MCAST

Enhancing Clarinet Sound Identification Using Machine Learning Algorithms

Luke Mifsud

Supervisor: Kristian Domancich

September - 2025

A dissertation submitted to the Institute of Information and Communication Technology in partial fulfilment of the requirements for the degree of BSc (Hons) in Software Development

Authorship Statement

This dissertation is based on the results of research carried out by myself, is my own composition, and has not been previously presented for any other certified or uncertified qualification.

The research was carried out under the supervision of Mr Kristian Domancich

September 2025	L.Mifsud
. . . . .	. . . . .
Date	Signature

Copyright Statement

In submitting this dissertation to the MCAST Institute of Information and Communication Technology, I understand that I am giving permission for it to be made available for use in accordance with the regulations of MCAST and the Library and Learning Resource Centre. I accept that my dissertation may be made publicly available at MCAST's discretion.

September 2025	L.Mifsud
.....	.....
Date	Signature

Acknowledgements

I would like to express my deepest apprecitation to everyone who has assisted me in the completetion of my dissertation. The support, guidance, and motivation they have offered have been immensly beneficial throughout the entirety of this trip.

First of all, I would like to extend my most profound gratitude to my mentor, Mr. Kristian Domancich, for his unwavering support, expertise, and patience during this entire process. His advice and criticism have been essential in helping me decide the best way to move this research forward. I am extremely thankful for the tremendous knowledge and understanding I got from his insight.

I also want to use this occasion to express my gratitude to all of the lecturers I had at the MCAST Institute of Information Communication Technology throughout the years for establishing a friendly environment that is suitable to learning and research. Their dedication to setting the greatest possible educational standard, assitance, support , and belief in me have been very important and acted as an inspiration for me throughout the entire process.

Lastly, I would like to mention the people close to me who have supported and helped me through this journey. My family has been my backbone throughout this degree, and I could not have achieved this without them. My friends who provided emotional support and courage through difficult times aswell as my colleagues, who helped me push through this process and achieve success. This dissertation would not have been possible without their support.

Thank you.

Abstract

Musical instrument recognition is an essential task in Music Information Retrieval (MIR), enabling applications in technology, education, cultural preservation, and therapeutic practices. Despite advancements in deep learning methods, identifying instruments with conflicting acoustic characteristics continues to present major challenges. The clarinet, in particular, exhibits pitch and timbral similarities to other instruments, including the flute, saxophone, and trumpet, resulting in frequent misclassification. This study analyzes the capability of neural networks to enhance clarinet recognition and examine the impact of model architecture and dataset capacity.

The experimentation utilized a section of the IRMAS dataset containing audio samples from the clarinet, flute, saxophone, and trumpet. Two dataset sized were used: the full dataset (100%) and a stratified sample (50%) to address the impact of dataset size. The audio data was preprocessed and normalized, and were transformed into Mel-Spectrograms, Mel-Frequency Cepstral Coefficients (MFCCs), and chroma features, which were concatenated to create an abstract input representation. Three types of neural networks were used: Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Convolutional Recurrent Neural Networks (CRNNs). Each group included two architectures, resulting in six experimental models. The evaluation metrics employed were accuracy, precision, recall, F1-score, and confusion matrices.

The CNN models showed the dependence on the scale of the dataset. A lightweight architecture revealed subpar results with the 50% subset but exhibited significant enhancement on the full dataset. The deeper architecture revealed enhanced resiliance to data volume yet continued to misclassify the clarinet, exposing problems with spatial feature extraction when distinguishing similar sounding instru-

ments.

The RNN models observed slight improvements in clarinet identification, specifically under very limited data quantity. However, their results exhibited inconsistencies over different dataset scales, and misclassifications remained among the clarinet and the rest of the instruments. This indicates that just like the CNNs, baseline temporal modeling struggles to effectively capture the clarinet's unique qualities.

The hybrid CRNN models achieved enhanced performance through the combination of spatial and temporal features. Both models exceeded the baseline CNNs and RNNs, with the second CRNN architecture obtaining an overall accuracy of 72.9% on the full dataset and yielding the strongest clarinet classification with an F1-score of 0.696. The CRNNs lowered misclassifications within the woodwinds and displayed enhanced adaptability among the dataset sizes.

The results suggest that clarinet recognition remains a difficult challenge due to its similarities with other instruments. Nonetheless, CRNNs proved to be the most efficient approach, revealing major enhancements compared to the baseline CNNs and RNNs. Dataset size, model architecture, and input representation are key variables that determine classification results. This shows the need for larger, balanced datasets and sophisticated hybrid architectures to enhance clarinet and instrument identification research.

Keywords: Music Information Retrieval, Clarinet Recognition, Neural Networks, Feature Representation, Hybrid Models.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
MIR	Music Information Retrieval
NN	Neural Network
ML	Machine Learning
DL	Deep Learning
FCN	Fully Convolutional Network
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
CRNN	Convolutional Recurrent Neural Network
MFCC	Mel Frequency Cepstral Coefficient
GRU	Gated Recurrent Units
BiGRU	Bidirectional Gated Recurrent Units
STFT	Short-Time Fourier Transform
LSTM	Long Short-Term Memory
MFS	Mel-Frequency Spectrogram
MLP	Multi-Layered Perceptron
SVP	Support Vector Machine
k-NN	k-Nearest Neighbours
BN	Batch Normalization

Chapter 1: Introduction

1.1 Research Background

Musical instrument recognition is an essential subject within the field of Music Information Retrieval (MIR), employing technological techniques to analyze, retrieve, and organize musical information. Over the years, the growing demand for reliable recognition techniques for instrument identification is caused by the expansion of services such as streaming platforms and digital music archives. These techniques provide a benefit not just for retrieval purposes, but can offer educational purposes, evaluation of musical performances and interactive music technology [1].

Prior to the emergence of deep learning, instrument recognition mainly focused on manually trained features and tradiional classifiers, including Support Vector Machine (SVM) and k-Nearest Neighbors (k-NN), often combined with feature representations such as Mel-Spectrograms, MFCCs and Chroma Features. Whilst these systems achieved partial success, they faced challenges with complex, polyphonic audio and failed to provide the necessary adaptability for many musical environments [2]. The emergence of deep learning revolutionized the field by allowing models to automatically acquire musical characteristics from data, reducing the reliance on manual design and greatly improving instrument recognition [3]. This evolution hgihlights the reason Neural Networks have emerged as one of the main pillars in MIR research.

The addition of deep learning was a breakthrough in MIR research, offering new methods for acquiring the complex acoustic and temporal features of audio recordings. Neural Networks (NNs) are widely used in instrument recognition due to its capability of capturing both spectral and temporal information which previous methods failed to convey. Neural Networks such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and their hybrid combination are some of the most popular NNs that have been explored, with each NN enhancing accuracy in instrument recognition under different circumstances.

In addition to music retrieval and educational uses, instrument recognition helps with maintaining cultural preservation as well as musical therapy. Old recordings can be automatically classified and preserved, ensuring that instruments like the clarinet, vital to both commnuity bands or orchestra traditions, and to the Western classical music community, remians available for future generations. In regards to musical therapy, instrument recognition can assist existing immersive applications that can modify music to improve mental and emotional health [4]. Expanding the scope of MIR to more diverse applications highlights the need for stronger recognition systems, hence pushing research into complex acoustic instruments, such as the clarinet.

Despite several developments within MIR, identification remains to present challenges specifically for instruments with overlapping characteristics. Instruments like the clarinet and the flute, both belonging to the woodwinds section, share similar pitch range, frequency, and timbral qualities, resulting in resembling harmonics. These similarities challenge machine learning models to consistently clas-

sify and identify these instruments.

1.2 Problem Statement

Even though deep learning has enhanced the overall precision of musical instruments, the clarinet continues to be one of the more problematic instrument to identify correctly. Its broad spectrum, dynamics and adaptive timbre makes it a very verstaile instruments, leaving it vulnerable to incorrect identification by computational models. Research by [5] showed that clarinet audio is mistaken for instruments like the flute and saxophone, specifically when the audio samples are short and contain overlapping harmonics, leading to reduced accuracy relative to other instruments.

This exposes an obvious weakness, where despite instrument recognition being improved overall through machine learning, an instrument like the clarinet is still lacking the accuracy and consistency required for a broader range of applications. To address this problem, it requires dedicated research on how different neural network architectures and input representations could improve the capability of capturing the clarinet's unique timbral and harmonic properties.

1.3 Rationale and Motivation

This research has been inspired by the lack of MIR studies on the clarinet due to its identification challenges despite the rest of the field making significant advancements. Despite the success neural networks have shown in instrument identification, their effectiveness with acoustical comparable instruments such as wood-

winds including the clarinet and flute remains inconsistent. The clarinet acts as a great case for exploring how neural networks structures and different input representations affect its accuracy.

This study has also been driven by both academic and personal factors. From an academic perspective, the improvement of clarinet identification contributes to MIR not just from a clarinet standpoint, but to other instruments possessing similar characteristics by offering approaches that are applicable to them. A broad range of practical applications include the development of software to aid clarinet students by giving them automatic feedback, tagging of audio within streaming platforms, and more consistent tools for interactive music technology [6].

On a personal view of the subject, the determination comes from my experience of playing the clarinet in a band club. This background offers an understanding of the clarinet's versatility and expressiveness as well as the importance of accurate recognition in a monophonic and polyphonic environments. By combining person experience with technical research, this study aims to improve the practical and theoretical aspects of clarinet sound identification.

1.4 Significance of the study

This study holds weight in both practical as well as academic settings. This study advances the field of Music Information Retrieval (MIR) by addressing an established challenge found in instrument recognition: the challenge of distinguishing the clarinet from acoustically similar instruments. This study offers a perspective into the development of classification models to handle acoustic characteristics by

analyzing several neural network designs, architectures, and input represenatations. The results obtained are relevant to clarinet identification and can be adapted to other instruments, specifically other woodwinds that face similar issues, hence helping in the progress of a more universal instrument identification framework.

This study provides siginficant benefits across several practical fields. The enhancement of clarinet identification could improve music education by giving automatic feedback to learners, allow digital platforms to give more precise data for audio recordings, and promote interactive technologies such as AI-assisted composition or AI based musical performance evaluation. As the MIR field continues to merge with the AI field, innovations in instrument recognition for promoting unique applications that promote creativity, organisation, and education [7].

1.5 Dissertation Structure

This dissertation is split into five chapters. The first chapter presents an overview on the subject and relating field, as well as discussing the issue statement and reasoning behind this study. The second chapter provides a literature analysis, evaluating past studies on instrument recognition, neural networks, and feature representation. The third chapter includes the research methodology covering the research techniques, data collection, experimentation methods, and model implementation. The fourth chapter presents the research findings and analysis of clarinet sound identification with neural network models, and examining their impact to the research aims. The fifth chapter concludes the study by summarizing the key results, identifying limitations, and proposing opportunites for future research.

Chapter 2: Literature Review

2.1 Convolutional Neural Networks for Instrument Classification

Audio processing has been revolutionised by deep learning, specifically in the field of music information retrieval through identification of musical instruments. Convolutional Neural Network (CNN) is a renowned method that has shown good results in extracting features from waveforms or spectrograms. Since CNN architectures are effective in learning patterns and time-frequency representations, it makes it a suitable candidate to analyse and identify various musical instruments, including the clarinet. This section addresses many applications and different architectures of CNN used for instrument identification and classification.

2.1.1 CNN architectures and their applications

[8] took a visual approach to detect note onsets in clarinet performances. The researchers implemented a 3D CNN model to identify note onset in each frame sequence. The model was trained and tested using the Clarinettists for Science (C4S) dataset, consisting of 36,000 annotated onsets from 4.5 hours of clarinettist's videos. The results indicate that the proposed method performs slightly better than the baseline and identified a gap of 60% between vision-based and audio-based onset detection due to poorly optimized precision and recall. In a different study by [9] focused on identification of certain instruments in polyphonic music, including the clarinet. Opposed to [8], [9] constructs a 2D CNN

model and uses audio data like mel-spectrograms as input. The model uses the sliding window technique to handle variable-length audio inputs. The IRMAS dataset was used, utilizing 10,000 audio clips for 11 different instruments. With F1 scores of 0.619 (micro) and 0.513 (macro), the proposed model outperformed sophisticated methods by 23.1% and 18.8%. The model identified a weakness on instruments with soft onsets such as cello and clarinet.

An alternative approach to instrument identification was suggested by [10] where the proposed model is made up of individual CNNs for every instrument (bass, guitar, drums, and piano) and extracting the Mel-Frequency Cepstral Coefficients (MFCC) from the raw audio signals. This approach allows flexibility, allowing the authors to add or remove an instrument submodel without retraining the entire network. The Slakh dataset was used for this model which contains 2100 audio tracks. This method had a high performance, achieving a precision high of 0.99 for drums and a precision low of 0.86 for the guitar.

Instead of classifying different instruments, [11] conducted a different approach to musical instrument identification using multiple spectrograms to understand how spectrograms affect a CNNs decision-making process. The NSynth database was used and was split into six different spectrogram types: STFT, Log-Mel, MFCC, Chroma, Spectral Contrast, and Tonnetz. The model was trained on each spectrogram type. The authors utilized integrated gradients to generate heatmaps to identify which spectrogram characteristics contribute to the model's prediction. The results from the heatmap show that each spectrogram captures unique features from the audio signal, with MFCC being the most accurate overall (0.62), how-

ever other features like Chroma and Tonnetz were very useful to identify pitch and harmonic relationships, so whilst the MFCC spectrogram seems like the best choice, the type of spectrogram can severely change the models performance depending on the instrument chosen.

Raw waveforms is another source of input compared to spectrograms or other time-frequency representations. Explored by [12], this model uses a 1D CNN to identify spatial features alongside a Bidirectional Gated Recurrent Units (BiGRUs) to take care of long-term dependencies. This model was trained using the IRMAS dataset, achieving good results with an F1-micro score of 60.77% and F1-macro score of 54.31%, compared to spectrogram-based models. The research shows that raw waveforms work particularly well with brass instruments, on the other hand, predominant instruments for example guitar and piano work better with a spectrogram-based input like Constant Q Transform (CQT) spectrogram. An upside compared to other studies; this model requires minimal preprocessing due to the absence of time-frequency representations.

Collectively, these studies demonstrate that factors and parameters such as input representation, model design, and onset characteristics – have a major impact on identifying musical instruments. The type of input representation shows an influence on the model's performance with spectrogram-based methods working greater for the better part of the instruments, despite that raw waveform still has some advantages, particularly in the brass section of instruments. The architecture of the model also plays a role, with opposing studies like [10] and [12], where [10] is designed for more flexibility by using multiple models for each

instrument, and [5] focuses more on hybrid models capturing long-term dependencies. While CNNs show very good potential in instrument identification, challenges like soft onsets still pose difficulties for classification which need further optimizations [9].

2.1.2 Sound quality analysis in a CNN Environment

Just like [10] and [11], [13] also prioritizes a spectrogram-based interpretation, however the focus is more on autocorrelation spectrograms to evaluate the quality of clarinet sounds in real time. Their study uses a dataset that contains clarinet recordings from both professionals and students. In contrast to other studies, this study gives importance to the sound quality instead of instrument identification. An 87.5% accuracy was achieved with autocorrelation spectrograms using AlexNet, surpassing results from other spectrograms like MFCC [11] however the same spectrogram with the GoogleNet architecture only achieved 50% accuracy.

2.2 Monophonic vs Polyphonic Music Analysis

The difference between monophonic (singular instrument performing at a time) and polyphonic (multiple instruments performing simultaneously) can vary which instrument identification techniques are applied. Monophonic sounds lack overlapping frequencies which makes identification more straightforward. Polyphonic sound presents additional challenges to separate the sources of the sounds and other interferences.

2.2.1 Monophonic Instrument Recognition

Over the years, many different approaches have been taken to improve monophonic accuracy, including the study from [14], where they introduce a scalogram-based method using detailed timbral features in which demonstrating that time-frequency representations can enhance monophonic recognition. On the other hand, [15] focused on comparing a CNNs and an RNNs model to determine which is more suited for monophonic recognition tasks, where the CNNs performed better as temporal dependencies are not that relevant for classification of isolated instrument sounds.

Despite its simplicity, monophonic recognition still faces challenges. [14] uses an unconventional method where it achieved good results at the cost of computational power. Whilst CNNs have become the favoured architecture for monophonic recognition, [16] emphasize the selection of dataset, as models trained on real-life recordings may yield better accuracy and generalisation of results over synthetic datasets like NSynth [17]. Furthermore, other inconsistencies such as timbre changes and instrument articulation can alter performance of monophonic identification hence research continues to refine monophonic techniques as it is still a fundamental aspect of musical instrument recognition and offers insights that support more complex classification tasks.

2.2.2 Polyphonic Instrument Recognition

In general, polyphonic instrument recognition is a much more challenging task compared to monophonic instrument recognition as a result of overlapping har-

monics and more complex spectral features. Conventional techniques such as non-negative matrix factorization (NMF), has made polyphonic identification easier by breaking down mixed signals into individual components. [18] experimented with NMF algorithms combined with a source-filter architecture, achieving a 59% recognition rate for six-note polyphony. However, this approach struggles with loss of spectral features and timbral resemblance of the different notes.

Integration of more advanced neural architectures and attention mechanisms were researched to mitigate these limitations. [19] developed an attention-based technique intended to improve datasets with weakly labelled data. Whilst the results did not lead to an improvement in precision compared to other models, the recall and F1-score metrics improved significantly across 20 instruments. The attention model (ATT) also performed better against certain other models regarding certain class imbalances, such as the clarinet and flute from the OpenMIC dataset.

As alluded to earlier, polyphonic sound recognition requires more advanced models to handle its complexity. [20] address this by utilising a CRNN hybrid architecture for polyphonic analysis. The approach of combining convolutional layers with gated recurrent unit (GRU) layers achieved an improvement over the baseline model proposed by [9]. The success of the hybrid architecture inclines more computational efficient and complex hardware is needed to overcome challenges in polyphonic recognition.

2.3 Spectrogram Representations in Instrument Recognition

Spectrograms represent the frequency aspect of an audio stream throughout time. They serve as an important input parameter for CNNs applied in the classification of musical instruments. Spectrograms can distinguish between several characteristics such as timbre variations, playing styles and articulations, all of which are essential to identify the clarinet from other woodwinds with similar features.

2.3.1 Mel Spectrogram

As was previously discussed in Section 1, spectrograms are frequently included in CNN-based models for instrument classification, specifically mel-spectrograms. In this section, mel-spectrograms are deeply discussed, highlighting how they extract time-frequency representations that are crucial in differentiating similarly characteristic instruments.

In the study by [21], mel spectrogram is the main tool used for changing raw audio signals into a suitable format for musical instrument recognition using CNNs. The audio is converted into a two-dimensional matrix representation, which enables the mel spectrogram to extract features using Short-Time Fourier Transform (STFT) to capture time and frequency and is used as the input for the CNN to differentiate between the instruments. A 92.8% accuracy rate was achieved mostly because of the incorporation of Max pooling layers, ReLu activation functions and Mel spectrograms. The researchers also discuss efficiency where mel spectrograms facilitate dimensionality reduction, optimizing the CNN during training.

In contrast to [21], [22], focus more on using sound segregation to pre segment the audio into three mono streams using LRM segregation before converting the input into mel frequency spectrograms (MFS). By segregating various audio features, this approach improves the overall accuracy and allowed the CNN to train from the spectrogram representations better. With F1 scored of 0.631 (micro) and 0.539 (macro), Relkar and Tejwani achieved competitive results, producing better accuracy than other studies like [9] that did not make the use of segregation techniques.

Log-mel spectrograms is an extension of the traditional mel spectrogram, combining the advantages of mel frequency scaling and logarithmic amplitude scaling, improving feature extraction for CNNs. [11] shows the notable performance of log-mel spectrograms, achieving an overall accuracy of 0.55, particularly excelling in identification of the flute and mallet. Since the log-mel spectrogram is planned to closely match how people sense loudness and pitch, this feature can help models distinguish the subtle qualities of the instruments sounds which possess unique harmonic structure. This is important for softer instruments, such as the flute, where these subtle distinctions are necessary for accurate classification.

Since mel-spectrograms are great at extracting spatial features, CNNs are frequently used with them. However, spectrograms can be integrated into other architectures, such as RNNs and hybrid CNN-RNN models, where they integrate frequency-based and temporal patterns together. Despite that, due to its success in visual pattern recognition, CNN models remain the dominant model to be paired with a spectrogram.

2.3.2 Mel-frequency cepstral coefficient

Mel-frequency cepstral coefficient (MFCC) is a feature utilized to characterize the sound spectrum in a way to resemble human hearing. Machine learning models can recognise instruments by considering the unique spectral features by extracting MFCCs from audio recordings.

MFCCs can be applied on a traditional single-time resolution or a multiscale resolution. As shown by [23], these different applications of MFCC can have an impact on the accuracy of the model. The proposed method involves implementing multiple MFCCs at different time resolutions and combining them to create multiscale features. Opposed to studies like [9] [16], monophonic instrument recordings are used for the dataset comprising of 2,755 real-life performance recordings, including the clarinet. The results reveal the effectiveness of multiscale MFCC features, outperforming single-time resolution features across different instrument classifications, with the OverCs feature performing the best. Unlike other traditional methods, the multiscale approach does not take scalability in consideration as it requires more computational power due to additional calculations being done compared to a single time resolution.

Similar to [23], [16] take on a similar direction in terms of classifying monophonic instrument recordings with MFCC. Their method employs a k-nearest neighbour (K-NN) algorithm to extract features from the MFCC for five different instruments (piano, cello, flute, violin, trumpet). Despite only using 90 training samples, which is relatively small for a dataset compared to other studies like [10], the model still showcased a 91.66% accuracy for the cello, piano, and

trumpet, a very good result giving further proof MFCC is an ideal feature for instrument identification and when paired up with other features it could further enhance accuracy. However, limitations were discussed particularly on the dataset, indicating that the model improve if the dataset is expanded. The authors also acknowledged that an increase in the number of instruments may reduce recognition accuracy. This suggests a compromise between the dataset and model generalization, despite no explicit threshold being established. This could result in overfitting if the dataset is not scaled accordingly to the addition of instruments, or under trained if there is a lack of samples.

The application of MFCCs alongside principal component analysis (PCA) and a multi-layered perceptron (MLP) for the classification of flute, violin, and piano audio samples is looked at by [24]. The authors opted to use two datasets: 2004 samples were used from the RWC Music Database, and an additional test set was gathered from the McGill University Master Samples (MUMS) database. The goal of this study was to find the ideal number of MFCCs needed for optimal musical instrument identification. The results show that combining the first 15 MFCCs and choosing four principal components of the PCA produces the best classification with an accuracy of 95.88%. Although a high result was achieved, Loughran et al. insist that further studies could achieve better results with additional spectral and temporal features to enhance overall identification accuracy.

Although noting differences in methodological approaches and findings, the examined studies all suggest that MFCC is suitable for musical instrument identification. By using sophisticated machine learning models and other approaches

such as dimensionality reduction [24], and other acoustic characteristics, the performance of MFCCs can still be improved. These additions are important to for resolving issues or limitations such as complex polyphonic music signals, where overlapping makes classification more difficult or in the case of a small dataset. These underlying problems emphasize the need for further studies on hybrid methods that can handle the complex nature of musical instrument identification with the combination of MFCCs and complementing techniques.

2.4 Recurrent Neural Networks and Hybrid Models

Recurrent Neural Networks (RNN) are essential to detect temporal dependences of audio signals. Particularly the Bidirectional Gated Recurrent Units (BiGRU) and Long Short-Term Memory (LSTMs), these architectures allow deep learning models to evaluate sequential patterns in raw instrument waveforms, as opposed to traditional classification techniques centred around spectral features. RNNs make use of temporal context to identify instrumental tones from another.

2.4.1 Recurrent Neural Networks for Sound Identification

Gated Recurrent Units (GRUs) were used by [25] to synthesize musical instrument sounds by primarily focusing on certain characteristics of these sounds, specifically the transient sounds produced being produced. Wyse and Huzaifah emphasize that transients can act as a crucial identifier for identifying various playing styles and instruments which directly relates to sound identification. Despite processing various inputs such as volume, pitch, and audio samples, the model still faces drawbacks when capturing transients. Some of these drawbacks

point to the dataset utilized for training (SynthEven and SynthOdd). The RNN's capability was hindered by this training set since it found it challenging to react to certain changes in volume that were not encountered during training, signalling that a larger and broad dataset and better training techniques are necessary to improve identification. Despite the networks success in responding to certain input parameters, the accuracy of recording transient sounds remained inconsistent.

An alternative approach by [12], uses Bidirectional GRUs (BiGRU) to capture sequential audio data by recording both past and future context to classify raw waveform inputs. They noticed that while BiGRU performed worse as a stand-alone model compared to CNNs, its integration in hybrid models was able to capture long term relationships better. This result shows that BiGRUs are a viable option for improving sound identification in more robust datasets due to its bidirectional nature, an important asset for recognition particularly in timbral changes in musical signals.

2.4.2 Hybrid CNN-RNN Models

Recent research has shifted towards hybrid models that incorporate spatial feature extraction capabilities of CNNs, with temporal sequence modelling of RNNs. By combining the advantages of both models, these hybrid architectures possess benefits and shown potential in instrument recognition tasks. BiGRU layers have been a main choice in these CRNN models as it has shown to enhance long-range dependencies in audio data. [26] investigated how different hybrid architectures for music identification work with various spectrogram representations. Following the study of [12], Ashraf et al. achieved an accuracy of 89.30% us-

ing the CNN-BiGRU model with mel-spectrograms, whilst a CNN-LSTM model performed better on MFCC features with a 76.40% accuracy. According to their results, Bidirectional recurrent layers can obtain more contextual information from sequential audio data over unidirectional architectures, aligning with the results of the previous study of [12], who found that a BiGRU-enhanced CNN yielded better classification results when working with raw waveforms. These studies highlight the choice of spectrogram representation is very important when it comes to hybrid models as different feature extractions affect the model's performance.

In addition to architectural choices, dataset selection is a crucial factor that influences feature learning and overall performance in hybrid CNN-RNN models. Differences in datasets such as class balance, background noise, labelling accuracy and polyphonic complexity can impact the effectiveness of deep learning architectures, especially hybrid architectures. [27] explored this aspect by using the Google AudioSet dataset with poorly labelled segments on CNN, LSTM, and hybrid CNN-LSTM models for speech and music identification. According to their results, the hybrid CNN-LSTM model beat solo CNNs and RNNs, obtaining an accuracy rate of 85%. Despite the good accuracy, classification errors were produced due to the nature of the dataset, with characteristics like background noise and unequal class distributions, made it more challenging for the model to distinct between speech and music in polyphonic settings. Furthermore, research utilizing other datasets such as the IRMAS dataset from [28] indicates that models trained on curated datasets with careful labelling tend to overfit monophonic instrument audio whilst more reliable architectures such as a CNN-Hybrid model is

needed to manage real-world variation such as polyphonic audio found in large datasets such as the AudioSet dataset [27]. This contrast underlines that to mitigate overfitting and enhance real-world adaptability, hybrid CNN-RNN models require both architectural improvements and a thorough dataset selection.

To support the findings from the examined literature, Table 2.1 presents a summary of some prior studies, emphasizing the model choices, dataset utilization, feature representation and key outcomes.

Table 2.1: *Comparison of prior studies on musical instrument recognition*

Author & Year	**Dataset**	**Features Used**	**Model Type**	**Key Findings / Results**
Solanki & Pandey (2019)	IRMAS	Mel-spectrograms	CNN	Achieved F1 = 0.619 (micro). Weakness on soft-onset instruments such as clarinet.
Relkar & Tejwani (2019)	Custom polyphonic dataset	Segmented Mel-spectrograms	CNN	Improved accuracy with segregation techniques. Outperformed baseline CNNs without segmentation.
Loughran et al. (2014)	RWC + MUMS	MFCC + PCA	MLP	95.88% accuracy using 15 MFCCs and 4 PCA components. Highlighted MFCC as highly effective for monophonic classification.
Wyse & Huzaifah (2017)	SynthEven & SynthOdd	Raw waveforms (transients)	RNN (GRU)	Effective at capturing transients but inconsistent under volume variation. Larger datasets needed.
Ashraf et al. (2023)	IRMAS	Mel-spectrograms / MFCC	Hybrid CNN-BiGRU, CNN-LSTM	CNN-BiGRU achieved 89.3% (mel-spectrograms). CNN-LSTM performed better with MFCCs. Hybrids outperformed standalone CNN/RNN.
Bosch et al. (2012)	IRMAS	Chroma, Mel-spectrograms	SVM + traditional methods	Demonstrated baseline MIR challenges. Frequent misclassification among woodwinds (clarinet vs flute).

In this literature review, several techniques for identifying similar instruments have been explored, particularly focusing on the factors affecting a model's performance such as feature representation and model design. Each model contains its strengths, CNNs perform particularly well in spatial feature extraction in combination with mel-spectrograms, RNNs handle temporal dependencies better when

using feature such as MFCCs, and the combination of these strengths, offer enhanced performance through the hybrid CNN-RNN model. These results provide a strong foundation for building an effective clarinet recognition system. This review serves as a comparison to previous studies and as guidance of this research by obtaining information on how different architectures interact with audio data. The solutions and challenges found in the studies- such as different input representations and model generalizability, provide guidelines for developing a framework appropriate for the objectives of my thesis. The examined literature functions as a foundation, affecting the experimental setup and assessment procedures that the methodology will explore.

Chapter 3: Research Methodology

3.1 Research Questions

The focus of this study is to explore different machine learning architectures in the successful identification of the clarinet from audio samples. This study aims to address three research questions, which are stated as follows:

- **RQ1:** Can neural network algorithms accurately recognize a clarinet sound from multiple instruments with similar pitch and frequency?
- **RQ2:** Which machine learning algorithm handles frequency and pitch similarities the best when isolating the clarinet from the other instruments?
- **RQ3:** Does the size of the dataset effect how well the machine learning algorithm can recognize a clarinet sound?

RQ1 will be addressed by training and evaluating three different neural network models using normalized audio data from multiple instruments as an input. The models' ability to capture clarinet sounds from similar instruments will be evaluated using feature extraction techniques. For RQ2, consistent input data and measure will be utilized to compare the performances of each model: CNN, RNN, and a hybrid CNN-RNN. By comparing the models, we can determine which algorithm performs better at distinguishing the clarinet from instruments with similar characteristics. In terms of RQ3, this study will examine how helpful the selected size of the dataset is in clarinet sound identification. While

the dataset size remains fixed throughout the experimentation, it will be assessed through comparison of the results of previous research that used different dataset sizes.

3.2 Research Approach

The proposed research adopts a quantitative research approach, with the primary focus on the gathering and analysis of numerical data examine how different machine learning models can identify clarinet sounds. This specific study aims to enhance the overall accuracy of clarinet sound identification, particularly in setting it apart from other instruments with similar harmonics, pitch, and timbral characteristics like the saxophone, flute, and trumpet.

Given the type of data and the purpose of this study, the best suited approach is of a quantitative nature. Feature extraction techniques such as Mel Spectrograms, Mel Frequency Cepstral Coefficients (MFCCs), and Chroma features can be utilized to transform the audio data retrieved from the dataset into numerical and image representations. These features will then be passed on as inputs for the neural network models, which will generate several numerical metrics including overall accuracy, precision, recall, and F1-score to objectively calculate the performance of the models.

The study consists of the training and testing of several deep learning models, focusing on Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) and a hybrid network of a CNN-RNN architecture, on a chosen curated dataset made up of musical instrument audio samples. The study aims to identify

which model architecture feature combination works best for identifying clarinet sounds by quantitatively compare each model's performance.

The obtained results are certain to be relied on and unaffected from subjective interpretation as the use of controlled experimentations conditions like fixed sample durations and uniform sample rates are a priority for this study to encourage quantitative outcomes and statistical confirmation.

3.3 Dataset Collection

This study uses part of the **IRMAS** dataset **(Instrument Recognition in Musical Audio Signals)** [28]. The dataset includes annotated audio clips obtained from real music recordings, with each clip marked based on the major instrument used. This polyphonic environment offers a more realistic and challenging basis for model training over isolated music recordings.

For the purpose of this study, four different instruments where chosen: the clarinet, trumpet, saxophone, and flute. These instruments were carefully selected to cover the challenge of identifying clarinet sounds to instruments with similar timbral and frequency characteristics and overlapping pitch ranges. The initial dataset consists 505 samples for the clarinet, 577 for the trumpet, 626 for the saxophone, and 451 for the flute. Despite being slighlty unequal, this study will retain the original class distributions. A stratified sampling approach was used to produce a subset comprising 50% of the dataset, with the goal to examine the effects of dataset size on model performance. This ensured that each class was equally represented in both the 100% dataset and the 50% subset. 3.1 shows the

number of samples utilized in both datasets.

Table 3.1: *Number of audio samples per instrument in the full dataset (100%) and stratified subset (50%).*

Instrument	**100% Dataset**	**50% Subset**
Clarinet	505	253
Trumpet	577	289
Saxophone	626	313
Flute	451	226
Total	2159	1081

The dataset is split into two subsets: training which will consist of 80% of the audio files, and testing having 20% of the audio files. This approach ensures balance between the learning, optimization and performance of the model and is a widely used approach for splitting a dataset. The audio clips in the dataset are of WAV format and will be resampled to a rate of 22,050 Hz to guarantee consistency. This rate is chosen as it provides a balance in computational efficiency and frequency resolution. All audio clips will be normalized to have a similar loudness level using the librosa library package in python to prevent the model from associating the volume with an instrument class. The normalized audio clips will be used for feature extraction methods like the Mel spectrograms, Mel Frequency Cepstral Coefficients (MFCCs) and Chroma features.

3.4 Experimental Variables

For this study, several experimental variables are taken into consideration to manage and control the evaluation of the models for clarinet identification. These experimental variables are independent variables, dependent variables, control variables and confounding variables.

The main independent variables of this research consist of the different model architectures and the types of input characteristics chosen. The explored model types are the following: Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and a hybrid Convolutional Recurrent Neural Network architecture. Additionally, three separate audio representations were chosen for the input characteristics: Mel Spectrograms, Chroma Features, and Mel Frequency Cepstral Coefficients (MFCCs), which will be assessed in combination with each other. These specific variables have been selected to analyse which configuration can effectively distinguish the clarinet from other similar instruments.

The dependent variables are comprised of the measurable outcomes that the models produce to show how well they work. These results include precision, recall, F1-Score and classification accuracy. All these metrics will be utilized to statistically assess the overall performance of each model by direct comparison with the other models in this study and in relation to results from previous studies. Other measurements such as confusion matrices will be presented to compare accurate and inaccurate predictions of each class compared to the clarinet.

To ensure an even comparison throughout various experimental conditions, some control variables are kept fixed during the experimentations. As previously mentioned in Subsection 3.2, parameters related to the dataset like sampling rate and class balance are kept constant. This consistency ensures that alterations to the model architectures and feature inputs are the sole reasons for changes in the acquired results.

Certain confounding variables may still influence certain aspects of this study

which are outside of the experimentats control. The chosen dataset, after being selected and pre-processed may still include negatives such as background noise and overall mediocre sound quality which can impact the training process and lead to classification inaccuracy. These elements will be discussed further during the analysis and concluding chapters as they are seen as limitations.

3.5 Experimental Methods

This section describes the implementation method for classifying clarinet sounds using deep learning models. The procedure is split into three main stages: data preprocessing, feature extraction, and model training.

3.5.1 Data Preprocessing

For this study, to ensure consistency in the audio samples, the audio data is standardized prior to feature extraction. As already mentioned in the dataset, the clips will be further standardized using the librosa package to resample each clip to 22,050 Hz. The uniformity of the refined dataset ensures the same characteristics and temporal and frequency resolution, minimizing inconsistencies and poor recording quality. After resampling, the audio files are further normalized to adjust for differences in amplitude. This process eliminates unintentional links between the volume levels of the instruments and the classes from being processed by the model. After normalization, the dataset is configured so that each instrument class contains a relatively similar number of samples around 450 each to balance it out. Stratified sampling is used to split the dataset into two groups: training (80%), and testing (20%), providing equal class distribution for each sub-

set.

3.5.2 Feature Extraction

The librosa package in python is used to extract a variety of audio characteristics required for the conversion of the raw audio file into a form that fits the models. Three main feature extraction methods were chosen for this study: Mel Spectrograms, Mel Frequency Cepstral Coefficients (MFCCs), and Chroma Features. Each method will capture unique features from the audio input to add to the model's overall performance. The matplotlib package serves as a helper package to output the generated images. The Mel Spectrograms were created by utilizing the `librosa.feature.melspectrogram()` function from the librosa package. The function calculates the short time power spectrogram of the audio stream and translates it onto the Mel scale. This method is used as it prioritizes lower frequencies to mimic the human auditory experience. Additionally, the `librosa.powert_to_db()` transforms the spectrograms into a decibel scale, which allows for better understanding of the amplitude dynamics. These time-frequency matrices are used as one of the feature inputs for the CNN and CNN-RNN hybrid models and will serve useful for capturing particularly the frequency patterns and temporal differences in the instruments sound.

The Mel Frequency Cepstral Coefficients (MFCCs) are extracted by using the `librosa.feature.mfcc()` function, a common tool for instrument and sound recognition. A default value of 13 coefficients per frame is used, which captures key timbral qualities that helps in distinguishing the instruments, were obtained from using the Mel-Scale powered spectrogram of every audio clip. These

coefficients were averaged across the time axis to obtain a fixed-length feature vector for each one of the samples. To observe these MFCCs, the function `librosa.display.specshow()` is used and were saved as images for further processing. The models of the RNN and CNN-RNN hybrid model use these features to understand temporal patterns related to each class of instruments.

Chroma features analyzes the distribution of energy over time for the 12 different pitch classes of the musical scale. These features are extracted by the `librosa.feature.chroma_stft()` method and derived from the short-time Fourier Transform (STFT) of the audio signal, to capture tonal and harmonic characteristics for the instrument classes with overlapping melodic ranges, like the flute, and the clarinet. Librosa.display.specshow() is used again to display the chroma matrices, which were then saved as images for CNN and CNN-RNN model processing. These Chroma features gather note information regardless of the octave, improving identification across similar harmonic instruments.

For the CNN models, these feature inputs were coverted into visual representations, enabling their concatenation into a three-channel input. In the RNN and CRNN models, these features were saved as **NumPy** arrays in order to preserve the temporal axis of the RNN.

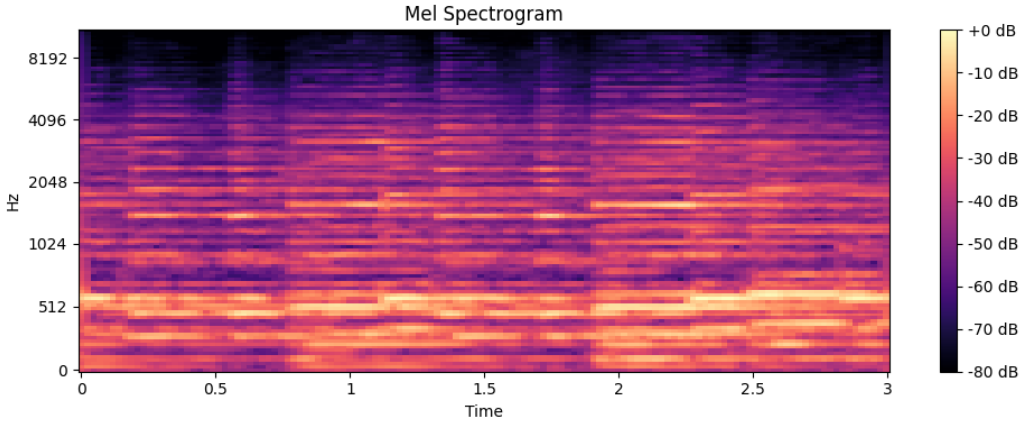

Figure 3.1: *Mel-spectrogram: captures time–frequency patterns.*

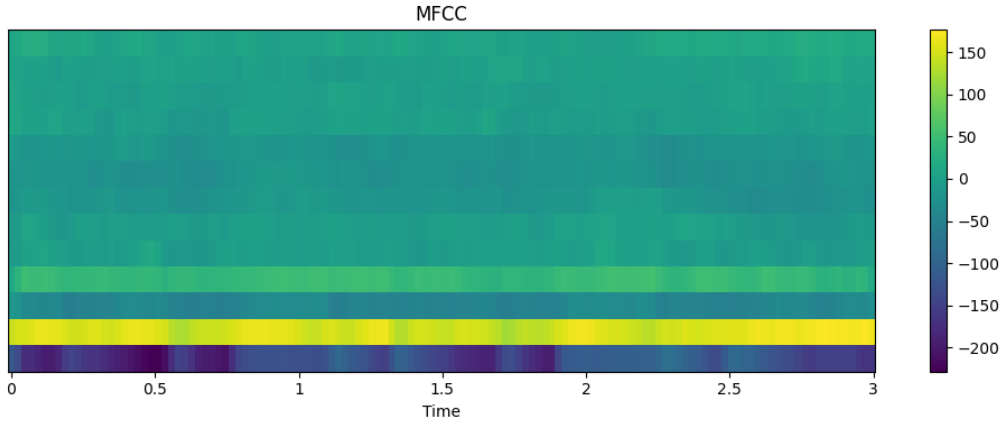

Figure 3.2: *MFCC: highlights timbral information.*

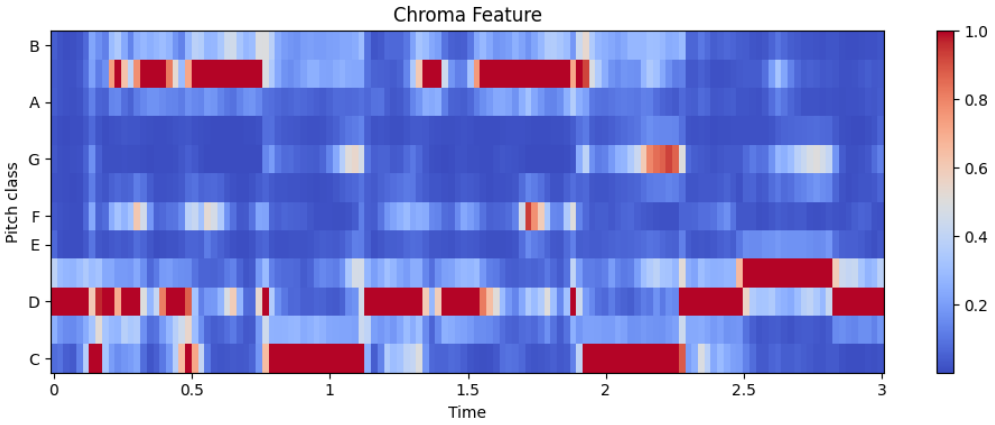

Figure 3.3: *Chroma features: represent pitch class distributions.*

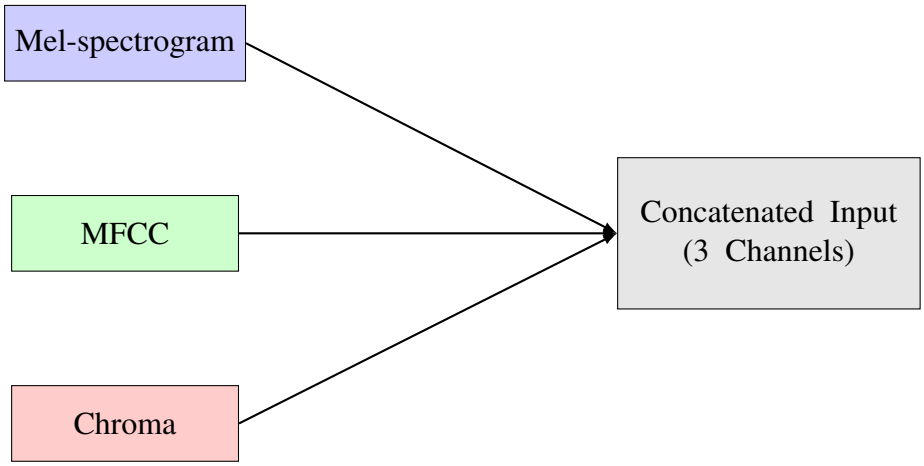

Figure 3.4: *Illustration of feature concatenation: Mel-spectrogram, MFCC, and Chroma are combined into a three-channel input for training.*

3.5.3 Model Construction and Training

The construction of the CNN models in this study was done using the Keras API from the TensorFlow package. The model takes on a sequential design, processing visual information as input throughout specialized layers to extract spatial and frequency patterns within the data.

The CNN architectures consists of these key components: Convolutional layers with Batch Normalization aids in feature extraction by identifying characteristics like edges and harmonic textures in the spectrograms and MFCC's time frequency domain, the ReLu activation function to add non-linearity representation, and pooling layers, which reduces the spatial dimensions to avoid overfitting. Fully linked (dense) layers which interpret the obtained features for classification purposes come after these layers. After this, Dropout regularization is used to improve model generalization throughout training.

The final output layer consists of an additional dense layer with a softmax activation function that turns the models' output into class probablilites of the

four chosen instruments. The Adam optimizer is used for optimization and is compiled using the categorical cross-entropy loss function, which is suitable for multi-class tasks.

The RNN models were built using the Keras API from the TensorFlow library, utilizing architectures customized specifically for sequential audio data. The models are designed to learn temporal patterns in the audio characteristics by processing time-series input data through dedicated recurrent layers.

Recurrent layers like the Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) layers, form the core foundation of the model. These layers are crucial for retaining significant temporal dynamics across audio frames. Additional variations are investigated in this study including unidirectional LSTM layers, stacked GRU layers which provide more temporal abstraction, and bidirectional recurrent layers help the models to account for future and past context within the audio sequence.

In certain scenarios, dropout layers are integrated in between recurrent layer or after to reduce overfitting and improve generalization. After filtering the recurrent outputs to obtain class-specific mappings, the dense layer uses a softmax activation function to provide probability distributions among the four classes. The models are trained on the Adam Optimizer and categorical cross-entropy loss, common for multi-class classification.

For this study, convolutional and recurrent layers are incorporated in the hybrid CNN-RNN models to capture both spatial and temporal characteristics patterns in audio characteristics. This hybrid architecture is also developed using

TensorFlow's Keras API, taking advantage of the CNN's ability to extract time-frequency characteristics with the RNN's capability to show sequential dependencies over time.

The convolutional side of the model takes care of discovering key aspects in harmonic patterns and features by processing image-like audio representation. Normalization methods such as Batch normalization is used for stability and faster convergence between the layers.

CNN-RNN models also employ dropout layers to enhance regularization and decrease overfitting. Like the other models mentioned, the closing layer is the dense layer that applies a softmax activation function. The models are compiled using the Adam optimizer and categorical cross-entropy loss function.

3.6 Experimentation Tests

This study presents an organized experimental environment with architectural modifications to evaluate and compare the performance of each machine learning model for clarinet sound identification. Three model types – CNN, RNN, and the hybrid CNN-RNN will include two distinct architectural configurations and will be trained and tested individually. Consequently, 12 independent experimental tests are created, each of which aims to explore how different model types as well as different model configurations affect classification performance whilst maintaining fixed variables like the dataset and input characteristics.

Google Colab is used for this study, utilizing its cloud-based GPU acceleration capabilities to carry out each experiment. For each experimentation scenario,

Mel spectrograms, MFCCs, and Chroma features are produced in the feature extraction stage of the implementation and are concatenated into a singular input representation.

3.6.1 CNN Experiments Architectures

The CNN experiments focus on varying depth of layers, kernel sizes and normalization techniques to evaluate performance.

- **CNN Model 1 (Lightweight CNN)**
  - 3 convolutional blocks: Conv2D (32, 64, 128 filters, kernel sizes $5\times5 \rightarrow 3\times3 \rightarrow 3\times3$)
  - Each block includes Batch Normalization, MaxPooling, and Dropout (0.2-0.35)
  - Global Average Pooling followed by Dense (128 units, ReLU)
  - Output layer: Dense (4 units, Softmax with label smoothing)
- **CNN Model 2 (Deeper CNN with Separable Convolutions)**
  - Data augmentation (RandomContrast, RandomTranslation) applied at input
  - Convolutional stack: Conv2D (64) $\rightarrow$ SeparableConv2D (128) $\rightarrow$ SeparableConv2D (128, dilation=2) $\rightarrow$ Conv2D (256)
  - Each convolution followed by Batch Normalization and MaxPooling
  - Global Average Pooling $\rightarrow$ Dense (160 units, ReLU)

  - Dropout regularization (0.2–0.35) applied before output
  - Output layer: Dense (4 units, Softmax with label smoothing)

3.6.2 RNN Experiments Architectures

The RNN models investigate the performance of multiple sequence modelling methods and recurring frameworks.

- **RNN Model 1 (Stacked LSTM)**
  - Input augmented with feature-level transformations
  - LSTM (128 units, return sequences=True) $\rightarrow$ Batch Normalization $\rightarrow$ Dropout (0.2)
  - LSTM (128 units) $\rightarrow$ Batch Normalization $\rightarrow$ Dropout (0.1)
  - Dense (128 units, ReLU) $\rightarrow$ Dropout (0.1)
  - Output layer: Dense (4 units, Softmax with label smoothing)
- **RNN Model 2 (Bidirectional GRU Stack)**
  - Bidirectional GRU (128 units, return sequences=True) $\rightarrow$ Batch Normalization $\rightarrow$ Dropout (0.1)
  - Bidirectional GRU (64 units, return sequences=True) $\rightarrow$ Batch Normalization $\rightarrow$ Dropout (0.1)
  - Bidirectional GRU (64 units) $\rightarrow$ Batch Normalization
  - Dense (160 units, ReLU) $\rightarrow$ Dropout (0.1)
  - Output layer: Dense (4 units, Softmax with label smoothing)

3.6.3 CRNN Experiments Architectures

The purpose of the CRNN models is to understand the combination of CNN depth and recurrent complexity of the RNN.

- **CRNN Model 1 (CNN + LSTM Hybrid)**
  - Convolutional stack: Conv2D (16 filters) $\rightarrow$ BN $\rightarrow$ MaxPooling (2×2) $\rightarrow$ Dropout (0.2)
  - Conv2D (32 filters) $\rightarrow$ BN $\rightarrow$ MaxPooling (2×2) $\rightarrow$ Dropout (0.2)
  - Reshape feature maps into sequences
  - LSTM (128 units) $\rightarrow$ Dropout (0.2)
  - Output layer: Dense (4 units, Softmax with label smoothing)
- **CRNN Model 2 (Frequency-only Pooling + Bidirectional GRUs)**
  - Input augmentation (RandomContrast, GaussianNoise)
  - Convolutional stack with frequency-axis pooling:
    - Conv2D (32 filters) $\rightarrow$ BN $\rightarrow$ MaxPool (1×2) $\rightarrow$ Dropout (0.15)
    - Conv2D (64 filters) $\rightarrow$ BN $\rightarrow$ MaxPool (1×2) $\rightarrow$ Dropout (0.15)
    - Conv2D (128 filters) $\rightarrow$ BN $\rightarrow$ MaxPool (1×2) $\rightarrow$ Dropout (0.20)
  - Reshape feature maps into sequences
  - Bidirectional GRU (128 units, return sequences=True) $\rightarrow$ Dropout (0.20)
  - Bidirectional GRU (64 units) $\rightarrow$ Dropout (0.20)
  - Output layer: Dense (4 units, Softmax)

Chapter 4: Analysis of Results and Discussion

4.1 Introduction to the Analysis Section

This section carefully evaluates the experimental results by comparing the three neural network types that were evaluated in this study: CNN, RNN, and CRNN. The study focuses on three important factors influencing the results: feature representation, dataset size, and model architecture. These variables are assessed not just for the performance metrics acquired (accuracy, precision, recall, and F1-score) but also in relation to other results reported in previous literature. This section goes beyond analysing metrics to address the factors behind the postiive and negative features certain architectures have in the realm of clarinet recognition.

4.1.1 Overall Results Review

For each experimentatal model evaluated, the clarinet repeatedly proved to be a complex instrument to classify, often getting misclassified for the other woodwinds of the flute and saxophone due to their shared timbral and spatial similarities. This coincides with the findings of [28] and [22], who similarly reported frequent misclassification among woodwinds. On the other hand, the trumpet came out as the most consistent instrument across all models, showing that timbral distinction overcomes pitch similarities in instrument classification tasks.

The importance of dataset size is emphasized in the CNN models. the lighter

CNN showed significant improvements when trained on the full dataset, supporting previous literature that highlights the reliance of larger datasets in convolutional approaches. RNN models, despite intended to capture temporal dependencies, did not outperform CNNs significantly in clarinet detection, signaling that temporal modelling on its own is insufficient for instruments with overlapping features, as seen by [25]. The hybrid CRNN models achieved the best overall performance, supporting existing research that suggests that hybrid architectures work better in complex polyphonic environments. Nevertheless, continuous ambiguity in clarinet recognition implies that even advanced hybrid methods are constrained when working with smaller datasets.

4.2 CNN Model Results

The Convolutional Neural Networks (CNNs) models were a crucial pillar for this research, revealing how instrument recognition can be promoted through spatial feature extraction. Each CNN model was evaluated under both 100% and 50% scenarios of the dataset. Below, is a table summarizing the metrics of the CNN experimentations.

Table 4.1: *Performance metrics of CNN Model 1 and Model 2 under 50% and 100% dataset conditions.*

Instrument	**CNN M1 (50%)**			**CNN M1 (100%)**			**CNN M2 (50%)**			**CNN M2 (100%)**		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Clarinet	0.263	0.200	0.227	0.522	0.475	0.497	0.545	0.480	0.511	0.543	0.436	0.484
Saxophone	0.304	0.222	0.257	0.539	0.440	0.485	0.536	0.476	0.504	0.550	0.568	0.559
Flute	0.571	0.267	0.364	0.537	0.644	0.586	0.619	0.578	0.598	0.589	0.589	0.589
Trumpet	0.432	0.828	0.568	0.692	0.776	0.732	0.622	0.793	0.697	0.667	0.759	0.710
Overall Acc.		0.389			0.581			0.583			0.593	

4.2.1 CNN Model 1

The lighter CNN (Model 1) emphasized the impact of dataset size. When training under 50% of the dataset, the data resulted in a low overall accuracy of 38.9%. The clarinet performed the worst across the classes achieving a precision of 0.263, a recall of 0.200, and an F1-score of 0.227. These poor results indicate the model's failure to accurately present the clarinet's unique harmonics due to insufficient data. The saxophone and trumpet achieved precision under 0.5, making the flute the only instrument having a precision of over 0.5, that of 0.571. The low scores across the classes highlight misclassification of overlapping timbre and spatial characteristics.

During training on the complete dataset, Model 1 showcased a significant improvement. Overall Accuracy rose to 58.1%, and clarinet performance doubled from the reduced data condition, achieving precision of 0.522. This result indicates that increased training examples helped the CNN's identification of the clarinet's attributes, enabling improved recognition between similar instruments. Despite a general advancement, the clarinet still trails behind the trumpet which looks to be the dominant instrument in these models achieving an F1-score of 0.732.

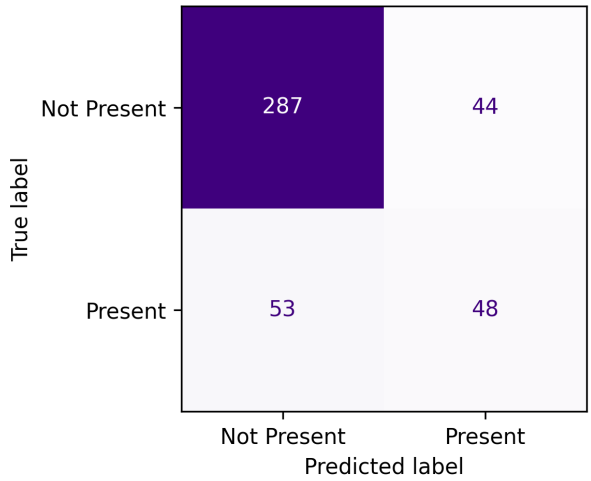

Figure 4.1: *Confusion matrix for clarinet classification using CNN Model 1 (50% dataset).*

4.2.2 CNN Model 2

As shown in Table 4.1, Model 2 achieved more consistent results and exhibting durability across changing dataset sizes. Whilst using 50% of the dataset, the model achieved an overall accuracy of 58.3%, resembling the performance of Model 1 with a complete dataset. Clarinet metrics improved under this model, achieving the highest precision across all models aswell as an F1-score of 0.511. This clear jump compared to Model 1 with the same dataset size proves that an enhancement in architecture permitted more effective feature extraction, hence improving the model's generalization with less data.

Utilizing the complete dataset, Model 2 achieved an accuracy of 59.3%. Particularly, clarinet recognition did not improve and instead dropped slightly to an F1-score of 0.484 relative to the same model with half the dataset. This can indicate declining advantages in additional data once model complexity is sufficient and could also suggest overfitting taking place when the deeper network fails to generalize more clarinet patterns and other instruments.

4.2.3 Discussion of CNN Results

The CNN experimentations shows two recurring patterns. Firstly, the dataset size significantly influenced the performance of Model 1, which resulted in poor outcomes but showed a clear improvement with the full dataset. In contrast, Model 2 showed more stability in performance, achieving very similar results irrespective of dataset size. This sensitivity to dataset size matches the findings of [22], where they showed that CNN classification accuracy greatly improved with the inclusion of additional segmentation and better data collection methods.

Secondly, clarinet recognition remained weak, similar to the other instruments among the CNN models. Confusion Matrices confirmed frequent misclassifications between the clarinet, flute, and saxophone, signalling a significant timbral overlap across woodwind instruments that the CNN models could not handle. [9] noted the same obstacles in their CNN IRMAS investigation, where instruments such as the clarinet and flute resulted as some of the lowest performing classes due to their reduced onset and spectral overlap. The addition of more data and a greater architecture did not improve performance, with F1-score remaining around the 0.500 mark. This result implies that convolutional feature extraction alone found it difficult to accurately recognize the clarinet. In contrast, the trumpet consistently got the best results, which despite sharing the same pitch with the clarinet, its unique timbral qualities as a brass instrument made it easy for the CNN Models to identify.

Collectively, the CNN manages to extract more spatial patterns with deeper architectures yet are limited in distinguishing between the clarinet, flute, and sax-

ophone. This limitation supports previous findings that CNNs, although appropriate for certain instruments, are insufficient for overlapping instruments [22]. This classifies the clarinet as one of the most challenging instrument for CNNs which emphasizes the need for alternative methodologies that incorporate temporal modeling. This will be assessed in the RNN tests.

4.3 RNN Model Results

The Recurrent Neural Networks (RNNs) were experimented to evaluate an alternative approach to the CNN by using temporal sequence modeling to enhance clarinet instrument identification. Similarly to the CNN tests, two different RNN architectures were evaluated under a 50% and 100% dataset setting. The results are summarised in the table 4.2 below.

Table 4.2: *Performance metrics of RNN Model 1 and Model 2 under 50% and 100% dataset conditions.*

Instrument	**RNN M1 (50%)**			**RNN M1 (100%)**			**RNN M2 (50%)**			**RNN M2 (100%)**		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Clarinet	0.483	0.560	0.519	0.547	0.465	0.503	0.556	0.500	0.526	0.500	0.515	0.507
Saxophone	0.493	0.571	0.529	0.529	0.512	0.520	0.426	0.460	0.443	0.508	0.536	0.521
Flute	0.382	0.289	0.329	0.440	0.367	0.400	0.367	0.244	0.293	0.518	0.478	0.497
Trumpet	0.549	0.483	0.514	0.607	0.784	0.684	0.534	0.672	0.595	0.655	0.638	0.646
Overall Acc.		0.486			0.544			0.481			0.546	

4.3.1 RNN Model 1

The lighter RNN model achieved an overall accuracy of 48.6% using half the dataset. The recognition of the clarinet was moderate, achieving precision of 0.483, a recall of 0.560, and an F1-score of 0.519. In comparison with CNN Model 1, this was a significant enhancement under identical conditions, suggesting that temporal sequence modelling increased the networks ability to spot clar-

inet specific patterns despite a constrained dataset.

The general confusion matrix (Figure 4.2) shows that clarinet audio samples were mistaken with the other woodwinds like the flute and saxophone highlighting a very common trend within the CNN and RNN. However, the clarinet achieved a greater number of correct classifications (28) relative to the CNN model using identical sample size. The clarinet specific confusion matrix (Figure 4.3) reveals that outside of the 28 correct classifications, there was 22 misclassifications, hence validating that enhancement in F1-score correlates to an increased recognition of clarinet samples.

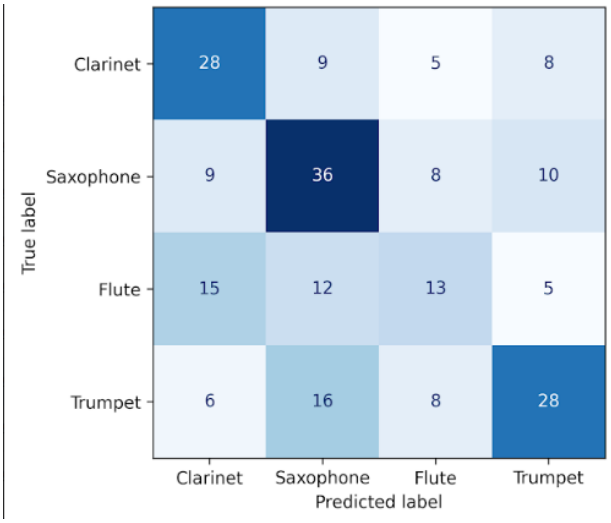

Figure 4.2: *Generic Confusion Matrix for RNN Model 1 (50% dataset).*

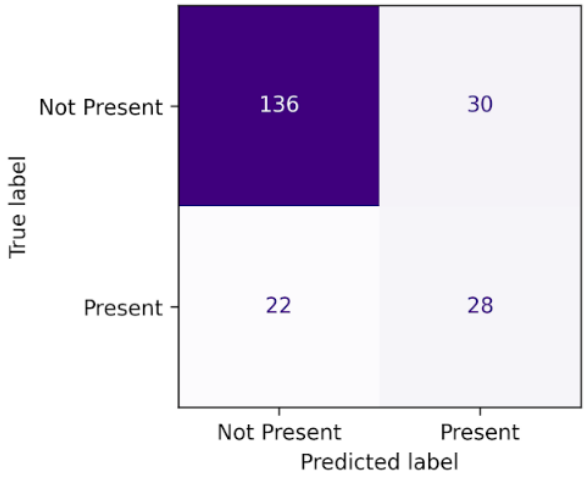

Figure 4.3: *Clarinet one-vs-rest confusion matrix for RNN Model 1 (50% dataset)*

After training with the complete dataset, Model 1 achieved an overall accuracy of 54.4%, 6% better than the model with half the dataset. The clarinet achieved better results with 0.547 precision, 0.465 Recall, and 0.503 F1-Score. In comparison to other woodwinds, the clarinet in this secnario achieved better results but is still lacking to the trumpet due to timbral differences. The clarinet specific confusion matrix (Figure 4.4) shows a trade-off between precision and recall: the model managed better accuracy in predicting the clarinet which increased precision but lacking additional clarinet samples resulting in a lower Recall.

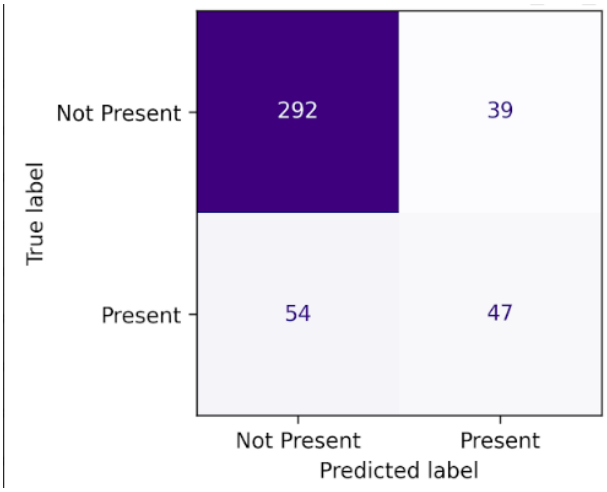

Figure 4.4: *Clarinet one-vs-rest confusion matrix for RNN Model 1 (100% dataset).*

4.3.2 RNN Model 2

Utilizing 50% of the dataset, the model achieved an overall accuracy of 48.1%, worse than RNN model 1 under identical conditions. The clarinet recognition achieved an accuracy of 0.556, a recall of 0.500, and an F1-score of 0.526, being the highest F1-score for the clarinet among the RNN results. Notably, clarinet precision exceeded that of the trumpet, indicating that deeper recurrent layers could seperate clarinet-specific characteristics from the brass timbral charactersitics of the trumpet more easily. However, this improvement resulted in a decline in recall, as the model predicted the clarinet less frequently. The higher F1-Score reflects a better balance between true and false positives instead of a concrete advancement in overall clarinet recognition.

Training Model 2 with the full dataset resulted in an improvement similar of that of Model 1. Model 2 achieved an overall accuracy of 54.6%, a small improvement over the 50% benchmark. The clarinet performed diminished, exhibiting an accuracy of 0.500, and a recall of 0.515, resulting in an F1-Score

of 0.507. The drop in performance implies that the model could not hold its superiority over the trumpet noted in the model with the reduced dataset despite the clarinets balance between precision and recall. This result alligns back with earlier patterns observed in CNN and RNN models with the trumpet standing out. The overall confusion matrix of this model (Figure 4.5) supports this trend, showing the clarinet is again being missclassified with the saxophone and flute despite the stronger recurrent architecture.

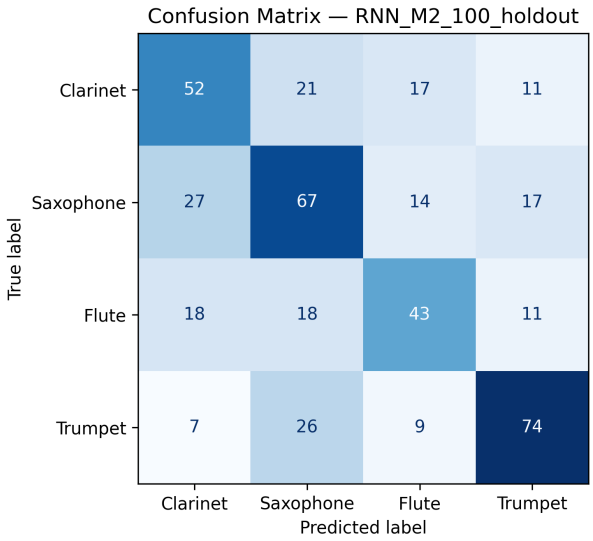

Figure 4.5: *Clarinet one-vs-rest confusion matrix for RNN Model 1 (100% dataset).*

4.3.3 Discussion of RNN Results

The RNN studies showed consistent yet minor enhancements in clarinet recognition. Both models achieved overall accuracies around the 50% mark, with only improvements of 6% in accuracy when expanding the dataset from 50% to 100%. This suggests that temporal modelling improved the stability of the networks with a lesser dataset, yet did not show a substantial improvement with increased train-

ing data. [25] observed similar limitations, where it was discussed that RNNs frequently struggle to catch certain transient sounds in smaller datasets, restricting the models capacity to generalize instruments with overlapping harmonics.

The clarinet performance averaged around 0.5 across all RNN experiments. Model 1 performed better across both datasets, whilst Model 2 improved mostly with the reduced data but not with the full dataset, suggesting recurrent layers managed to enhance recognition under limitied data conditions. This observation aligns with the study of [22], in which it was found that temporal models offered marginal improvements in identifying similar instrument classes, unless a much bigger and balanced dataset are utilized. The RNNs managed to capture sequential data but proved ineffective in precisely differentiating the woodwinds similar timbral qualities, as shown by frequent misclassifications between the clarinet, saxophone, and flute, with the trumpet being unaffected. This model highlights two main limitations, the small dataset size restricted the recurrent layers ability to extract sequential dependencies, and the lack of more sophisticated methods, such as attention based mechanisms, might have also limited the models from identifying longer range temporal features.

4.4 CRNN Model Results

Convolutional Recurrent Neural Networks (CRNNs) combine both previous types of NNs integrating spatial feature extraction with temporal sequential modeling, aiming to overcome the limitations discovered in Independent CNNs and RNNs. These models were intended to have the best overall classification performance.The

results are summarised in Table 4.3 below.

Table 4.3: *Performance metrics of CRNN Model 1 and Model 2 under 50% and 100% dataset conditions.*

Instrument	**CRNN M1 (50%)**			**CRNN M1 (100%)**			**CRNN M2 (50%)**			**CRNN M2 (100%)**		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Clarinet	0.593	0.700	0.642	0.650	0.663	0.657	0.684	0.520	0.591	0.787	0.624	0.696
Saxophone	0.633	0.603	0.618	0.652	0.736	0.692	0.671	0.746	0.707	0.707	0.696	0.702
Flute	0.537	0.489	0.512	0.678	0.678	0.678	0.667	0.578	0.619	0.808	0.656	0.724
Trumpet	0.821	0.793	0.807	0.847	0.716	0.776	0.681	0.810	0.740	0.679	0.914	0.779
Overall Acc.		0.653			0.701			0.676			0.729	

4.4.1 CRNN Model 1 Results

Across the three Model 1 architectures, the CRNN achieved the most robust results. Utilizing 50% of the dataset, the model exceeded both CNN and RNN baselines, attaining an overall accuracy of 65.3%. The clarinet exhibited a significant improvement with a precision of 0.593, recall of 0.700, and F1-Score of 0.642, a considerably higher result than the findings obtained in the CNN and RNN tests.

When trained on the entire dataset, Model 1 enhanced its its overall accuracy of 70.1% across all instruments. The clarinet performance saw an increase in precision (0.650) and decrease of (0.663), yet maintained a consistent F1-Score of 0.657. This results indicates that the clarinet kept its performance despite dataset descrepancies, in contrast to the previous models, where performance stagnated. This consistency in results indicates that Model 1 generalized clarinet-specific features without sufferring from overfitting. Other instruments exhibited consistent performance aswell, proving the efficiency of the CRNN not just in clarinet recognition but for the whole instrument group.

4.4.2 CRNN Model 2 Results

CRNN Model 2, the more complex hybrid network, yielded the best results across all experiments. The 50% data set experiment achieved an average accuracy of 67.6%, a slight increase compared to Model 1 and exceeding the best results from the CNN and RNN on a complete dataset. In this scenario, the clarinet proved to be more conservative than Model 1, obtaining a higher pecision (0.684) but a very diminsihed recall (0.520) resulting in diminished F1-Score. These metrics show as that the model reduced the number of false positives, yet lost a significant number of real clarinet samples. This trend can be shown through the clarinet-specific confusion matrix below. (Figure 4.6)

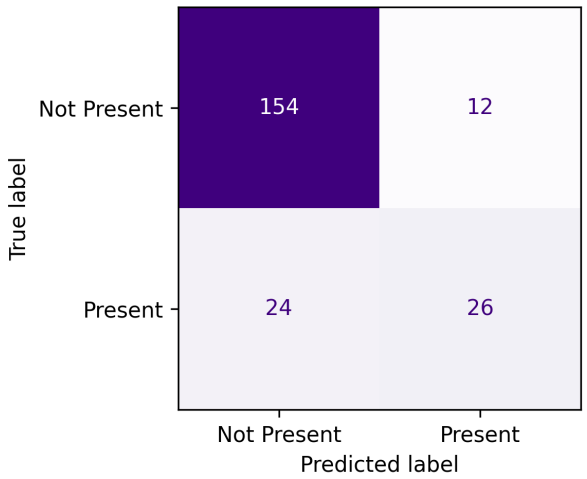

Figure 4.6: *Clarinet one-vs-rest confusion matrix for CRNN Model 2 (50% dataset).*

Model 2 with the full dataset acquired an overall accuracy of 72.9%, the highest among all experimentations. The clarinet scored a precision of 0.787, a recall 0f 0.624, and an F1-Score of 0.696.The confusion matrix below (Figure 4.7) highlights the decrease in clarinet misclassifications from the flute and

saxophone relative to other models.

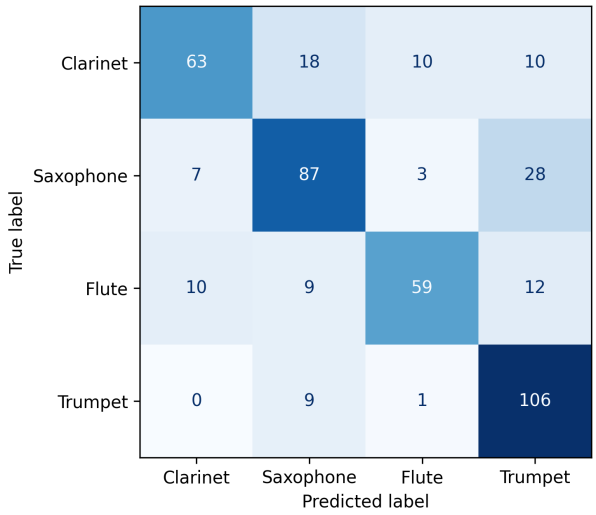

Figure 4.7: *General confusion matrix for CRNN Model 2 (100% dataset).*

4.4.3 Discussion of CRNN Results

The CRNN models clearly outperformed individual CNNs and RNNs, confirming the efficacy of combining spatial and temporal processing. Both architectures surpassed the 65% mark and improved upon that with the larger dataset, overcoming the issue found in the baseline models. These results are backed up by [26], who showed that similar CNN-BiGRU models obtained much better accuracy compared to independent architectures, particularly in a polyphonic environment. Similarly, other prior research confirmed the combination of convolutional and recurrent layers is efficient in identifying the clarinet against other overlapping instruments.

Model 1 displayed consistent results where as Model 2 attained the maximum accuracy across all experimentations. The clarient one-vs-rest confusion matrix for the 50% dataset shows fluctuations between precision and recall, and the 100% general matrix presented less misclassifications across the instruments, signifying

an enhancement in clarinet recognition. Despite the improvement, some miss-classifications still persisted, supporting the research of [28] that the clarinet and other woodwinds are fundementally challengins to distinguish due to their similar attributes.

CRNNs achieved the best results for this task, cutting the gap between trumpet recognition and reducing woodwind confusions, which allowed the clarinet to achieve better recognition. Nonetheless, limitations still persisted with hybrid models. Once again, the small dataset prevented the generlization of the models, and the reliance on Google Colab's limited computational resources is a major factor that prevented further testing of deeper, complex architectures, which could further mitigate woodwind confusion. Despite these drawbacks, the results clearly show that CRNNs are the most feasible architecture for clarinet recognition in the context of this study.

Chapter 5: Conclusions and Recommendations

5.1 Summary of the Study

This Study attempted to address the challenge of clarinet sound identification within the field of Music Information Retrieval (MIR). Despite recent developments through deep learning that have enhanced musical instrument detection, certain instruments like the clarinet still presents challenges due to its similar characteristics with other woodwind instruments like the flute and saxophone. The study attempted to examine if several neural network systems can improve the accuracy of clarinet recognition and investigate the impact on model performance dataset sizes have.

The experimental framework was guided by three research questions:

- RQ1: Can neural network algorithms accurately recognize a clarinet sound from multiple instruments with similar pitch and frequency?
- RQ2: Which machine learning algorithm handles frequency and pitch similarities the best when isolating the clarinet from the other instruments?
- RQ3: Does the size of the dataset effect how well the machine learning algorithm can recognize a clarinet sound?

The study utilized a subset of the IRMAS dataset containing audio clips from four instruments: Clarinet, Flute, Saxophone, and Trumpet. Two dataset sizes

were used, one complete set (100%) and half the dataset (50%). This was applied by using a stratfied sampling method to generate the datasets and reduce instrument bias. All audio clips were processed and normalized before feature extraction. Three feautre representations were employed: Mel-Spectrogram, MFCCs, and Chroma Features. This representations were concatenated to create standardizes input features across all models.

Three types of Neural Network models were assessed: Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and hybrid Convolutional Recurrent Neural Networks (CRNNs). Each variant included two different architectures, for a total of six models. Training and evaluation were conducted on Google Colab utilizing its GPU acceleration, with performance measured through metrics including: precision, recall, F1-Score, and confusion matrices.

The results indicated that the clarinet is one of the most difficult instrument to recognize. CNNs exhibited a notable fluctuation in results due to dataset size, with the deeper model providing better generalization despite still continuing to experience issues in distinguishing the clarient, primarily from the other woodwinds. RNNs provded marginally better results through temporal patterns, however these results were not consistent across the two datasets. The CRNNs easily surpassed the CNN and RNN baseline models, underlining the importance of integrating temporal patterns with spatial features extraction. CRNN Model 2 obtained the greates overall score, confirming the usefulness of hybrid architectures in clarinet sound identification.

The study determined how well neural networks are able to distinguish the

clarinet, although the performance is dependent on dataset size, model architecture, and toning of the concatenated feature representation. The findings provide empirical evidence that hybrid models offer the best results, whilst highlighting the persistent issues of separating the clarinet from acoustically similar instruments.

5.2 Discussion of Key Findings

The findings of this research offer important insights into the fundemental issue of clarinet recognition in MIR. Despite recent advancements, it remains a challenge due to its acoustic resemblence to other instruments. An analysis of CNN, RNN, and CRNN models with different dataset sizes yielded key results relevant to the research topics presented.

5.2.1 Research Question 1

The first research question investigated the capacity neural networks have to correctly identify clarinet sounds amongst other instruments possessing similar characteristics. The results indicate that neural networks are able to identify clarinets to some level, yet their accuracy is dependent on architecture and training conditions. Convolutional Neural Netowrks and Recurrent Neural Networks encountered issues in isolating the clarinet primarily from the flute and saxophone, the most similar instruments, highlighting the spectral and timbral parallels identified in previous research. The CRNN models showed a considerable improvement, indicating that the combination of spatial and temporal analysis works better for clarinet recognition. These results reaffirm the clarinet's classification as an elabo-

rate topic within MIR and suggest that despite the advancements neural networks have made, there is always room for improvement.

5.2.2 Research Question 2

The second research question builds upon the first and discusses which model is most suited for handling overlapping acoustic characteristics of instruments. The analysis between the CNNs, RNNs, and CRNNs show that the hybrid models were the most effective under the study's chosen environment. CNNs displayed competence from the spatial data gathered from the concatenated Mel Spectrograms but were sensitive to dataset fluctuations and suffered in identifying clarinet-specific features. RNNs exhibited limited improvements in predicting temporal sequences, despite being less prone to dataset limitations, its overall accuracy did not improve due to the timbral similarities overlapping of the instruments being too strong for the RNN models. On the contrary, CRNN's incorporation of both spatial and temporal sequencing led to an enhancement not just in clarinet recogniton, but also in its overall performance, achieving the best result of the study. This result backs up existing research arguing in favor of hybrid models for complicated audio recognition tasks, and the findings of this study add to the case that CRNNs are a primary method for clarinet recognition.

5.2.3 Research Question 3

The third research question focuses on the impact clarinet recognition takes when using different sample sizes. The results show that dataset size can influence the models performance, as significantly seen in the CNNs, where a bigger dataset

improved outcomes massively. For RNNs, dataset sizes provided smaller discrepancies, indicating that temporal modelling is more suited to handle limitations associated with smaller sample sizes. CRNNS displayed the most consistent results, achieving superior performance under both dataset samples. These different findings across all three models confirm the importance of dataset capacity for clarinet recognition, but also revealing the need for balanced methodologies, whereby, the appropriate size of the dataset is dependent on the choice of model architecture.

5.2.4 Concatenated Feature Representation

This study's unique aspect was the use of concatenated feature representation that included Mel Spectrograms, MFCCs, and Chroma features. Despite not being the main study investigation, the results suggest concatenation improved the input data and aided in the effectiveness of the CRNNs. This approach is still mostly unexplored within the MIR field, as a large amount of research focuses on a singular form for input representation. Concatenated feature representation is able to capture a wide selection of acoustic information, hence facilitating improved classification for the clarinet and other instrument similarities.

5.3 Limitations of the Study

This study, despite its strengths, has various limitations that must be acknowledged.

The quantity and balance was a major limitation of the study. While the IRMAS dataset provided a reliable foundation, the subsets utilized had a limited

number of samples for each instrument, especially for the clarinet. This issue hampered the models' potential for better instrument generlization and increased their vulnerability to class imbalances.

Additionally, employing a singlular dataset guaranteed that all experimentations were performed under fixed conditions. This lowers the external validity of the outcomes, as the models were not trained or tested with other datasets which contain different authentic recordings that include other levels of background noise, recording equipment, and performance conditions.

The experimentation environment posed another unfortunate limitation. Training and testing were carried out on Google Colab, which altough is partially free, it has restrictions on GPU time allocation, memory space, and session consistency. These constraints limited the amount of experimentations that could be achieved, affected the range of architectural changes that were investigated, which resulted in the use of more simpler models, and the affordability to fine-tune hyperparamter optimization.

This leads to the limitation in the architectual scope. While a variety of CNN, RNN, and CRNN archtiectures were employed, more complex and sophisticated frameworks such as attention mechanisms, or Transformer-based models were not explores. These might handle the timbral conflicts among the instruments more efficiently as well as the handling of the robustness of the concatenated representation of feature extraction.

Ultimately, the selection in feature extraction could have hindered the models' performance. The research used a combination of Mel Spectrograms, MFCCs,

and Chroma features. This enhancement in input increased its dimensional complexity, potentially leading to overfitting given the size of the dataset. Tuning of this feature representation, such as a different choice of feature inputs, or enhanced augmentation to highlight the different features within the concatenated image might mitigate this limitation.

5.4 Recommendations for Future Work

Based on the findings and limitations of the study, various suggestions for future research on clarinet sound identification and the wider recognition of musical instruments could be given.

An intriguing avenue is the methodological exploration of concatenated feature representatinon for instrument recogntion. This work combined Mel Spectrograms, MFCCs, and Chroma features into a single input, generating more robust results. Although this methodology has already been researched in other areas, such as voice recognition, it is still barely researched within the realm of instrument recognition. Each feature representation captures unique acoustic data, ranging from spectral details, timbral features, and harmonic contents of the instrument, their combination offers possibilities for more intricite models. Future research could investigate further advanced techniques of this feature concatenation, including dimensionality reduction to address overfitting, different choices of feature representations to pick the most significant features, or a comparison of different concatenated frameworks to further enhance the best choice of feature representation. These techniques can show enhanced results in further studies re-

garding the clarinet and other instruments.

Apart from feature concatenation, the enlargement and diversity of the dataset is significant. Bigger and balanced datasets that include audio recordings from various environments and performances could improve generlization and limit class imbalance. Data augmentation techniques such as pitch shifting and addition of background noise can mimick real world scenarios which makes the dataset more reliable. Future researchers may also incorporate their own dataset by capturing their own musical recordings through their personal instrument. This, then can be tested and compared with other datasets to further improve dataset requirements for instrument recognition.

Furthermore, modern architectures like attention-based CRNNs or Transformer based models deserve further research. These models have proven to be efficient in other audio classification tasks and may result in additional gains when used for clarinet recognition with similar sounding instruments. These advanced models could be evaluated within real-world implementations of clarinet identification. Embedding recognition models into educational environments or integrating it with interactive music systems would reinforce its practical value and strengthen the field of MIR technologies.

5.5 Final Remarks

This dissertation ends by acknowledging that within the Music Information Retrieval field, clarinet recognition is among the most complex. This study does not serve to deliver a complete answer; yet, it offers a progressive path, show-

casing how model selection and experimentation environment can bridge the gap between present limitations and future possibilities.

The importance of this study is not just in its outcomes, but also in the opportunities it brings. Contributing to the ongoing research in MIR builds a foundation for future developments, aiming for more reliable clarinet recognition systems and improved applications that integrate creativity, technology, and music.

List of References

[1] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, "Content-based music information retrieval: Current directions and future challenges," *Proceedings of the IEEE*, vol. 96, no. 4, pp. 668–696, 2008.

[2] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[3] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proc. 26th Int. Conf. Machine Learning (ICML)*, 2009, pp. 609–616.

[4] D. Herremans, C.-H. Chuan, and E. Chew, "A functional taxonomy of music generation systems," *ACM Computing Surveys*, vol. 50, no. 5, p. 69, 2017.

[5] S. Essid, G. Richard, and B. David, "Musical instrument recognition by pairwise classification strategies," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1401–1412, 2006.

[6] A. Livshin and X. Rodet, "Musical instrument identification in continuous recordings," in *Proc. 7th Int. Conf. Digital Audio Effects (DAFx-04)*, Naples, Italy, 2004, pp. 222–227.

[7] E. J. Humphrey, J. P. Bello, and Y. LeCun, "Feature learning and deep architectures for music information retrieval," *IEEE Transactions on Multimedia*, vol. 16, no. 2, pp. 366–376, 2013.

[8] A. Bazzica, J. C. van Gemert, C. C. S. Liem, and A. Hanjalic, "Vision-based detection of acoustic timed events: a case study on clarinet note onsets," *arXiv preprint arXiv:1706.09556*, 2017. [Online]. Available: https://arxiv.org/abs/1706.09556

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

[10] A. Blaszkiewicz, M. Greszczuk, and M. Śliwiński, "Musical instrument identification using deep learning approach," *Sensors*, vol. 22, no. 8, p. 3033, 2022.

[11] R. Chen, A. Ghobakhlou, and A. Narayanan, "Interpreting CNN models for musical instrument recognition using multi-spectrogram heatmap analysis: A preliminary study," *Frontiers in Artificial Intelligence*, vol. 7, p. 1499913, 2024.

[12] K. Avramidis, A. Kratimenos, C. Garoufis, A. Zlatintsi, and P. Maragos, "Deep convolutional and recurrent networks for polyphonic instrument classification from monophonic raw audio waveforms," *arXiv preprint arXiv:2102.06930*, 2021. [Online]. Available: https://arxiv.org/abs/2102.06930

[13] F. C. de la O, F. F. de Vega, and F. J. R. Díaz, "Analyzing quality clarinet sound using deep learning: A preliminary study," *IEEE Access*, vol. 10, pp. 82 853–82 861, 2022.

[14] D. Chatterjee, A. Dutta, D. Sil, and A. Chandra, "Deep single shot musical instrument identification using scalograms," *arXiv preprint arXiv:2108.03569*, 2021. [Online]. Available: https://arxiv.org/abs/2108.03569

[15] M. Yun and J. Bi, "Deep learning for musical instrument recognition," Project Report, University of Rochester, 2017. [Online]. Available: https://hajim.rochester.edu/ece/sites/zduan/teaching/ece477/projects/2017/MingqingYun_JingBi_ReportFinal.pdf

[16] M. S. Nagawade and V. R. Ratnaparkhe, "Musical instrument identification using MFCC," in *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, May 2017, pp. 2198–2202.

[17] Magenta Project, "NSynth dataset: Neural audio synthesis," 2017, dataset. [Online]. Available: https://magenta.tensorflow.org/nsynth

[18] T. Heittola, A. Klapuri, and T. Virtanen, "Musical instrument recognition in polyphonic audio using source-filter model for sound separation," in *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, 2009, pp. 327–332. [Online]. Available: https://ismir2009.ismir.net/proceedings/OS3-2.pdf

[19] S. Gururani, M. Sharma, and A. Lerch, "An attention mechanism for musical instrument recognition," *arXiv preprint arXiv:1907.04294*, 2019. [Online]. Available: https://arxiv.org/abs/1907.04294

[20] C. R. Lekshmi and R. Rajan, "Predominant instrument recognition in polyphonic music using convolutional recurrent neural networks," in *Proceedings of the 15th International Symposium on Computer Music Multidisciplinary Research (CMMR)*, Online, November 2021, pp. 185–193.

[21] A. Solanki and S. Pandey, "Music instrument recognition using deep convolutional neural networks," *International Journal of Information Technology*, vol. 11, no. 2, pp. 1–12, 2019.

[22] C. Relkar and V. Tejwani, "Musical instrument identification using machine learning," *International Journal of Multidisciplinary Research in Science, Engineering and Technology*, vol. 2, no. 9, pp. 1825–1830, September 2019.

[23] B. L. Sturm, M. Morvidone, and L. Daudet, "Musical instrument identification using multiscale mel-frequency cepstral coefficients," in *Proceedings of the 18th European Signal Processing Conference (EUSIPCO)*, Aalborg, Denmark, August 2010, pp. 477–481.

[24] R. Loughran, J. Walker, M. O'Neill, and M. O'Farrell, "The use of mel-frequency cepstral coefficients in musical instrument identification," *Journal of New Music Research*, vol. 37, no. 3, pp. 231–244, 2008.

[25] L. Wyse and M. Huzaifah, "Conditioning a recurrent neural network to synthesize musical instrument transients," in *Proceedings of the 16th Sound and Music Computing Conference (SMC)*, Malaga, Spain, June 2019.

[26] M. Ashraf, F. Abid, I. U. Din, J. Rasheed, M. Yesiltepe, S. F. Yeo, and M. T. Ersoy, "A hybrid CNN and RNN variant model for music classification," *Applied Sciences*, vol. 13, no. 3, p. 1476, 2023.

[27] D. de Benito-Gorrón, A. Lozano-Diez, D. T. Toledano, and J. González-Rodríguez, "Exploring convolutional, recurrent, and hybrid deep neural networks for speech and music detection in a large audio dataset," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2019, no. 9, pp. 1–18, May 2019.

[28] J. J. Bosch, J. Janer, F. Fuhrmann, and P. Herrera, "A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals," in *13th International Society for Music Information Retrieval Conference (ISMIR)*, 2012, pp. 559–564.

Chapter A: Sample Code

Below is the link to the GitHub project. It contains all the Google Colab jupyter notebooks that have all the source code for this dissertation.

GitHub link: https://github.com/lukemifsud/DissertationColabCode