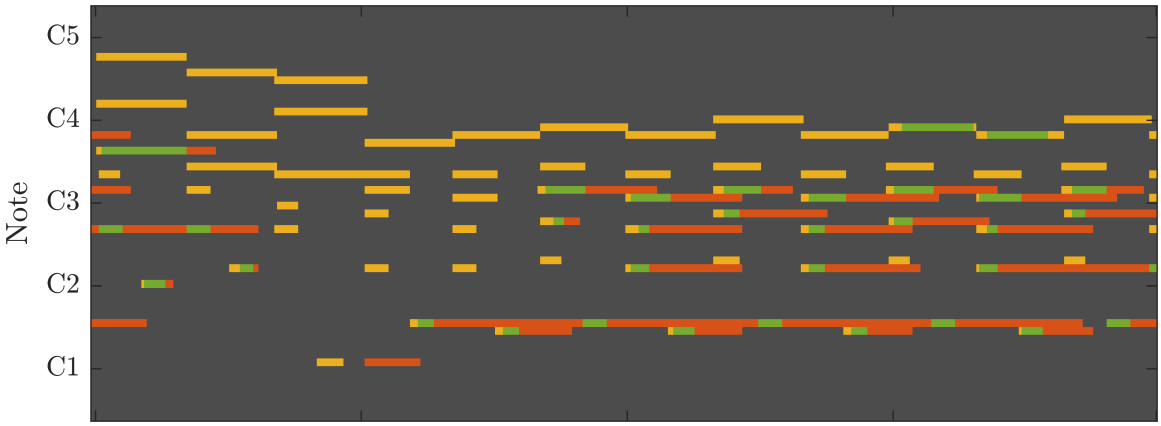


(b) Note Activation Function

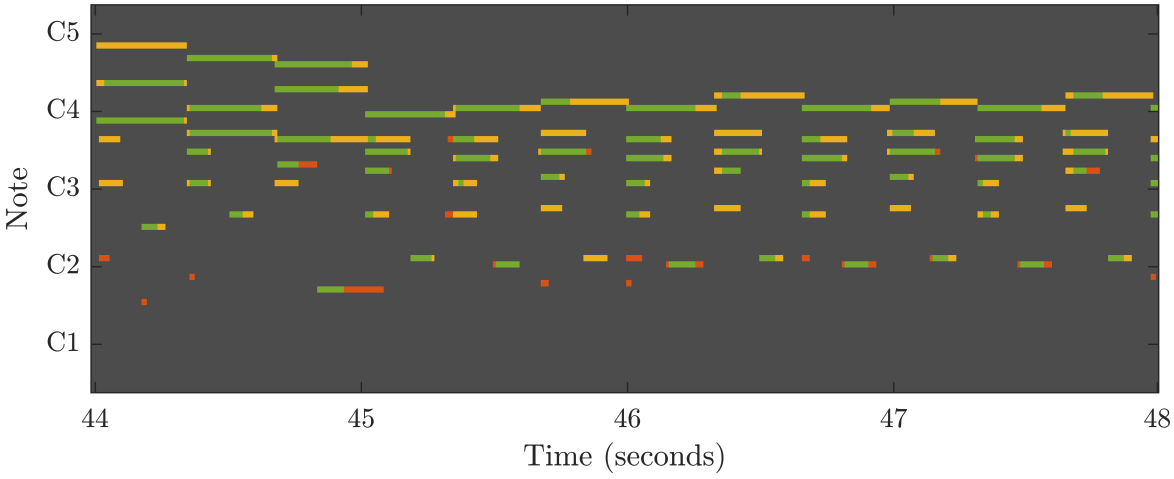


(c) Estimated and Reference Transcription

Fig. 3.3 Transcription of 10 seconds of MAPS_MUS-chpn_op25_e4_ENSTDkAm.wav, a recording which is not part of the training set. 3.3a the log-magnitude semitone filtered spectrogram. 3.3b the activation function from the Multitask ResNet & BiLSTM. 3.3c the estimated transcription restricted by onset predictions, with missed reference frames in yellow, falsely predicted frames in red and correctly estimated frames in green.



(a) Estimated and Reference Transcription from [4]



(b) Estimated and Reference Transcription from the Multitask ResNet & BiLSTM

Fig. 3.4 Comparison of the transcription quality between the Onset and Frames system and the Multitask ResNet & BILSTM on 4 s of MAPS_MUS-chpn_op25_e4_ENSTDkAm.wav. With missed reference frames in yellow, falsely predicted frames in red and correctly estimated frames in green.

Chapter 4

Discussion

The first part of this chapter discusses the specifics of the experiments and compares them to one another, as well as to recent publications. The second part will open up to a broader discussion which puts a focus on the Multitask ResNet & BiLSTM, its strong and weak aspects and directions for improvement. Finally, the conclusion will answer the questions posed in Chapter 1 and recommend steps for future research.

Through a combination of deep learning methods build upon the ResNet architecture it was possible to significantly advance the state-of-the-art in polyphonic piano transcription on the MAPS dataset. The experiments applied different preprocessing routines and where step by step extended from the baseline CNN architecture. This approach helped identifying complementary postprocessing techniques to supplement the final transcription system. Furthermore, comparability to other publications was ensured through the use of frame-level and note-level metrics, as well as by committing to the widely used configuration II of the MAPS data set.

Table 4.1 Comparison of highest score results on the MAPS dataset with recent publications.

	Frame			Note			Note w/ offset		
	P	R	F1	P	R	F1	P	R	F1
Sigtia et al., 2016 [8]	-	-	64.14	-	-	54.89	-	-	-
Kelz et al., 2016 [9]	74.50	67.10	70.60	-	-	-	-	-	-
Hawthorn et. al., 2018 [4]	**88.53**	70.89	**78.30**	**84.24**	**80.67**	**82.29**	51.32	49.31	50.22
ResNet	72.21	71.95	71.38	73.82	67.30	70.01	47.62	43.44	45.18
ResNet w/ OHPCP	71.39	63.35	66.56	73.73	68.69	70.72	46.76	43.32	44.72
ResNet & BiLSTM	72.64	74.55	73.28	74.72	66.23	69.85	54.16	48.17	50.72
Multitask ResNet & BiLSTM	77.62	**75.93**	76.39	81.18	70.94	75.43	**59.03**	**51.78**	**55.22**

Kelz et al. reimplementation The frame-level results of the publication have been replicated with sufficient accuracy, with a negative deviation of the F1-Score by

one percentage point. The slightly lower score stands in contrast to the results of the reimplementation by Hawthorne et al. [4], which state a slightly better score of plus one percentage point compared to the original findings. However, these small deviations are most likely due to differences in the implementation and do not dispute the results.

Unsurprisingly, the improvements to the postprocessing stage (onset detection, onset heuristic) have almost no influence on frame-level results, since the number of frames with a note onset is very small compared to the total number of frames in a song ($1\,\mathrm{s} \mathrel{\hat{=}} 100\,\mathrm{frames}$). Furthermore, when counting all pitch occurrences which have already been predicted correctly, the number of newly added pitch occurrences by the postprocessing stage is diminished even further. This result illustrates the fundamental problem with frame-level metrics, even though the exact beginning of a note in a melody is crucial for a successful transcription it is valued equally to any other frame-wise pitch occurrence of the melody.

On this metric the reimplemented model performed much better than the reimplementation by Hawthorne et al. [4], which states a Note with offset F1-Score of 23.14. At this point, it can only be assumed that the deviation is due to differences in the implementation and training scheme, since the data and note-level evaluation method are the same. However, the results show that the note-level metrics greatly benefit from adding onset predictions and onset heuristic, respectively.

ResNet The ResNet model is in direct comparison to the CNN by Kelz et al. [9], since it builds upon a similar architecture as described in Section 2.3.1. By examining Table 3.3, one can observe a slight overall improvement in frame-level metrics of about one point due to a better recall score of the ResNet. This could be explained by the use of skip connections in the ResNet. The inevitable energy decay of a note in the spectrogram may be compensated for by reinforcing deeper layers with previous ones and therefore keeping the note energy above the detection threshold, leading to a better frame-wise recall. This is also illustrated by Fig. 3.2, the ResNet and CNN scores are almost perfectly aligned over time, wherein the ResNet is continuously better with an offset of about five points.

The note-level metrics of the ResNet architecture in Table 3.4 demonstrate considerably better recall scores but a slightly worse precision score compared to the CNN. Hence, the ResNet model benefits far more from adding onset predictions, since these contribute mainly to the precision of detecting a note. Together with its better recall performance the ResNet architecture is able to

outperform the CNN in the note F1-Score, by 3.41 points when adding onset predictions and 2.27 points when adding the onset heuristic.

ResNet with OHPCP In contrast to the ResNet experiment described above, this time OHPCPs were used as input data while the model architecture remained largely the same. Due to the lower frequency resolution of OHPCPs it is possible to transfer a wider time context of 15 frames to the ResNet, without having to considerably increase the number of parameters in the model. Despite of this advantage, the ResNet with OHPCP performed poorly on most frame-level metrics, except when considering the precision. Unsurprisingly, this also coincides with the F1-Score progression in Fig. 3.2. However, when considering the note-level metrics in Table 3.4 there is only a minimal difference between this model and the ResNet model using spectrogram representation. Nevertheless, the present findings indicate that OHPCPs in their current form are inferior to semitone-filtered spectrograms.

ResNet & BiLSTM Similar to the previous experiments, the frame-level metrics are not much influenced by the additional postprocessing steps. Overall, this model shows the second best results on frame-wise scores and a healthy ratio between precision and recall. Furthermore, by evaluating the results in tabel 3.4, this architecture showcases an impressive advancement in note F1-Score of 5 to 11 points compared to the previous experiments. In case of the score adapting the onset heuristic it even performs on par with the current state-of-the-art model, as can be seen in Table 4.1.

Compared to the transcription system presented in [4] this combination of convolutional and recurrent architecture is much simpler. It uses only a single path to train frame-wise pitch occurrences and incorporates onset information only from the separate onset detector during the postprocessing stage. When observing Fig. 3.2, it seems the model is considerably better in detecting pitch occurrences of notes with a long sustain. Most likely, this is due to the addition of the BiLSTM layer, which enables the model to incorporate more temporal information and thereby exceeding the narrow time window of five frames used in the ResNet model.

Multitask ResNet & BiLSTM In terms of the mean frame-wise F1-Score of the average note, depicted in Fig. 3.2, the Multitask ResNet & BiLSTM also demonstrates positive properties. While the score of all other models starts to rapidly decrease past the 300 ms mark, the score of this model is able to maintain a

comparatively shallow decrease. Furthermore, the note offset frame detection does not sink below an F1-Score of 50 as is the case for the ResNet & BiLSTM model, which indicates a better ability to correctly detect notes with a long sustain.

The concurrent training of onset detection and general pitch occurrence detection proves a significant improvement of the note with offset metric over the previous experiments and existing piano transcription systems. Furthermore, this system outperforms the frame-level and note-level metrics in every aspect when comparing the results of all experiments conducted. The findings show that training a model in a multitask fashion is superior to using separately trained models for onset detection and frame-wise detection. Compared to the baseline reimplementation of the CNN model this system improves note with offset transcription by 78 % and advances the current state-of-the-art piano transcription system by 10 % (see Table 4.1).

The presented results in Table 4.1 showcase that high frame-level scores do not necessarily indicate better note-level scores, which emphasize on the importance of the onset and offset frame. In this context, the note with offset metric has been deemed one of the more perceptually relevant scores in describing the quality of an automatic piano transcription system [4]. However, it is difficult to truly represent the audible differences of a transcription system in a printed form, Fig. 3.3 is an attempted in doing so, using an exemplary piece from the test-set. The final transcription illustrates the reference and estimated frame-wise occurrences, thereby revealing that the transcription system produces only very few false positive predictions but still misses many notes completely. Furthermore, by comparing the note activation with the final transcription, one can identify many cases where a note is present in the activation function but is not conveyed to the piano-roll. This indicates that there still is a mismatch between the performance of the onset detector and the pitch occurrence detection which leaves room for improvements.

It is apparent that the Multitask ResNet & BiLSTM performs much better in the example chosen in Fig. 3.4 than the Onset and Frames system, since it produces fewer false positive predictions and at the same time is able to detect most of the frames present according to the ground truth.

The ability of the Multitask ResNet & BiLSTM to generalize to new data is remarkable, since it was trained on synthetic audio data and successfully tested on

recorded audio. However, the dataset still limits the validity of the results, as the piano pieces are in their majority from classical genres, posing the question if other genres (e.g. solo jazz piano) could be transcribed in equal quality. This weakness of the dataset is directly related to the need for more data with accurate labels.

Obviously the note-wise measures provide a more realistic performance metric, since the human reception of music does not compare to frame-wise note activations. Furthermore, the frame-wise metric weights all pitch occurrences equally, although some are more important than others, like the onset frame of a note. This problem is partly addressed through the incorporation of an additional onset detection, which shows even better results when trained jointly with a general pitch occurrence detection. However, the impact of the onset heuristic postprocessing step indicates that it is not enough to only focus on the onset of a note. The variations in energy of a tone over time is a complex function, especially when also considering other instruments. Therefore, it might be necessary to advance the current frame-wise loss functions to a form more capable of describing the note-level loss of a transcription model.

The task of correctly predicting the offset of a note still remains the greatest weakness of all considered transcription systems, since adding the offset condition to the evaluation scheme leads to a 20 to 30 points decrease in the F1-Score (Table 4.1). However, the detection of offsets still has not been addressed to the same extent as the task of detecting note onsets. More precisely, offsets in the proposed transcription system are only determined implicitly by assuming the last of a progression of pitch occurrences is the offset. In many cases this approach seems to be a good approximation of the actual note offset and it may even produce a pleasing transcription. Nevertheless, this aspect of the transcription system leaves much room for improvement and should be addressed more thoroughly in future work. A obvious approach would be to implement a third task in the mutltitask transcription system trained to detect the offset frame.

Using an onset heuristic might be considered an anti-pattern in the context of deep learning, since the goal is to let the model learn all necessary features to achieve a good solution. In addition to the already mentioned limitations (see Section 3.1.1) the usage of this heuristic further restricts the ability of the model to generalize, since other piano recordings most likely will have a different attack and decay ratio. A possible solution would be to let the model individually learn the appropriate value for the onset heuristic. However, completely omitting the onset heuristic still leaves the Multitask ResNet & BiLSTM to perform better on the note with offset score than any previous piano transcription systems.

An attempt has been made to further describe the quality of a transcription model by evaluating its ability to, on average, predict individual frames of a note, see Fig. 3.2. The illustration greatly helps in assessing the performance of a model but also reveals the lack of statistical knowledge on the training data (e.g. shortest note duration, highest/lowest note played, etc.). For example, there is a distinct inflection in the F1-Score of all models around 440 ms (Fig. 3.2), which most likely is caused by a peculiarity of the underlying data and which in turn might be explainable by its statistical properties. This kind of supplementary statistical analysis of the data seems to be neglected throughout recent publications and should be undertaken as future work.

4.1 Conclusion

In order to answer the questions posed in Section 1.2 a succession of experiments was conducted to investigate the impact of residual learning and complementary deep learning methods on the task of polyphonic piano transcription. Two main topics were addressed during the investigation of the newly developed transcription systems, do the application of these new deep learning methods advance frame-level and note-level metrics and are the advancements in numbers also perceptually relevant.

Without the additionally investigated methods, the ResNet achieves a barely significant improvement over the original CNN architecture. However, there is an overall better recall performance and the complementary methods allow the ResNet to achieve much greater improvements than is possible with the CNN, both in frame-level and note-level metrics. Overall, the Multitask ResNet & BiLSTM achieves an improvement of 76 % over the baseline CNN and an improvement of 10 % over the current state-of-the-art transcription system in the note with offset score.

The answer to the second topic is twofold, in order to describe the perceptual relevance it is necessary to define a metric which to a reasonable extend can capture this vague property. This has been done by concentrating on the note-level metric, since it has been found to correlate better with the perceptual quality of a transcription. Therefore, the Multitask ResNet & BiLSTM indeed produces perceptually relevant piano transcriptions. Furthermore, an exemplary investigation has been made by directly comparing the excerpt of a transcribed piano recording. The results show that the proposed system in some cases is able to outperform existing transcription systems by detecting more notes correctly and producing fewer false positive predictions.

The success of deep learning in speech, computer vision and natural language processing is mainly build on large high quality datasets being available to the research community. In AMT there still is a lack of high amounts of data, especially for instruments other than the piano. However, even the MAPS dataset is comparatively small and future work on deep learning transcription systems is in dire need for more data.

The findings of this thesis showcase the importance of the onset detection in producing convincing transcriptions. However, the discrepancy between the note activation function and the actual transcription illustrates, how the current aggregation of onsets and the following pitch occurrences still is flawed, leaving many notes undetected. Therefore, a great improvement could be made by further developing postprocessing methods or by integrating the aggregation process into the deep learning model. Similarly, this can be extended to the correct detection of note offsets.

Since comparing single excerpts is not a reliable method and single metrics are not able to describe a property as complex as perceptual relevance, future research should also incorporate listening tests to evaluate the transcription quality of a model. After all it should be the human perception determining if the transcription of a musical piece is able to grasp its essence.

References

[1] Schedl, Markus; Emilia Gómez; Julián Urbano; et al. (2014): "Music information retrieval: Recent developments and applications." In: *Foundations and Trends® in Information Retrieval*, **8**(2-3), pp. 127–261.

[2] Klapuri, Anssi and Manuel Davy (2007): *Signal processing methods for music transcription*. Springer Science & Business Media.

[3] Moorer, James A (1977): "On the transcription of musical sound by computer." In: *Computer Music Journal*, pp. 32–38.

[4] Hawthorne, Curtis; et al. (2017): *Onsets and frames: Dual-objective piano transcription*. Online. URL https://arxiv.org/abs/1710.11153. Access 16.05.2019.

[5] Benetos, Emmanouil; Simon Dixon; Dimitrios Giannoulis; Holger Kirchhoff; and Anssi Klapuri (2013): "Automatic music transcription: challenges and future directions." In: *Journal of Intelligent Information Systems*, **41**(3), pp. 407–434.

[6] Böck, Sebastian and Markus Schedl (2012): "Polyphonic piano note transcription with recurrent neural networks." In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Kyoto, Japan, pp. 121–124.

[7] Li, Samuel (2017): *Context-Independent Polyphonic Piano Onset Transcription with an Infinite Training Dataset*. Online. URL https://arxiv.org/abs/1707.08438. Access 31.05.2018.

[8] Sigtia, Siddharth; Emmanouil Benetos; and Simon Dixon (2016): "An end-to-end neural network for polyphonic piano music transcription." In: *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, **24**(5), pp. 927–939.

[9] Kelz, Rainer; et al. (2016): "On the Potential of Simple Framewise Approaches to Piano Transcription." In: *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*. New York City, United States, pp. 475–481.

[10] Goodfellow, Ian; Yoshua Bengio; and Aaron Courville (2016): *Deep learning*. Cambridge, Massachusetts, London, England: MIT Press.

[11] Dieleman, Sander and Benjamin Schrauwen (2014): "End-to-end learning for music audio." In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Florence, Italy, pp. 6964–6968.

[12] Collins, Nick (2005): "Using a pitch detector for onset detection." In: *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*. London, UK, pp. 100–106.

[13] Pertusa, Antonio and José Iñesta (2009): *Note Onset Detection Using One Semitone Filter-Bank For MIREX 2009*. Online. URL https://core.ac.uk/download/pdf/16368526.pdf. Access 23.06.2018.

[14] O'Hanlon, Ken and Mark D. Plumbley (2014): "Polyphonic piano transcription using non-negative matrix factorisation with group sparsity." In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Florence, Italy, pp. 3112–3116.

[15] Marolt, M.; A. Kavcic; M. Privosnik; and S. Divjak (2002): "On detecting note onsets in piano music." In: *Proceedings of the 11th IEEE Mediterranean Electrotechnical Conference (MELECON)*, 386. Cairo, Egypt, pp. 385–389.

[16] Lacoste, Alexandre and Douglas Eck (2006): "A supervised classification algorithm for note onset detection." In: *Journal on Applied Signal Processing (EURASIP)*, **2007**(1).

[17] LeCun, Yann; Léon Bottou; Yoshua Bengio; and Patrick Haffner (1998): "Gradient-based learning applied to document recognition." In: *Proceedings of the IEEE*, vol. 86. pp. 2278–2323.

[18] Krizhevsky, Alex; Ilya Sutskever; and Geoffrey E Hinton (2012): "ImageNet Classification with Deep Convolutional Neural Networks." In: F. Pereira; C. J. C. Burges; L. Bottou; and K. Q. Weinberger (Eds.) *Advances in Neural Information Processing Systems 25*. Curran Associates, Inc., pp. 1097–1105.

[19] Simonyan, Karen and Andrew Zisserman (2014): "Very Deep Convolutional Networks for Large-Scale Image Recognition." In: *International Conference on Learning Representations (ICRL)*, pp. 1–14.

[20] Lostanlen, Vincent and Carmine-Emanuele Cella (2016): "Deep convolutional networks on the pitch spiral for music instrument recognition." In: *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*. NewYork City, United States, pp. 612–618.

[21] Schlüter, Jan (2017): *Deep Learning for Event Detection, Sequence Labelling and Similarity Estimation in Music Signals*. Ph.D. thesis, Johannes Kepler Universität Linz, Department of Computational Perception, Linz.

[22] Schlüter, Jan and Sebastian Böck (2014): "Improved musical onset detection with convolutional neural networks." In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Florence, Italy, pp. 6979–6983.

[23] Thomé, Carl and Sven Ahlbäck (2017): *Polyphonic Pitch Detection with Convolutional Recurrent Neural Networks*. Online. URL http://www.music-ir.org/mirex/abstracts/2017/CT1.pdf. Access 31.05.2018.

[24] Emiya, Valentin; Roland Badeau; and Bertrand David (2010): "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, **18**(6), pp. 1643–1654.

[25] Kelz, Rainer and Gerhard Widmer (2017): "An experimental analysis of the entanglement problem in neural-network-based music transcription systems." In: *Proceedings of the AES Conference on Semantic Audio*. Erlangen, Germany.

[26] Gómez, Emilia (2006): "Tonal description of music audio signals." In: *Department of Information and Communication Technologies*.

[27] Böck, Sebastian; Filip Korzeniowski; Jan Schlüter; Florian Krebs; and Gerhard Widmer (2016): "madmom: a new Python Audio and Music Signal Processing Library." In: *Proceedings of the 24th ACM International Conference on Multimedia*. Amsterdam, The Netherlands, pp. 1174–1178.

[28] Ewert, Sebastian and Mark Sandler (2016): "Piano transcription in the studio using an extensible alternating directions framework." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, **24**(11), pp. 1983–1997.

[29] Raffel, Colin; et al. (2014): "mir_eval: A transparent implementation of common MIR metrics." In: *In Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR*. Citeseer.

[30] LeCun, Yann (1989): "Generalization and network design strategies." In: *Technical Report CRG-TR-89-4*. University of Toronto.

[31] He, Kaiming; Xiangyu Zhang; Shaoqing Ren; and Jian Sun (2016): "Deep residual learning for image recognition." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 770–778.

[32] Srivastava, Nitish; Geoffrey Hinton; Alex Krizhevsky; Ilya Sutskever; and Ruslan Salakhutdinov (2014): "Dropout: a simple way to prevent neural networks from overfitting." In: *The Journal of Machine Learning Research*, **15**(1), pp. 1929–1958.

[33] Ioffe, Sergey and Christian Szegedy (2015): *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. Online. URL https://arxiv.org/abs/1502.03167. Access 05.06.2019.

[34] Glorot, Xavier; Antoine Bordes; and Yoshua Bengio (2011): "Deep sparse rectifier neural networks." In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. pp. 315–323.

[35] Graves, Alex (2012): "Supervised sequence labelling." In: *Supervised sequence labelling with recurrent neural networks*. Springer, pp. 5–13.

[36] Schuster, Mike and Kuldip K Paliwal (1997): "Bidirectional recurrent neural networks." In: *IEEE Transactions on Signal Processing*, **45**(11), pp. 2673–2681.

[37] Sainath, Tara N; Oriol Vinyals; Andrew Senior; and Haşim Sak (2015): "Convolutional, long short-term memory, fully connected deep neural networks." In: *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. South Brisbane, Queensland, Australia: IEEE, pp. 4580–4584.

[38] Kim, Yoon; Yacine Jernite; David Sontag; and Alexander M. Rush (2015): *Character-Aware Neural Language Models*. Online. URL http://arxiv.org/abs/1508.06615. Access 09.03.2019.

[39] Caruana, Rich (1997): "Multitask learning." In: *Machine learning*, **28**(1), pp. 41–75.

[40] Keskar, Nitish Shirish; Dheevatsa Mudigere; Jorge Nocedal; Mikhail Smelyanskiy; and Ping Tak Peter Tang (2016): *On large-batch training for deep learning: Generalization gap and sharp minima*. Online. URL https://arxiv.org/abs/1609.04836. Access 19.03.2019.

[41] Abadi, Martín; et al. (2016): "TensorFlow: A System for Large-Scale Machine Learning." In: *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. Savannah, GA: USENIX Association, pp. 265–283. URL https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi.

[42] Böck, Sebastian and Gerhard Widmer (2013): "Maximum filter vibrato suppression for onset detection." In: *Proceedings of the 16th International Conference on Digital Audio Effects (DAFx)*. Maynooth, Ireland.

[43] Raffel, Colin; et al. (2014): "mir_eval: A Transparent Implementation of Common MIR Metrics." In: *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*. Taipei, Taiwan, pp. 367–372.

[44] Xi, Qingyang; Rachel M. Bittner; Johan Pauwels; Xuzhou Ye; and Juan Pablo Bello (2018): "GuitarSet: A Dataset for Guitar Transcription." In: *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*. pp. 453–460.