Musical Instrument Recognition Using Their Distinctive Characteristics
in Artificial Neural Networks

Babak Toghiani-Rizi
bato9963@student.uu.se
Department of Information Technology
Uppsala University

Marcus Windmark
mawi2661@student.uu.se
Department of Information Technology
Uppsala University

***Abstract*—In this study an Artificial Neural Network was trained to classify musical instruments, using audio samples transformed to the frequency domain. Different features of the sound, in both time and frequency domain, were analyzed and compared in relation to how much information that could be derived from that limited data. The study concluded that in comparison with the base experiment, that had an accuracy of 93.5%, using the attack only resulted in 80.2% and the initial 100 Hz in 64.2%.**

I. Introduction

Musical instruments come in a wide spectrum of shapes and sizes and the characteristics of one's sound can just as well be distinct or similar to another instrument [1].

The materials it was built with, the aging of its material, how the material was processed and even the style of the musician playing it may have an impact on how it sounds. An amateur comparing two instruments, built with different materials, may hear a difference between them, yet is still able to determine that they are the same type of instrument.

In this study, machine learning techniques were used to compare different characteristics of musical instruments and study their ability to distinguish a range of instruments. This was performed using the frequency spectrum of the audio signal, together with an Artificial Neural Network [2] (ANN), comparing the accuracy of the network in the cases of the following experiments:

- The whole sample
- The attack of the sound
- Everything but the attack of the sound
- The initial 100 Hz of the frequency spectrum
- The following 900 Hz of the frequency spectrum

II. Feature Selection

The process of feature selection is an essential part to be able to use all information contained in the data. This section outlines the different possible use cases of the frequency spectrum that can be used as feature vectors.

A. Base Experiment

The feature vector that was used for the base experiment was constructed using the frequency spectrum of the original audio sample. This spectrum was then represented as partitions, described further in section III-A4 about pre-processing the data, and the overall properties of it was used to classify the instruments.

The reason for using the representation of frequency domain as the basis for the feature vector, rather than the time domain, was because the frequency spectrum is a better way to represent and compare audio signals in, without taking length into consideration. Seen in figure 1 is an example of how different signals differ in the frequency spectrum of the signal. The frequency spectrum can be used as a discrete way of representing something that in reality is continuous, simplifying the process of computing calculations and doing pre-processing [3].

B. Instrument Attack

Many features characterize instruments and one important aspect is the attack transient. As shown in figure 2, the transient of an instrument consists of several parts. Starting from the onset point, the attack is the significant rise of the sound, ending in a longer decay period.

Clark [4] performed a study on the importance of the different parts of a tone for human recognition and concluded that having only the attack resulted in a good accuracy of recognizing most instruments. The

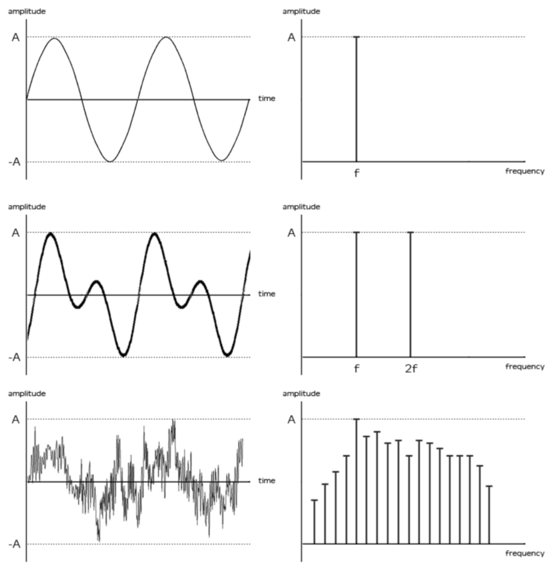


Fig. 1. A comparison of representing different signals in time and frequency domain.

hypothesis of this study was that the same would apply for an automated system and analyzing the importance of the attack by having only the attack as well as not having it at all.

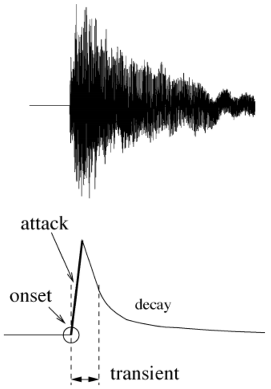


Fig. 2. The beginning of the sound of an instrument consists of an onset, attack, transient period and decay. [5]

C. Low and High Frequencies

Following the standard of the Equal-tempered scale as described by Michigan Technology University [6], all of the audio samples used to train the network were in the fourth octave, ranging the frequency spectrum of the tones between 261.63 Hz (C_4) and 493.88 Hz (B_4) (see table I). In order to study a larger spectrum than just the tone frequency, and at the same time limit the scope of the experiment, only the frequencies ranging between 1 and 1000 Hz were used to construct the feature vector.

Tone	**Frequency (Hz)**
C_4	261.63
$C_4^\#$	277.18
D_4	293.66
$D_4^\#$	311.13
E_4	329.63
F_4	349.23
$F_4^\#$	369.99
G_4	392.00
$G_4^\#$	415.30
A_4	440.00
$A_4^\#$	466.16
B_4	493.88

TABLE I
FREQUENCY RANGE OF THE TONES IN THE SAMPLES USED TO TRAIN THE NETWORK.

III. DATASET

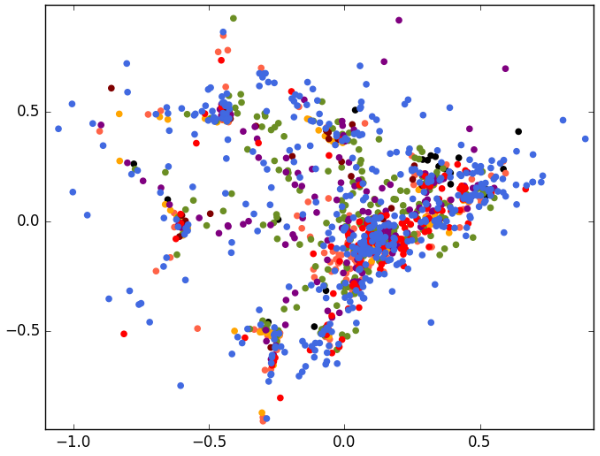


Fig. 3. Principal Component Analysis plot - showing Banjo in black, Cello in pale red, Clarinet in orange, English horn in green, Guitar in maroon, Oboe in purple, Trumpet in red and Violin in royal blue.

The dataset used in this report was the London Philharmonic Orchestra dataset [7], consisting of recorded samples from 20 different musical instruments. For each instrument, the samples range over its entire set of tones played in every octave with different levels of strength (*piano*, *forte*) and length. In addition to that, the dataset also includes samples where different playing techniques are used with the instrument, such as *vibrato*, *tremolo*, *pizzicato* and *ponticello*.

In order to limit the scope of this project, the following eight instruments were selected to train the model: Banjo, Cello, Clarinet, English horn, Guitar, Oboe, Trumpet and Violin. This set of instruments was chosen because of the high quality of the samples and them ranging over the three instrument families Brass, String and Woodwind. The number of samples of each instrument is shown in table II.

To avoid handling potential different harmonics in the same tone across the octaves, only the samples of recordings done in the fourth octave were used.

Index	**Instrument**	**Samples**
1	Banjo	23
2	Cello	166
3	Clarinet	131
4	English horn	234
5	Guitar	29
6	Oboe	155
7	Trumpet	140
8	Violin	366
Total		**1244 samples**

TABLE II
DISTRIBUTION OF THE INSTRUMENT SAMPLES IN THE DATASET.

A. *Pre-processing of Base Experiment*

Before training the network, the audio samples were pre-processed, as discussed in section II. Initially, since the audio samples in time domain are continuous (as shown in fig. 4), they had to be transformed into a discrete representation.

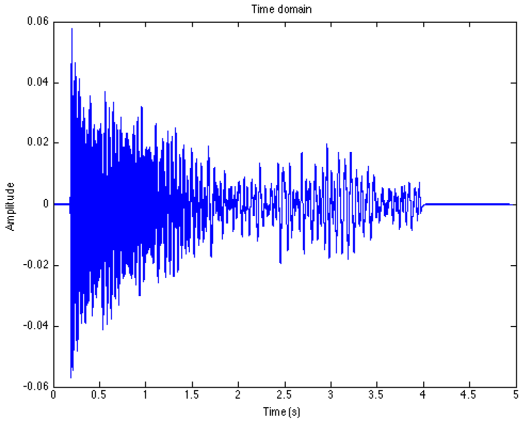


Fig. 4. Unprocessed sample in time domain of a guitar playing the tone E4.

1) Fast Fourier Transform: The first step of the pre-processing consisted of transforming the audio sample from time domain to frequency domain by using the Fast Fourier Transform [8] (FFT), resulting in a frequency spectrum (see example in fig. 5). The spectral components played an essential part in the further steps creating the feature vector.

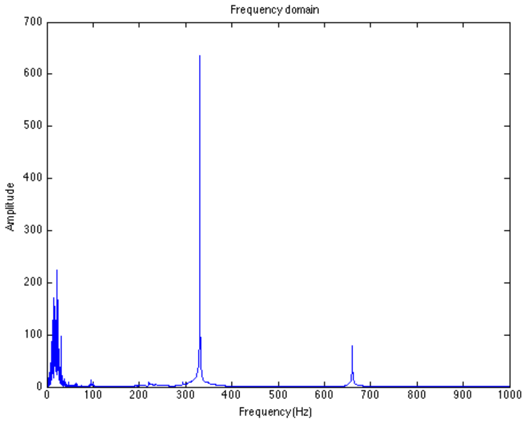


Fig. 5. The sample in figure 4 transformed to the frequency domain.

2) Spectrum Cut: As motivated in section II-C, the resulting frequency spectrum was cut off for all frequencies above 1000 Hz, leaving the range of 1-1000 Hz.

3) Frequency shift: Since the purpose of network was not to learn, or even take into consideration, that each different tone has different base frequencies, all transformed samples were also shifted to be represented in the pitch of an A_4 (440 Hz). An example is shown in fig. 6. This resulted in audio samples regardless of tone having a similar representation and therefore minimizing the risk of the network learning to classify different tones rather than classifying musical instruments.

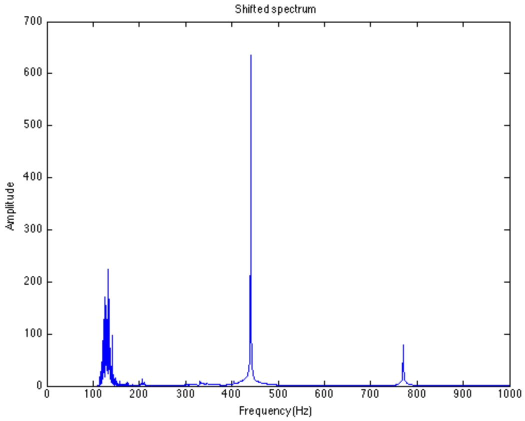


Fig. 6. The audio sample with a shifted spectrum.

4) Partitioning the Frequency Spectrum: Having the frequency spectrum as basis, it was partitioned into ranges of frequencies, avoiding the inefficient case that would have been one feature per hertz. The process of partitioning was also to avoid a potential risk of overfitting the model. The spectrum was divided into 50 sections and each section was represented by the average frequency of that range, as seen in fig. 7.

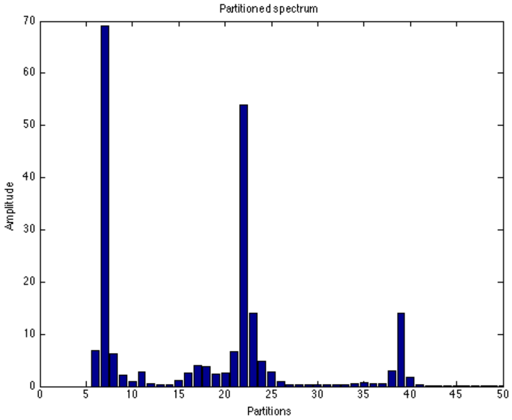


Fig. 7. The partitioned frequency spectrum, where each section represented the average amplitude of that frequency range.

5) Normalization: In order to train a network on the co-relation between the amplitude tops of a sample and not the actual values, the data was normalized according to the feature scaling method, scaling the range of amplitudes between [0,1] (see eq. (1)). The normalization resulted in an emphasized co-relation of a sample's amplitudes across the frequency spectrum, allowing each sample to equally contribute to the training of the model (see fig. 8).

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \qquad (1)$$

B. Pre-processing of Comparative Experiments

Extending the pre-processing of the base experiment described in section III-A, the pre-processing of the comparative datasets slightly differed in order to be used in the different experiments.

1) Attack experiment: To produce the feature vectors used in the experiments focusing on the attack, a specific technique of attack extraction was performed. This process was made in the time domain, before the transformation using FFT.

To find the onset point of the audio signal, it was divided into windows, each consisting of 10 ms of data. For each window, the Root Mean Square (RMS) energy was calculated and summed up. Using these discrete energy partitions, each was compared to the RMS energy of the overall signal and the onset was defined as the point where this energy was 10 dB over the signal average.

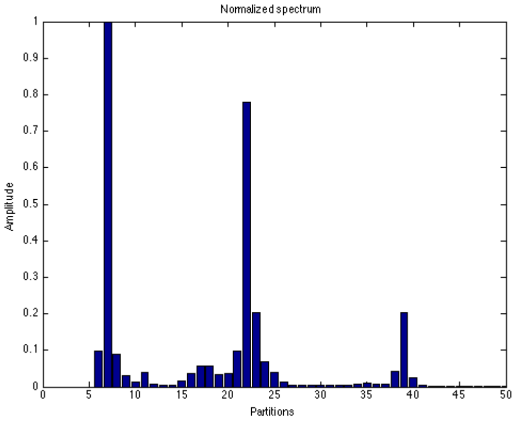
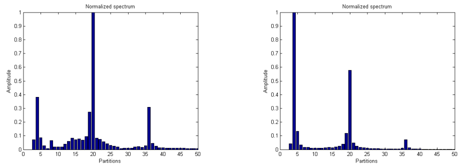


Fig. 8. Frequency domain

There are many algorithms for finding the true length of the transient period (or attack), but considering that the diverse set of instruments in this project would have resulted in finding the onset to be a project on its own, the best alternative was to have a fixed transient length. Having a transient length of around 80 ms was shown in a study by Iverson [9] to capture the important aspects of the onset in experiments with human subjects. In this project though, using a little longer transient length proved to work better, so the fixed transient length used was 100 ms.

In the experiment without an attack, the start of the steady period was shifted to be 200 ms after the end of the attack. This was done to be sure to totally exclude all characteristics from the attack.

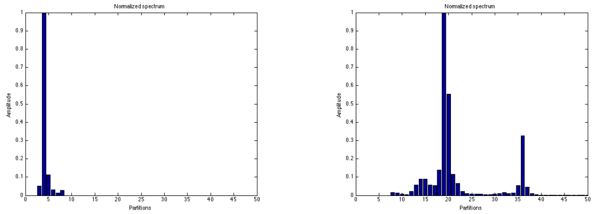
A comparison of the frequency spectrum of both the attack and without attack can be seen in fig. 9.



Only attack *Without attack*

Fig. 9. Comparison plots of feature vectors with or without attack.

2) Frequency spectrum experiment: The difference between the pre-processing of the base experiment and the frequency spectrum experiment was that the amplitudes outside of the range considered in the experiments were zeroed. In the case of the experiment containing the first 100 Hz, only this range of the data was preserved, similarly for the experiment with the following 900 Hz.



The first 100 Hz *The following 900 Hz*

Fig. 10. Comparison plots of feature vectors with initial 100 Hz and the following 900 Hz.

IV. Training

The model used for training was a Multilayer Perceptron [10] (MLP) with Early Stopping [11], using Resilient Back Propagation [12] (Rprop) as the learning heuristic. The network consisted of 50 inputs, 30 hidden nodes and eight different outputs, each representing one of the instruments.

A. Learning Algorithm

Resilient Back Propagation is a learning heuristic used for supervised learning. It takes only the sign of the partial derivative over all patterns into account (see eq. (2)), and then acts independently on each weight update, shown in eq. (3).

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + \Delta w_{ij}^{(t)} \tag{2}$$

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t)} & , \text{ if } \frac{\partial E}{\partial w_{ij}}^{(t)} > 0 \\ +\Delta_{ij}^{(t)} & , \text{ if } \frac{\partial E}{\partial w_{ij}}^{(t)} < 0 \\ 0 & , \quad \text{else} \end{cases} \tag{3}$$

B. Multilayer Perceptron

Multilayer Perceptrons [10] (MLP) are feed forward ANNs where sets of inputs are mapped to appropriate outputs using layers of nodes in a directed graph (see fig. 11). The nodes in the layers consist of perceptrons, binary classifiers with a function deciding whether an input belongs to one class or the other using eq. (4) to eq. (7).

$$y = f_h(S) \tag{4}$$

$$f_h(S) = \begin{cases} 1, S > 0 \\ 1, S \leq 0 \end{cases} \tag{5}$$

$$S = \sum_{n}^{i=1} w_i x_i - \theta = \tag{6}$$

$$= \sum_{n}^{i=0} w_i x_i \quad \text{where} \quad \begin{cases} x_0 = -1 \\ w_0 = \theta \end{cases} \tag{7}$$

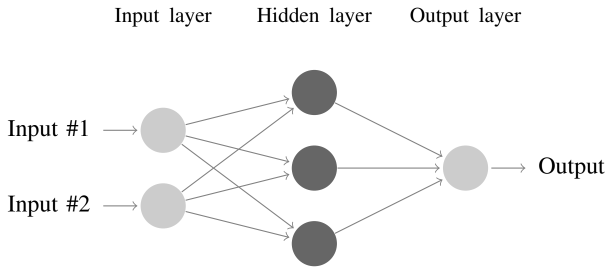


Fig. 11. Basic MLP setup

C. Hidden Nodes

Using 30 hidden nodes in a network could in some cases lead to inefficiency and overfitting, but due to the fact that the amplitude in the frequency spectrum had large variances and that the data was normalized, many of the elements would be close to 0. Therefore, the risk of overfitting is greatly reduced, as many of the inputs would most likely not have a large effect on the error, and thus necessarily not negatively influence the weight changes in the training.

Data split	**Size**
Training data	60%
Validation data	20%
Test data	20%

TABLE III
How the data was distributed in Early Stopping

D. Early Stopping

Early Stopping [11] is a regularization method to avoid overfitting by splitting the data into three subsets: training, validation and test. The network uses the training set to improve its performance while validating with the validation set, up to the point that the validation error is increasing. It then retrains the network with the training data up to the epoch where the validation error was at a minimum, and then uses the test set to test its performance.

To optimize the training accuracy and to avoid overfitting, the network was trained using Early Stopping [11]. The maximum fail parameter was set to 150 epochs and the total numbers of epochs were set to 500. The dataset was then split as shown in table III.

V. Experimental Results

As stated in section II about the feature selection, five different experiments were performed. A summarized result of the accuracy is shown in table IV. In this section, we are going to present the results from the experiments and then have a further discussion in section VI.

The accuracy of each experiment was measured using an average of 10 runs of each experiment.

Experiment	**Accuracy**
Base experiment	93.5%
Only attack	80.2%
Without attack	73.2%
First 100Hz	64.2%
Following 900Hz	90.6%

TABLE IV
THE AVERAGE OF SIX TIMES RESULT OF THE FIVE DIFFERENT EXPERIMENTS RANGED FROM 66% TO 94%.

A. Base Experiment

Training the network with the base feature vector resulted in an average accuracy of 93.5% and a sample confusion is shown in fig. 12. An example of the error performance of the early stopping algorithm is shown in fig. 13, stopping the training of the network after 207 epochs.

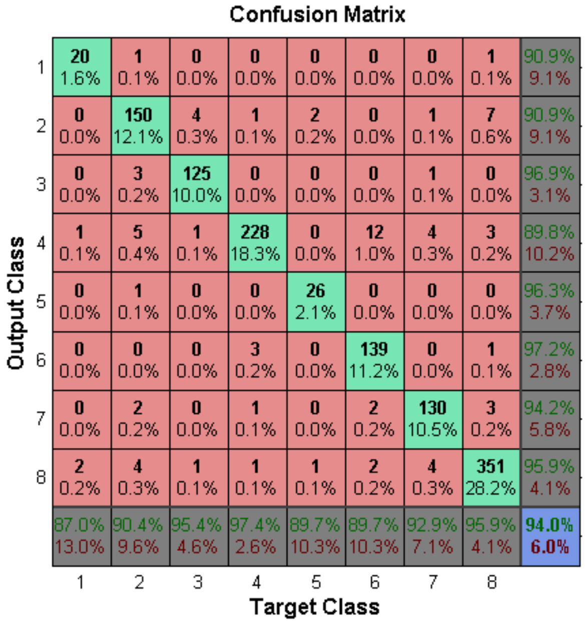
B. Impact of the Attack

The network was trained with the two different aspects of feature vectors either containing only the attack or excluding it. The feature vector using only the attack resulted in an accuracy of 80.2%, as seen in fig. 14, and the result of excluding the attack had an accuracy of 73.2%, shown with an example in fig. 15.

C. Impact of the Frequency Cut

The network was trained with two different versions of feature vectors, one with the initial 100 Hz and another with the following 900 Hz.

Training the network with the initial 100 Hz resulted in an accuracy of 64.2% (fig. 16), in comparison to training the network with the following 900 Hz, which resulted in an accuracy of 90.6% (see fig. 17).



	1	2	3	4	5	6	7	8	
1	20 1.6%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	90.9% 9.1%
2	0 0.0%	150 12.1%	4 0.3%	1 0.1%	2 0.2%	0 0.0%	1 0.1%	7 0.6%	90.9% 9.1%
3	0 0.0%	3 0.2%	125 10.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	96.9% 3.1%
4	1 0.1%	5 0.4%	1 0.1%	228 18.3%	0 0.0%	12 1.0%	4 0.3%	3 0.2%	89.8% 10.2%
5	0 0.0%	1 0.1%	0 0.0%	0 0.0%	26 2.1%	0 0.0%	0 0.0%	0 0.0%	96.3% 3.7%
6	0 0.0%	0 0.0%	0 0.0%	3 0.2%	0 0.0%	139 11.2%	0 0.0%	1 0.1%	97.2% 2.8%
7	0 0.0%	2 0.2%	0 0.0%	1 0.1%	0 0.0%	2 0.2%	130 10.5%	3 0.2%	94.2% 5.8%
8	2 0.2%	4 0.3%	1 0.1%	1 0.1%	1 0.1%	2 0.2%	4 0.3%	351 28.2%	95.9% 4.1%
	87.0% 13.0%	90.4% 9.6%	95.4% 4.6%	97.4% 2.6%	89.7% 10.3%	89.7% 10.3%	92.9% 7.1%	95.9% 4.1%	**94.0%** **6.0%**

Fig. 12. The confusion matrix from one of the training sessions of the base experiment.

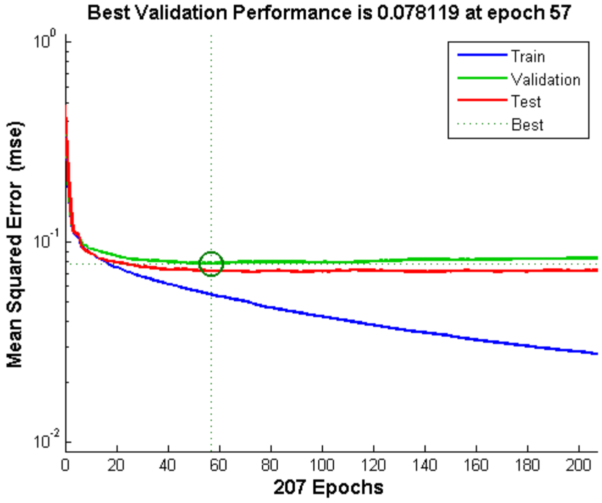
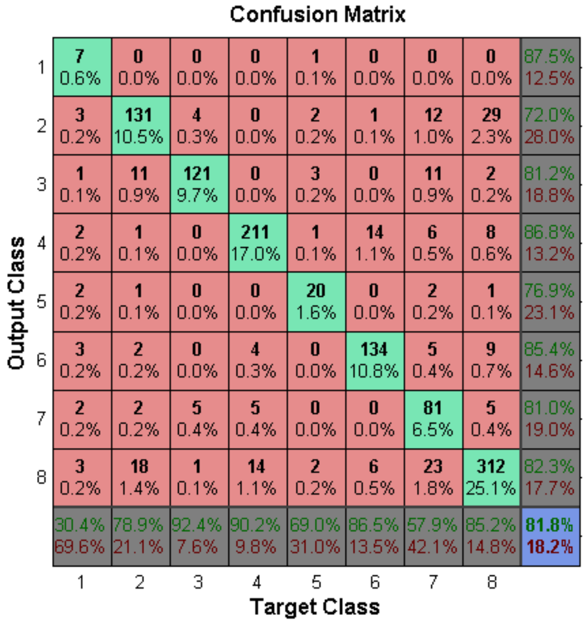


Fig. 13. Early Stopping plot from one of the training sessions of the base experiment.

VI. Discussion

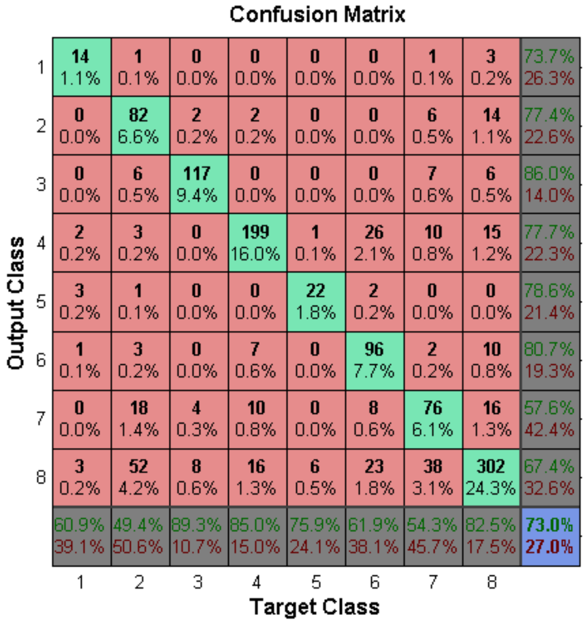
The result of the base experiment, as presented in the confusion matrix in fig. 12, had an overall accuracy of 93.5%. Considering that this was a simple experiment based only on partitioning and normalizing the frequency spectrum, the high accuracy was surprising.

Inspecting the confusion matrix in fig. 12, the most difficult instrument to predict was English horn, being



	1	2	3	4	5	6	7	8	
1	7 0.6%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	87.5% 12.5%
2	3 0.2%	131 10.5%	4 0.3%	0 0.0%	2 0.2%	1 0.1%	12 1.0%	29 2.3%	72.0% 28.0%
3	1 0.1%	11 0.9%	121 9.7%	0 0.0%	3 0.2%	0 0.0%	11 0.9%	2 0.2%	81.2% 18.8%
4	2 0.2%	1 0.1%	0 0.0%	211 17.0%	1 0.1%	14 1.1%	6 0.5%	8 0.6%	86.8% 13.2%
5	2 0.2%	1 0.1%	0 0.0%	0 0.0%	20 1.6%	0 0.0%	2 0.2%	1 0.1%	76.9% 23.1%
6	3 0.2%	2 0.2%	0 0.0%	4 0.3%	0 0.0%	134 10.8%	5 0.4%	9 0.7%	85.4% 14.6%
7	2 0.2%	2 0.2%	5 0.4%	5 0.4%	0 0.0%	0 0.0%	81 6.5%	5 0.4%	81.0% 19.0%
8	3 0.2%	18 1.4%	1 0.1%	14 1.1%	2 0.2%	6 0.5%	23 1.8%	312 25.1%	82.3% 17.7%
	30.4% 69.6%	78.9% 21.1%	92.4% 7.6%	90.2% 9.8%	69.0% 31.0%	86.5% 13.5%	57.9% 42.1%	85.2% 14.8%	81.8% 18.2%

Fig. 14. The confusion matrix from one of the training sessions using only the attack to train the network.



	1	2	3	4	5	6	7	8	
1	14 1.1%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.1%	3 0.2%	73.7% 26.3%
2	0 0.0%	82 6.6%	2 0.2%	2 0.2%	0 0.0%	0 0.0%	6 0.5%	14 1.1%	77.4% 22.6%
3	0 0.0%	6 0.5%	117 9.4%	0 0.0%	0 0.0%	0 0.0%	7 0.6%	6 0.5%	86.0% 14.0%
4	2 0.2%	3 0.2%	0 0.0%	199 16.0%	1 0.1%	26 2.1%	10 0.8%	15 1.2%	77.7% 22.3%
5	3 0.2%	1 0.1%	0 0.0%	0 0.0%	22 1.8%	2 0.2%	0 0.0%	0 0.0%	78.6% 21.4%
6	1 0.1%	3 0.2%	0 0.0%	7 0.6%	0 0.0%	96 7.7%	2 0.2%	10 0.8%	80.7% 19.3%
7	0 0.0%	18 1.4%	4 0.3%	10 0.8%	0 0.0%	8 0.6%	76 6.1%	16 1.3%	57.6% 42.4%
8	3 0.2%	52 4.2%	8 0.6%	16 1.3%	6 0.5%	23 1.8%	38 3.1%	302 24.3%	67.4% 32.6%
	60.9% 39.1%	49.4% 50.6%	89.3% 10.7%	85.0% 15.0%	75.9% 24.1%	61.9% 38.1%	54.3% 45.7%	82.5% 17.5%	73.0% 27.0%

Fig. 15. The confusion matrix from one of the training sessions excluding attack to train the network.

incorrectly predicted as an Oboe. This may not come as a surprise, since these who instruments are both of the woodwind family and hard to distinguish even for most non-musicians. Shown in fig. 18 is the scatter plot of the dataset, as shown in fig. 3, but limited to only these two instruments. The features have a major overlap, with no definite distinction between the two classes. This shows that the classifying network had a hard time utilizing the spectrum information for the feature vector defined in this experiment.

A. The Impact of the Attack

Comparing the two experiments using either only the attack or totally excluding it as feature vectors confirmed the hypothesis of the importance of the attack, as discussed in section II-B. Inspecting the results, the difference in accuracy was 80.2% compared to only 73.2%.

As seen in the confusion matrices of the experiments in fig. 14 and fig. 15, there are several instruments with a clear classification advantage for the attack feature vector.

The biggest difference was the classification of trumpet, having an accuracy of 81% with the attack and only 57.6% without. Comparing the spectrum between the two experiments, the frequency amplitudes in each case have distinct features, as shown in fig. 19. Besides the two clear frequency tops present in both experiments, having no attack resulted in many more frequencies with lower amplitudes in between these tops. One reason for the low accuracy achieved in this case could be due to all this noise, resulting in less distinction of the important frequencies.

B. The Impact of the Initial Frequency Spectrum

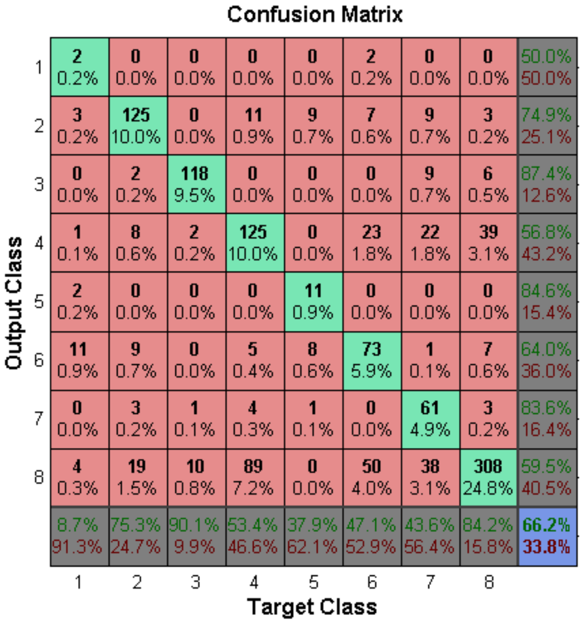
One conclusion the accuracy of training the network with only the initial 100 Hz of the spectrum says is that it seems to hold valuable information. It is represented in a feature vector with a magnitude of a five times less information than the feature vector used in the base experiment, but drops only 29.3% percentage points (to a 64.2% accuracy).

Whether 63.2% accuracy is good can be considered debatable, but there is still some information held in to the lower frequencies that opens up possibilities to further study in future work (see section VII-B).

C. Possible Introduction of Bias

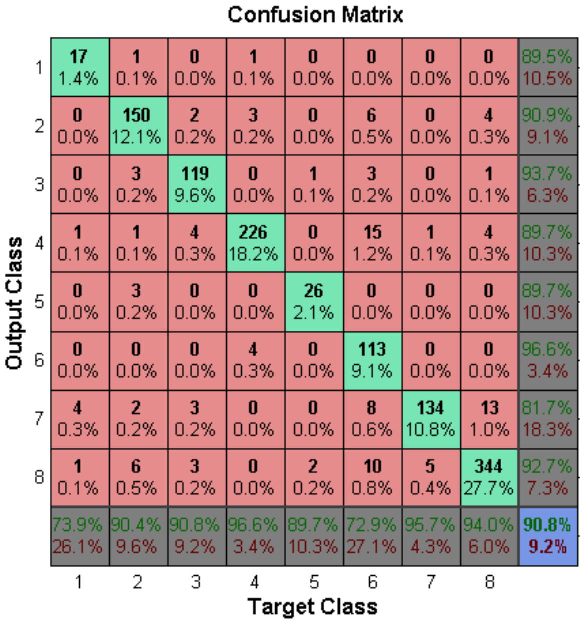
In this experiment, since raw data was both modified and limited in order to construct the feature vector, there is a risk that some bias was introduced that could have impacted the results.

For example, only using the initial 1000 Hz of the frequency spectrum may cause loss of information re-


Confusion Matrix
Output Class
Target Class

Output Class \ Target Class	1	2	3	4	5	6	7	8	
1	2 0.2%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 0.2%	0 0.0%	0 0.0%	50.0% 50.0%
2	3 0.2%	125 10.0%	0 0.0%	11 0.9%	9 0.7%	7 0.6%	9 0.7%	3 0.2%	74.9% 25.1%
3	0 0.0%	2 0.2%	118 9.5%	0 0.0%	0 0.0%	0 0.0%	9 0.7%	6 0.5%	87.4% 12.6%
4	1 0.1%	8 0.6%	2 0.2%	125 10.0%	0 0.0%	23 1.8%	22 1.8%	39 3.1%	56.8% 43.2%
5	2 0.2%	0 0.0%	0 0.0%	0 0.0%	11 0.9%	0 0.0%	0 0.0%	0 0.0%	84.6% 15.4%
6	11 0.9%	9 0.7%	0 0.0%	5 0.4%	8 0.6%	73 5.9%	1 0.1%	7 0.6%	64.0% 36.0%
7	0 0.0%	3 0.2%	1 0.1%	4 0.3%	1 0.1%	0 0.0%	61 4.9%	3 0.2%	83.6% 16.4%
8	4 0.3%	19 1.5%	10 0.8%	89 7.2%	0 0.0%	50 4.0%	38 3.1%	308 24.8%	59.5% 40.5%
	8.7% 91.3%	75.3% 24.7%	90.1% 9.9%	53.4% 46.6%	37.9% 62.1%	47.1% 52.9%	43.6% 56.4%	84.2% 15.8%	66.2% 33.8%

Fig. 16. The confusion matrix from one of the training sessions using the initial 100 Hz to train the network.


Confusion Matrix
Output Class
Target Class

Output Class \ Target Class	1	2	3	4	5	6	7	8	
1	17 1.4%	1 0.1%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	89.5% 10.5%
2	0 0.0%	150 12.1%	2 0.2%	3 0.2%	0 0.0%	6 0.5%	0 0.0%	4 0.3%	90.9% 9.1%
3	0 0.0%	3 0.2%	119 9.6%	0 0.0%	1 0.1%	3 0.2%	0 0.0%	1 0.1%	93.7% 6.3%
4	1 0.1%	1 0.1%	4 0.3%	226 18.2%	0 0.0%	15 1.2%	1 0.1%	4 0.3%	89.7% 10.3%
5	0 0.0%	3 0.2%	0 0.0%	0 0.0%	26 2.1%	0 0.0%	0 0.0%	0 0.0%	89.7% 10.3%
6	0 0.0%	0 0.0%	0 0.0%	4 0.3%	0 0.0%	113 9.1%	0 0.0%	0 0.0%	96.6% 3.4%
7	4 0.3%	2 0.2%	3 0.2%	0 0.0%	0 0.0%	8 0.6%	134 10.8%	13 1.0%	81.7% 18.3%
8	1 0.1%	6 0.5%	3 0.2%	0 0.0%	2 0.2%	10 0.8%	5 0.4%	344 27.7%	92.7% 7.3%
	73.9% 26.1%	90.4% 9.6%	90.8% 9.2%	96.6% 3.4%	89.7% 10.3%	72.9% 27.1%	95.7% 4.3%	94.0% 6.0%	90.8% 9.2%

Fig. 17. The confusion matrix from one of the training sessions using following using 900 Hz to train the network.

garding overtones or harmonics that potentially could improve the results in either of the experiments.

Also, since the frequency spectrum is normalized, all information regarding the actual amplitudes is lost in trade for how the amplitudes of the frequencies for a single sample co-relates. There may be some information in the amplitudes that would further improve the results of either of the experiments.

VII. Future Work

The results of the performed experiments bring us some conclusions that could be further studied.

A. *Further Studying the Attack*

The current experiments using the attack are all very simple in their nature, by only separating the attack and then creating the normalized feature vector of the frequency spectrum. An interesting future work would be to use the characteristics of the attack itself and from that create specialized features. The shape of the attack is an important factor and using features such as rise and decay time, as well as the derivative of them both, would possibly help classify instruments.

Other features of the attack include the RMS energy, as well as the rise time from the point of onset to the point where the RMS energy is maximized. Having the feature vector contain several of these characteristics, or alternatively using them together with the currently used normalized spectrum, would lead to many interesting experiments and possible extensions.

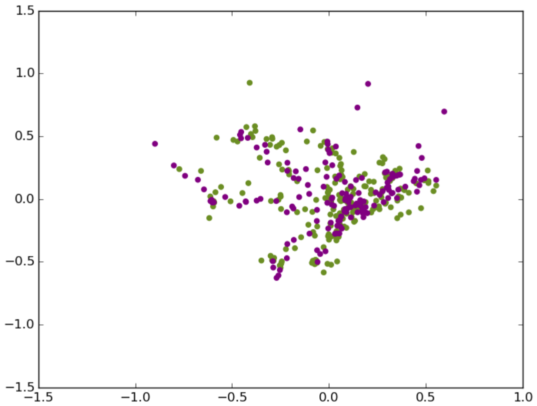
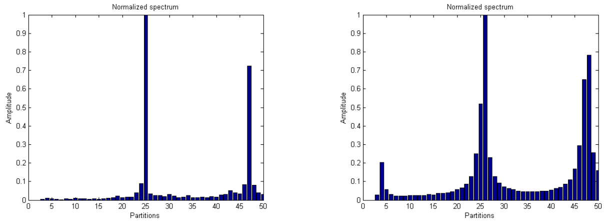


Fig. 18. Scatter plot of the Principal Component Analysis limited to only English horn in green and Oboe in purple shows the features almost totally overlapping.

B. *Further studying the initial 100 Hz*

As discussed in section VI, the initial 100 Hz seems to hold some key information that can be used in musical instrument recognition. Though it did not outperform the following 900 Hz, one could argue that there is



Only attack *Without attack*

Fig. 19. Comparison plots of feature vectors with or without attack in a sample file of trumpet.

still an interesting aspect of how well it performed considering that its accuracy dropped 29.3% percentage points when the information in the feature vector 90% smaller. Perhaps a different training method could be used, or the data pre-processed differently that would impact the accuracy.

C. Expanding the Studied Frequency Range

As pointed out in section VI-C, there may be some information of overtones or harmonies beyond 1000 Hz that was overlooked in this experiment. Further studies on this experiment could potentially explore if expanding the range of frequencies used to construct the feature vector possibly can improve the accuracy of the network. In some related work (see section VIII) show examples of using up to 1500 Hz to train their models.

D. Using the Mel-frequency Cepstrum

Mel-frequency cepstrum coefficients (MFCC) are commonly used in the application of speech recognition and are also applicable in classifying musical instruments. The essential part of MFCC is taking the combining the discrete cosine transform with the power spectrum of the Fourier Transform of a signal [13]. The MFCC has been used by some studies to test the effectiveness of using the cepstrum to classify instruments; so applying this alternative view of the spectrum to the experiments in this study would lead to an interesting comparison.

VIII. Related work

A. Human Recognition of Audio Signals

There have been a number of studies on the human ability to distinguish musical instruments from each other by investigating the auditory properties of the sound itself. McAdams and Bigand [14] identified three parts of the musical sound event (attack, middle sustain and final decay) and compared how they did, or did not, impact in humans identifying the instrument depending on external factors. For example, they studied the importance of the attack by noticing the large drop in performance when the attack was cut out, but also how this reduction minimized when vibrato was used. Therefore, drawing the conclusion that the most important information of an instrument exists in the first part of the sound event, but in absence of that information, additional information still existed in the sustain portion is augmented slightly when changes to the pattern that specify the resonance structure of the source was present, such as vibrato.

B. Musical Instrument Recognition Systems

There have been many studies performed on musical instrument recognition. When classifying instruments, there are many characteristics defining them that can be used. The studies described in this section used a range of different features and they can all be divided into the two categories spectral (frequency) and temporal (time).

Early studies used statistical pattern-recognition techniques. An early example is the study performed by Bourne [15] 1972, which used spectral data of the sound as input to a Bayesian classifier to distinguish between three instruments. The same methods have been applied later as well; Fujinaga [16] used properties of the steady-state of the sound together with a k-nearest neighbor [17] (kNN) classifier, achieving an accuracy of 50% using 23 instruments.

In a study performed by Brown [18], methods from the well-researched field of Speaker Identification were used to determine the most effective features to be able to separate four different instruments. Using features such as Cepstral coefficients, histogram differentials and cross correlation, an accuracy of around 80% was obtained.

Eronen [19] used a wide set of features, covering both spectral and temporal properties of the musical instruments, was used together with a Gaussian classifier in combination with kNN). Out of a total of 30 instruments, the instrument family was successfully classified with 94% accuracy, whereas individual instruments had an accuracy of 80%.

Kaminskyj [20] used the RMS energy envelope of the temporal spectrum as features to compare the performance of ANNs with a simple kNN classifier, setting k to 1, to classify four instruments. Using kNN performed best, achieving a 100% accuracy, compared to around 96% in the case of ANN, which is motivated by kNN being able to take more info from the RMS into account than the 32 weights used in ANN.

IX. Conclusion

The approach of this paper was to evaluate the features in a system performing musical instrument recognition. The five different experiments that were performed all used a feature vector of length 50, containing a normalized frequency spectrum of the audio signal.

Training an ANN to perform musical instrument recognition was highly successful, and using a feature vector based on the first 1000 Hz of the frequency spectrum reached an average accuracy of 93.5%. Also, studying the impact of certain smaller aspects in the sound, such as a more limited frequency range or a limited portion of the audio sample, seemed to prove that these aspects held some key information that still enabled a fairly good distinction between different instruments with an accuracy at best being 80.2% and worst 64.2%.

This study lays the foundation for interesting future work, further examining these aspects and experimenting with different representations of them to optimize accuracy.

References

[1] C. Sachs, *The history of musical instruments*. Courier Corporation, 2012. 1

[2] S.-C. Wang, "Artificial neural network," in *Interdisciplinary Computing in Java Programming*. Springer, 2003, pp. 81–100. 1

[3] S. A. Broughton and K. M. Bryan, *Discrete Fourier analysis and wavelets: applications to signal and image processing*. John Wiley & Sons, 2011. 1

[4] M. Clark Jr, D. Luce, R. Abrams, H. Schlossberg, and J. Rome, "Preliminary experiments on the aural significance of parts of tones of orchestral instruments and on choral tones," *Journal of the Audio Engineering Society*, vol. 11, no. 1, pp. 45–54, 1963. 1

[5] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler, "A tutorial on onset detection in music signals," *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 5, pp. 1035–1047, 2005. 2

[6] M. T. U. B. H. Suits, Physics Department. (1998) Physics of music - notes. [Online]. Available: http://www.phy.mtu.edu/~suits/Physicsofmusic.html 2

[7] L. P. Orchestra. (2016, apr) Sound samples. [Online]. Available: http://www.philharmonia.co.uk/explore/make_music 2

[8] E. W. Weisstein, "Fast fourier transform," 2015. 3

[9] P. Iverson and C. L. Krumhansl, "Isolating the dynamic attributes of musical timbrea)," *The Journal of the Acoustical Society of America*, vol. 94, no. 5, pp. 2595–2603, 1993. 4

[10] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Prentice Hall PTR, 1998. 5

[11] L. Prechelt, "Automatic early stopping using cross validation: quantifying the criteria," *Neural Networks*, vol. 11, no. 4, pp. 761–767, 1998. 5

[12] M. Riedmiller, *Rprop-Description and Implementation Details: Technical Report*. Inst. f. Logik, Komplexität u. Deduktionssysteme, 1994. 5

[13] J. C. Brown, "Computer identification of musical instruments using pattern recognition with cepstral coefficients as features," *The Journal of the Acoustical Society of America*, vol. 105, no. 3, pp. 1933–1941, 1999. 9

[14] S. E. McAdams and E. E. Bigand, "Thinking in sound: The cognitive psychology of human audition." in *Based on the fourth workshop in the Tutorial Workshop series organized by the Hearing Group of the French Acoustical Society.* Clarendon Press/Oxford University Press, 1993. 9

[15] J. B. Bourne, "Musical timbre recognition based on a model of the auditory system," Ph.D. dissertation, 1972. 9

[16] I. Fujinaga, "Machine recognition of timbre using steady-state tone of acoustic musical instruments," in *Proceedings of the International Computer Music Conference*. Citeseer, 1998, pp. 207–10. 9

[17] P. Cunningham and S. J. Delany, "k-nearest neighbour classifiers," *Multiple Classifier Systems*, pp. 1–17, 2007. 9

[18] J. C. Brown, O. Houix, and S. McAdams, "Feature dependence in the automatic identification of musical woodwind instruments," *The Journal of the Acoustical Society of America*, vol. 109, no. 3, pp. 1064–1072, 2001. 9

[19] A. Eronen and A. Klapuri, "Musical instrument recognition using cepstral coefficients and temporal features," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, vol. 2. IEEE, 2000, pp. II753–II756. 9

[20] I. Kaminskyj and A. Materka, "Automatic source identification of monophonic musical instrument sounds," in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 1. IEEE, 1995, pp. 189–194. 9