MUSICAL INSTRUMENT IDENTIFICATION

Shubham, 150102079, ECE

Abstract

In this project, we present a system for the classification of musical instruments belonging to different instrument families exploiting the various temporal and spectral features that characterize the "timbre" of a particular instrument. Many features covering both spectral and temporal properties and their influences on a musical sound were investigated, and their extraction algorithms were designed. The features were extracted from the dataset that consisted of 558 samples covering the full pitch ranges of 10 orchestral instruments from the string, brass and woodwind families. The classification results consolidated the dependence of an instrument's timbral uniqueness on the features, especially the spectral ones. The correct instrument family was recognized with ~96 % accuracy and individual instruments within the families with over 95 % accuracy for each of the three families considered. Also, a hierarchical classification framework is utilized keeping the mind the taxonomic nature of musical sounds.

Link to Codes Used

1. Introduction

1.1 Introduction to Problem

Musical Instrument Identification by a computer has many scientific as well as practical applications such as automatic annotation of musical data, structured coding and ultimately developing a system that can understand music enough to collaborate with a human in real-time performance. The goal of this project is to identify musical instruments from their audio samples using a statistical pattern-recognition technique.

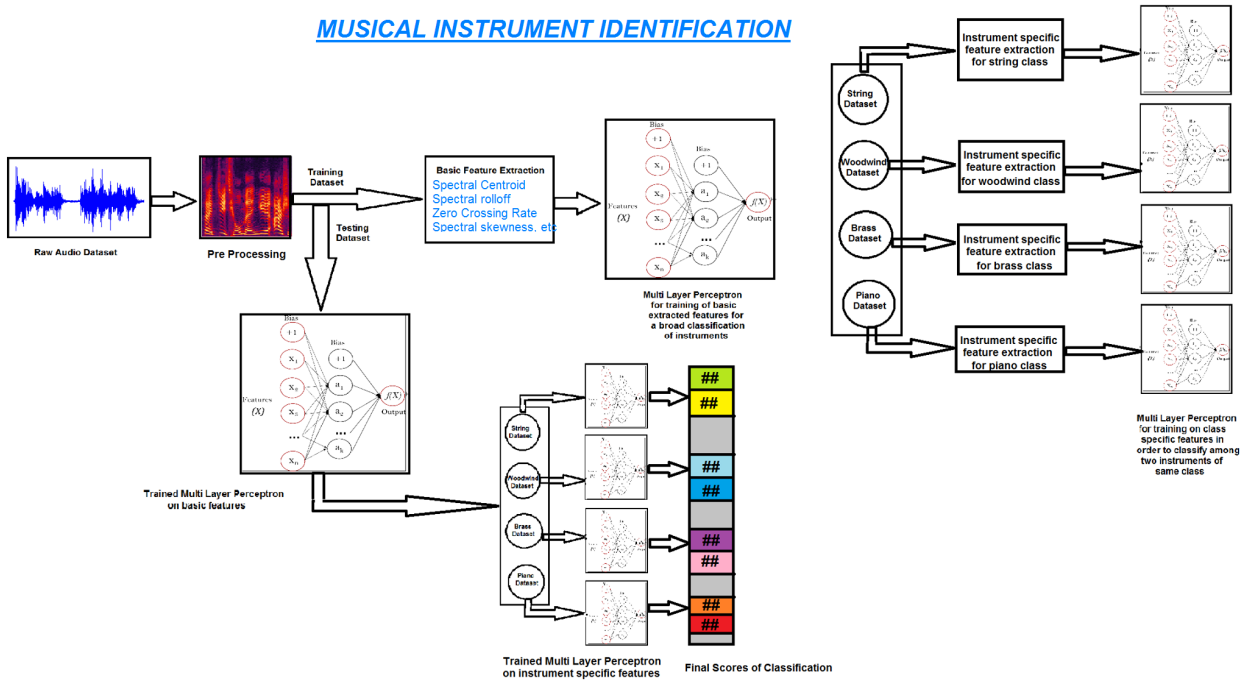
1.2 Motivation

There are many scientific and practical applications in which musical instrument identification by a computer would be useful. Some of them are:

- Automatically annotating musical multimedia data
- Transcribing musical performances for purposes of teaching, theoretical study, or structured coding
- Ultimately, developing a system that can understand music enough to collaborate with a human in real-time performance

By attempting to build such systems, we stand to learn a great deal about the human system we seek to emulate.

1.3 Figure - Block diagram of the Musical Instrument Identification approach



1.4 Literature Review

[1] Giulio Agostini,Maurizio Longari and Emanuele Pollastri,"Musical Instrument Timbres Classification with Spectral Features",*EURASIP Journal on Applied Signal Processing 2003.*

This paper addresses the problem of musical instrument classification from audio sources. A detailed analysis of spectral features and their precise description is presented along with their relative salience in the classification process. A number of classifying methods such as Discriminant Analysis techniques, Support Vector Machines and k-nearest neighbours have been tested and compared.

[2] Keith D. Martin and Youngmoo E. Kim, "Musical instrument identification: A pattern-recognition approach",*Presented at the 136 th meeting of the Acoustical Society of America, October 13, 1998*, MIT Media Lab Machine Listening Group Rm. E15-401, 20 Ames St., Cambridge, MA 02139.

This paper emphatically demonstrates that the acoustic properties studied in the literature as components of musical timbre are indeed useful features for musical instrument recognition. Starting from basic features, it follows a correlogram-based approach in classifying the instruments. Prominent features have also been discussed which influenced the classification process among the instruments belonging to a single family.

3] Antti Eronen and Anssi Klapuri, "Musical Instrument Recognition using Cepstral Coefficients and temporal features",*Signal Processing Laboratory* , Tampere University of Technology P.O.Box 553, FIN-33101 Tampere, FINLAND

In this paper, a system for pitch-independent musical instrument recognition is presented. A wide set of features covering both spectral and temporal properties of sounds was investigated, and their extraction algorithms were designed. Very high accuracy was attained with a certain set of features. Also, the utilization of a hierarchical classification framework is considered.

[4] Antti Eronen,"Comparison of features for Musical Instrument Recognition",*Signal Processing Laboratory, Tampere University of Technology P.O.Box 553, FIN-33 101 Tampere, Finland .*

The paper intends to compare different features with regard to recognition performance in a musical instrument recognition system. The performance of earlier described features relating to the temporal development, modulation properties, brightness, and spectral synchronicity of sounds is also analysed. The errors made by the recognition system are compared with the results reported by a human perception experiment.

1.5 Proposed Approach

Why do different musical instruments have different sounds?
Various characteristics of sound, such as loudness (related to energy) and pitch (related to frequency) determine how our brain perceives a musical instrument. But, if a clarinet and a piano play notes of the same pitch and loudness, the sounds will still be quite distinct to our ears. What, then, discriminates the sound of a clarinet from a violin?

The answer is Timbre! Timbre distinguishes different types of musical instruments, such as string instruments, wind instruments, and percussion instruments. It also enables listeners to distinguish different instruments in the same category (e.g. a clarinet and an oboe).
Musical instruments do not vibrate at a single frequency: a given note involves vibrations at many different frequencies, often called harmonics, partials, or overtones. The relative pitch and loudness of these overtones along with other acoustic features give the note a characteristic sound we call the timbre of the instrument.

Timbre can be successfully used to distinguish between the 4 broad classes of musical instruments. After that, we may use class-specific features to distinguish the instrument within the class.

Thus to achieve the goal of detecting the instrument being played in a given sound clip, we propose a two-step approach:

[1] In the first step we take advantage of the fact that musical instruments can be divided into 3 broad classes: Woodwind; String and Brass. Thus, we use a simple multilayer perceptron trained using basic timbre features: Spectral Centroid, Zero Crossing rate, Spectral Rolloff and Spectral Skewness among others. This helps to find the basic class to which the given musical instrument belongs and leaves us with the task of finding the exact instrument from within this basic class.

[2] In the second step, we train a classifier with individual class-specific features. This helps us to accurately identify the musical instrument within the basic class as detected earlier.

1.6 Report Organization

This project report is organized as follows.

1. First, we give some background information on the importance of automatic Musical Instrument recognition and the motivation behind it.
2. This is followed by the section on the proposed approach where we discuss the approach towards the classification task. Some details about feature properties and their mathematical constructs are also presented.
3. Then, a brief description of the dataset is followed by classification techniques employed in the project.
4. Finally, the results are presented along with the conclusion of the project. A short summary and possible future extensions close the report.

2. Proposed Approach

The approach towards the identification of musical instruments relies on the set of features, collectively known as "Timbre". There are two broad types of features of a music signal specifying its properties:

- The temporal features (time domain features), which are simple to extract and have easy physical interpretation, like the energy of the signal, zero crossing rate, maximum amplitude, minimum energy, etc.
- The spectral features (frequency-based features), which are obtained by converting the time-based signal into the frequency domain using the Fourier Transform, like fundamental frequency, frequency components, spectral centroid, etc.

Timbre depends primarily upon the spectral features, although it also depends upon the sound pressure and the temporal characteristics of the sound. Some of the prominent features are described below :

[1] **Amplitude Envelope:-** It refers to the changes in the amplitude of a sound over time and is an influential property as it affects our perception of timbre.

[2] **Zero-crossing rate:-** The zero-crossing rate is the rate of sign changes along a signal, i.e., the rate at which the signal changes from positive to negative or back. This is a good measure of the pitch as well as the noisiness of a signal. This feature has been used heavily in both speech recognition and music information retrieval, being a key feature to classify percussive sounds.

$$zcr = \frac{1}{T-1}\sum_{t=1}^{T-1} 1_{\mathbb{R}_{<0}}(s_t s_{t-1})$$

where **s** is a signal of length T and **1** is an indicator function.

.

[3] **Pitch** is a perceptual property of sounds that allows their ordering on a frequency-related scale. Pitch may be quantified by frequency; "high" pitch means very rapid oscillation, and "low" pitch corresponds to slower oscillation.

[4] **Spectral Envelope:-** Spectral envelope is a curve in the frequency-amplitude plane, derived from a Fourier magnitude spectrum. It wraps tightly and smoothly around the magnitude spectrum, linking the peaks.

[5] **Spectral Centroid:-** Spectral Centroid is simply the centroid of the spectral envelope, and is one of the most important attributes governing timbre.

$$\text{Centroid} = \frac{\sum_{f=f_{\min}}^{f_{\max}} f \cdot E(f)}{\sum_{f=f_{\min}}^{f_{\max}} E(f)},$$

[6] **Spectral Rolloff Frequency:-** This is a measure of the amount of the right-skewness of the Energy spectrum. The spectral roll off point is the frequency in the spectrum below which 85% of the total energy is contained. This ratio is fixed by default to 85 %. (ref. *Tzanetakis and Cook, 2002*)

[7] **Intensity:-** The sum of the energy in the spectral envelope approximates the instantaneous loudness of the signal. Tracking this over time leads to simple measures of amplitude modulation, which can reveal tremolo (an important feature for brass instruments). Intensity is the basis of many other spectral features like **Spectral Rolloff frequency** among others.

[8] **Spectral Crest:-**It is defined as the ratio of peak and RMS value of the spectrum. It is usually expressed in dB, thus it's alternatively defined as the level difference between the peak and the RMS value of the waveform. Most ambient noise has a crest factor of around 10 dB while impulsive sounds such as gunshots can have crest factors of over 30 dB.

$$C = \frac{|x_{\text{peak}}|}{x_{\text{rms}}}$$

$$C_{\text{dB}} = 20 \log_{10} \frac{|x_{\text{peak}}|}{x_{\text{rms}}}.$$

[9] **Spectral flatness or tonality coefficient:-**(also popularly known as Wiener entropy) is a measure used in digital signal processing to quantify how noise-like a sound is, as opposed to being tone-like. The meaning of tonal in this context is in the sense of the amount of peaks or resonant structures in a power spectrum, as opposed to a flat spectrum of black noise. A high spectral flatness (approaching 1.0 for black noise) indicates that the spectrum has a similar amount of power in all spectral bands — this would sound similar to black noise, and the graph of the spectrum would appear relatively flat and smooth. A low spectral flatness (approaching 0.0 for a pure tone) indicates that the spectral power is concentrated in a relatively small number of bands — this would typically sound like a mixture of sine waves, and the spectrum would appear "spiky".

$$\text{Flatness} = \frac{\sqrt[N]{\prod_{n=0}^{N-1} x(n)}}{\frac{\sum_{n=0}^{N-1} x(n)}{N}} = \frac{\exp\left(\frac{1}{N} \sum_{n=0}^{N-1} \ln x(n)\right)}{\frac{1}{N} \sum_{n=0}^{N-1} x(n)}$$

[10] **The skewness of a spectrum:-** The skewness of a spectrum is the third central moment of this spectrum, divided by the 1.5 power of the second central moment.

[11] **Spectral Slope:-** The spectral slope is — similar to the spectral decrease — a measure of the slope of the spectral shape. It is calculated using a linear approximation of the magnitude spectrum more specifically, a linear regression approach is used. In the presented form, the linear function is modelled from the magnitude spectrum.

$$v_{\mathrm{SSl}}(n) = \frac{\sum_{k=0}^{\kappa/2-1}(k-\mu_k)(|X(k,n)|-\mu_{|X|})}{\sum_{k=0}^{\kappa/2-1}(k-\mu_k)^2}$$

$$= \frac{\mathcal{K}\sum_{k=0}^{\kappa/2-1} k\cdot|X(k,n)| - \sum_{k=0}^{\kappa/2-1} k\cdot\sum_{k=0}^{\kappa/2-1}|X(k,n)|}{\mathcal{K}\cdot\sum_{k=0}^{\kappa/2-1}k^2-\left(\sum_{k=0}^{\kappa/2-1}k\right)^2}.$$

[12] **Spectral leakage/Decrease:-** The spectral decrease estimates the steepness of the decrease of the spectral envelope over frequency. The result of the spectral decrease is a value vSD(n) <= 1. Low results indicate the concentration of the spectral energy at bin 0.

$$v_{\mathrm{SD}}(n) = \frac{\sum_{k=1}^{\kappa/2-1}\frac{1}{k}\cdot\left(|X(k,n)|-|X(0,n)|\right)}{\sum_{k=1}^{\kappa/2-1}|X(k,n)|}.$$

[13] **Variance of Spectral Centroid:-** Its the variance of the spectral centroid over our signal. This is a useful feature as it tells the very nature of our spectral spread over the frequency range.

[14] **The mel-frequency cepstrum (MFC) :-** is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency. The difference between the normal cepstrum and the mel-frequency cepstrum is that in the MFC, the frequency bands are equally spaced on the mel scale, which approximates the human auditory system's response more closely than the linearly-spaced frequency bands used in the normal cepstrum. This frequency warping can allow for better representation of sound

Step by Step Explanation of MFCC Feature Extraction:-

1. Frame the signal into short frames.
2. For each frame calculate the periodogram estimate of the power spectrum.
3. Apply the mel filterbank to the power spectra, and sum the energy in each filter.
4. Take the logarithm of all filterbank energies.
5. Take the DCT of the log filterbank energies.
6. Keep DCT coefficients 2-13, and discard the rest.

Intuitive Understanding of Each step in MFCC feature extraction:-

An audio signal is constantly changing, so to simplify things we assume that on short time scales the audio signal doesn't change much (when we say it doesn't change, we mean statistically i.e. statistically stationary, obviously the samples are constantly changing on even short time scales). This is why we frame the signal into 20-40ms frames. If the frame is much shorter we don't have enough samples to get a reliable spectral estimate, if it is longer the signal changes too much throughout the frame. The next step is to calculate the power spectrum of each frame. This is motivated by the human cochlea (an organ in the ear) which vibrates at different spots depending on the frequency of the incoming sounds. Depending on the location in the cochlea that vibrates (which wobbles small hairs), different nerves fire informing the brain that certain frequencies are present. Our periodogram estimate performs a similar job for us, identifying which frequencies are present in the frame. The periodogram spectral estimate still contains a lot of information not required for Musical Instrument Recognition. In particular, the cochlea can not discern the difference between two closely spaced frequencies. This effect becomes more pronounced as the frequencies increase. For this reason, we take clumps of periodogram bins and sum them up to get an idea of how much energy exists in various frequency regions. This is performed by our Mel filterbank: the first filter is very narrow and gives an indication of how much energy exists near 0 Hertz. As the frequencies get higher our filters get wider as we become less concerned about variations. We are only interested in roughly how much energy occurs at each spot. The Mel scale tells us exactly how to space our filterbanks and how wide to make them. Once we have the filterbank energies, we take the logarithm of them. This is also motivated by human hearing: we don't hear loudness on a linear scale. Generally to double the perceived volume of a sound we need to put 8 times as much energy into it. This means that large variations in energy may not sound all that different if the sound is loud, to begin with. This compression operation makes our features match more closely what humans actually hear. Why the logarithm and not a cube root? The logarithm allows us to use cepstral mean subtraction, which is a channel normalisation technique. The final step is to compute the DCT of the log filterbank energies. There are 2 main reasons this is performed. Because our filterbanks are all overlapping, the filterbank energies are quite correlated with each other. The DCT decorrelates the energies which mean diagonal covariance matrices can be used to model the features in e.g. the HMM classifier. But notice that only 12 of the 26 DCT coefficients are kept. This is because the higher DCT coefficients represent fast changes in the filterbank energies and it turns out that these fast changes actually degrade the recognition performance, so we get a small improvement by dropping them.

3. Experiments & Results

3.1 Dataset Description

The dataset chosen for the task at hand is the audio samples taken from the University of Iowa - Electronic Music Studios. The samples taken belong to the three instrument families namely - Brass, String and Woodwind.

Instruments used were:-

STRING:
1)Viola - 100 samples
2)Violin - 90 samples

BRASS:
1)Trumpet - 71 samples
2)TenorTrombone - 33 samples
3)Tuba - 37 samples
4)Horn - 44 samples

WOODWIND:
1)Saxophone - 32 samples
2)EbClarinet - 39 samples
3)Oboe - 35 samples
4)Flute - 77 samples

There are a total of 558 samples of which an 80 % - 20 % division has been done to be used for training and testing, respectively.

The link to the dataset is here: Audio Samples
Dataset Original link: University of Iowa: Audio Samples Site

3.2 Discussion

Individual Training:-

Basic family classifier

Training - 446
Testing - 112

```
Epoch 246/250
446/446 [==============================] - 0s - loss: 0.0807 - acc: 0.9686
Epoch 247/250
446/446 [==============================] - 0s - loss: 0.0934 - acc: 0.9552
Epoch 248/250
446/446 [==============================] - 0s - loss: 0.1078 - acc: 0.9574
Epoch 249/250
446/446 [==============================] - 0s - loss: 0.0897 - acc: 0.9709
Epoch 250/250
446/446 [==============================] - 0s - loss: 0.0677 - acc: 0.9776
Test loss:0.0680611383702
Test accuracy:0.964285714286
```

Training accuracy = 97.76%
Testing accuracy = 96.42%

Brass classifier

Training - 148
Testing - 37

```
Epoch 94/100
148/148 [==============================] - 0s - loss: 0.0034 - acc: 1.0000 - val_loss: 2.3036e-07 - val_acc: 1.0000
Epoch 95/100
148/148 [==============================] - 0s - loss: 9.2180e-05 - acc: 1.0000 - val_loss: 2.2875e-07 - val_acc: 1.0000
Epoch 96/100
148/148 [==============================] - 0s - loss: 3.3736e-04 - acc: 1.0000 - val_loss: 2.2875e-07 - val_acc: 1.0000
Epoch 97/100
148/148 [==============================] - 0s - loss: 0.0097 - acc: 0.9932 - val_loss: 2.2714e-07 - val_acc: 1.0000
Epoch 98/100
148/148 [==============================] - 0s - loss: 1.3896e-04 - acc: 1.0000 - val_loss: 2.2392e-07 - val_acc: 1.0000
Epoch 99/100
148/148 [==============================] - 0s - loss: 0.0143 - acc: 0.9932 - val_loss: 2.1103e-07 - val_acc: 1.0000
Epoch 100/100
148/148 [==============================] - 0s - loss: 4.0681e-04 - acc: 1.0000 - val_loss: 2.0137e-07 - val_acc: 1.0000
Test loss:2.01367155695e-07
Test accuracy:1.0
```

Training accuracy = 100%
Testing accuracy = 100%

String classifier

Training - 152
Testing - 38

```
Epoch 95/100
152/152 [==============================] - 0s - loss: 0.0922 - acc: 0.9605
Epoch 96/100
152/152 [==============================] - 0s - loss: 0.0815 - acc: 0.9737
Epoch 97/100
152/152 [==============================] - 0s - loss: 0.1185 - acc: 0.9539
Epoch 98/100
152/152 [==============================] - 0s - loss: 0.1538 - acc: 0.9474
Epoch 99/100
152/152 [==============================] - 0s - loss: 0.1087 - acc: 0.9671
Epoch 100/100
152/152 [==============================] - 0s - loss: 0.1083 - acc: 0.9671
Test loss:0.00981693783481
Test accuracy:1.0
```

Training accuracy = 96.7%
Testing accuracy = 100%

Woodwind classifier

Training - 146
Testing - 37

```
Epoch 195/200
146/146 [==============================] - 0s - loss: 0.0090 - acc: 1.0000
Epoch 196/200
146/146 [==============================] - 0s - loss: 0.0046 - acc: 1.0000
Epoch 197/200
146/146 [==============================] - 0s - loss: 0.0207 - acc: 0.9863
Epoch 198/200
146/146 [==============================] - 0s - loss: 0.0142 - acc: 0.9863
Epoch 199/200
146/146 [==============================] - 0s - loss: 0.0103 - acc: 0.9932
Epoch 200/200
146/146 [==============================] - 0s - loss: 0.0109 - acc: 0.9932
Test loss:0.233520901686
Test accuracy:0.945945945946
```

Training accuracy = 99.3%
Testing accuracy = 94.59%

6 layer Neural Networks with different hyperparameters have been used for classifiers.

Features used in individual classifiers:-

Basic classifier and brass classifier:-

1. Spectral centroid
2. Zero crossing
3. Spectral roll off
4. Spectral Crest
5. Spectral Decrease
6. Spectral Flatness
7. Spectral Skewness
8. Spectral Slope

String classifier:-

1. Spectral Centroid
2. Fundamental frequency
3. Spectral centroid variance
4. Spectral Crest
5. Spectral Decrease
6. Spectral Flatness
7. Spectral Skewness
8. Spectral Slope

Woodwind classifier:-

1. Spectral Centroid
2. Spectral Crest
3. Spectral Decrease
4. Spectral Flatness
5. Spectral Skewness
6. Spectral Slope
7. Fundamental Frequency
8. Variance of Spectral Centroid
9. MFCC

4. Conclusions

4.1 Summary

As proposed in our approach, we were successfully able to build a two-stage musical instrument detector. The various temporal and spectral features were successfully extracted using efficient algorithms, implemented in MATLAB. We were able to achieve decent test accuracy (96.42% for the basic classifier, 100% for the brass family, 96.7% for the string family and 94.59% for the woodwind family). This shows that the features we had chosen for each of the classifiers based on the unique traits of each class were quite appropriate. We have also tried to ensure minimal use of inbuilt MATLAB functions and have written code for the features and support functions(like spectrogram, Radix 2 FFT, etc) to ourselves using the algorithms discussed in class (EE320- Digital Signal Processing).

4.2 Future Extensions

The area of Musical instrument Classification is vast and many dimensions are still unexplored. For a first extension, we would like to work on classifying Indian Classical musical instruments.