Musical Instrument Recognition

Collection Editors:
Kyle Ringgenberg
Yi-Chieh Wu
Patrick Kruse

Musical Instrument Recognition

Collection Editors:

Kyle Ringgenberg
Yi-Chieh Wu
Patrick Kruse

Authors:

Patrick Kruse
Michael Lawrence
Kyle Ringgenberg
Yi-Chieh Wu

Online:

< http://cnx.org/content/col10313/1.3/ >

C O N N E X I O N S

Rice University, Houston, Texas

Collection structure revised: December 14, 2005
PDF generated: February 4, 2011
For copyright and attribution information for the modules contained in this collection, see p. 23.

Table of Contents

1 Introduction and Background

2 Implementation and Results

3 Wrap-Up and The Team

Chapter 1

Introduction and Background

1.1 Introduction[1]

1.1.1 Introduction

This project aims to accurately detect the pitches and instruments of a signal. To accomplish this, we intend to record and analyze the entire range of a few instruments, and then use this analysis to decompose monophonic, or one instrument, and polyphonic, or multiple instrument, signals into their component instruments and pitches. To keep the program managable, we will limit the polyphonic signals to three instruments.

The applications of this project are far reaching, at least within the realm of musical instrument signal processing. A better understanding of musical timbre and tone is to be gained from the analysis of each instrument, and more specifically by comparing the signals produced by different instruments on the same pitch. Another, more practical goal of the musical instrument recognition project is automatic music transcription, or music to MIDI conversion.

1.2 Mathematics of Music Theory[2]

1.2.1 Simple Music Theory

For those of you unfamiliar with music, we offer a (very) brief introduction into the technical aspects of music.

The sounds you hear over the airwaves and in all manner of places may be grouped into 12 superficially disparate categories. Each category is labeled a "note" and given an alphasymbolic representation. That is, the letters A through G represent seven of the notes and the other five are represented by appending either a pound sign (#, or sharp) or something that looks remarkably similar to a lower-case b (also called a flat).

Although these notes were conjured in an age where the modern theory of waves and optics was not dreamt of even by the greatest of thinkers, they share some remarkable characteristics. Namely, every note that shares its name with another (notes occupying separate "octaves," with one sounding higher or lower than the other) has a frequency that is some rational multiple of the frequency of the notes with which it shares a name. More simply, an A in one octave has a frequency twice that of an A one octave below.

As it turns out, **every** note is related to every other note by a common multiplicative factor. To run the full gamut, one need only multiply a given note by the 12th root of two n times to find the nth note "above" it (i.e. going up in frequency). Mathematically:

(nth note above base frequency) = (base frequency)$2^{\frac{n}{12}}$

[1] This content is available online at <http://cnx.org/content/m13195/1.3/>.

[2] This content is available online at <http://cnx.org/content/m12461/1.5/>.

1.2.2 Harmonics

The "note" mentioned above is the pitch you most strongly hear. Interestingly, however, there **are** other notes extant in the signal your ear receives. Any non-electronic instrument actually produces many, many notes, all of which are overshadowed by the dominant tone. These extra notes are called **harmonics**. They are responsible for the various idiosyncracies of an instrument; they give each instrument its peculiar flavor. It is, effectively, with these that we identify the specific instrument playing.

1.2.3 Duration and Volume

We will also make a quick note (no pun intended) for the other two defining characteristics of a musical sound. **Duration** is fairly self-explanatory; notes last for a certain length of time. It is important to mention that in standard music practices most notes last for a length of time relative to the **tempo** of the music. The tempo is merely the rate as which the music is played. Thus, by arbitrarily defining a time span to be equal to one form of note duration we may derive other note durations from that.

More concretely: taking a unit of time, say one minute, and dividing it into intervals, we have **beats per minute**, or **bpm**. One beat corresponds, in common time, to a quarter note. This is one quarter of the longest commonly-used note, the whole note. The length of time is either halved or doubled to find the nearest note duration to the base duration (and so on from there). The "U.S. name" for the duration of the notes is based on their fraction of the longest note. Other, archaic, naming conventions include the English system replete with hemi demi semi quavers and crotchets (for more information, follow the supplemental link on the left of the page).

Volume, on the other hand, is based on the signal power and is not so easily quantifiable. The terms in music literature are always subjective (louder, softer) and volume-related styles from previous eras are heavily debated ("but certainly Mozart wanted it to be louder than **that**!"). For our project, we save the information representing the volume early on, then normalize it out of the computations to ease the comparisons.

1.3 Common Music Terms[3]

Upon discussion of the implementation and general work done during the course of this project, a number of specific musical terms will be used. For those with a bit of musical background, these should be very straightforward. However, considering the steriotypical divide between the arts and the sciences, the following is a psuedo-comprehensive list of common terms that will be referenced within this report.

- Articulation: Characteristics of the attack, duration, and decay (or envelope) of a given note.
- Intonation: Correctness of a produced pitch as compared to the accepted musical norm.
- Tone: Quality or character of a sound.
- Timbre: Combination of qualities of a sound that distinguishes it from other sounds of the same pitch and volume.

1.4 Matched Filter Based Detection[4]

1.4.1 Shortcomings of the Matched Filter

Upon initial glance, one would be inclined to assume that implementing a simple matched filter would be a fairly straightforward, and relatively precise means of accomplishing this projects goal. This is, however simple incorrect. There are several key issues involved with the implementation of a matched filter that deem it an unsatisfactory algorithm in this particular instance.

[3] This content is available online at <http://cnx.org/content/m13197/1.2/>.
[4] This content is available online at <http://cnx.org/content/m13186/1.3/>.

Upon initial glance, one would be inclined to assume that implementing a simple matched filter would be a fairly straightforward, and relatively precise means of accomplishing this projects goal. This is, however simple incorrect. There are several key issues involved with the implementation of a matched filter that deem it an unsatisfactory algorithm in this particular instance.

Furthermore, a second, and more key issue arises with the implementation of this algorithm. For a matched filter to function correctly, we must be able to match pitches precisely. Herein lies a hidden challenge, detecting what musical pitch the player is attempting to create. This is non-trivial for two reasons. Firstly, and most obviously, not all intonation will be the same. Variants of up to 20 cents in pitch can regularly exist between different performing groups… with that number drastically increasing with extraneous factors, such as the musical maturity of the group. With that issue recognized, let's simple assume that our players are perfectly in tune. A simple analysis of the Fourier Transform does not lead to straightforward detection of pitch, as some have assumed in the past. Simply put, the highest spike in the frequency domain is not necessarily the pitch the artist played, there are a number of instruments, such as the trumpet, where the played pitch is represented by the 3rd (or even higher) harmonics, depending on various conditions. For these reasons, it is very obvious that pitch detection is a non-trivial process, with even the best algorithms incurring some degree of error.

Hence, for these two key reasons, recording variants and pitch detection, along with several other minor issues, it becomes quite obvious that matched filtering is an unacceptable means of implementing instrument recognition.

Chapter 2

Implementation and Results

2.1 System Overview[1]

Pitch and Instrument Recognition System Diagram

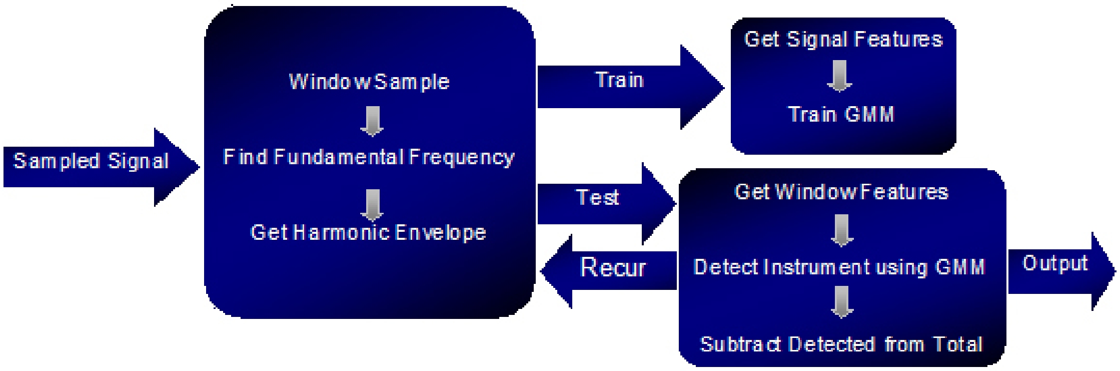


Figure 2.1: System Flowchart.

The system takes some training songs and creates an output vector of features that characterize the signal. A Gaussian mixture model (GMM) is trained to identify patterns and predict an output instrument classification given a set of features.

Each digitized signal was windowed into smaller chunks for feature processing. In training, features were calculated for each window and concatenated into a single vector to be fed into the GMM for training. In testing, features were calculated for each window and fed into the GMM for classification. If multiple notes were to be detected, we recurred on the same window until we found the maximum number of notes or until a note could no longer be detected (as evaluated using a cutoff threshold for what constitutes silence).

From a user standpoint, the user must input a set of training songs, which includes a wav file and the instrument that produced the sound at specific times. Once the system is trained, the user can then input a new song, and our algorithm will output the song in "piano roll" format, i.e. the pitch and instrument of notes plotted over time.

[1] This content is available online at <http://cnx.org/content/m13200/1.1/>.

2.2 Pitch Detection[2]

2.2.1 Pitch Detection

Detecting the pitch of an input signal seems deceptively simple. Many groups have tackled this challenge by simply taking the Fourier transform of the signal, and then finding the frequency with the highest spectral magnitude. As elegant as it may seem, this approach does not work for many musical instruments. Instead, we have chosen to approach the problem from a more expandable point of view.

One of the problems with finding the fundamental frequency lies in simple definition. In our case, we will define this as being the frequency that the human ear recognizes as being dominate. The human auditory system responds most sensitively to the equivalent of the lowest common denominator of the produced frequencies. This can be modeled by finding the strongest set of frequencies amplitudes, and taking the lowest frequency value of that group. This process is quite effective, though it does rely on the condition that the fundamental frequency actually exists, and isn't just simulated via a combination of higher harmonics. The following example illustrates this more concretely.

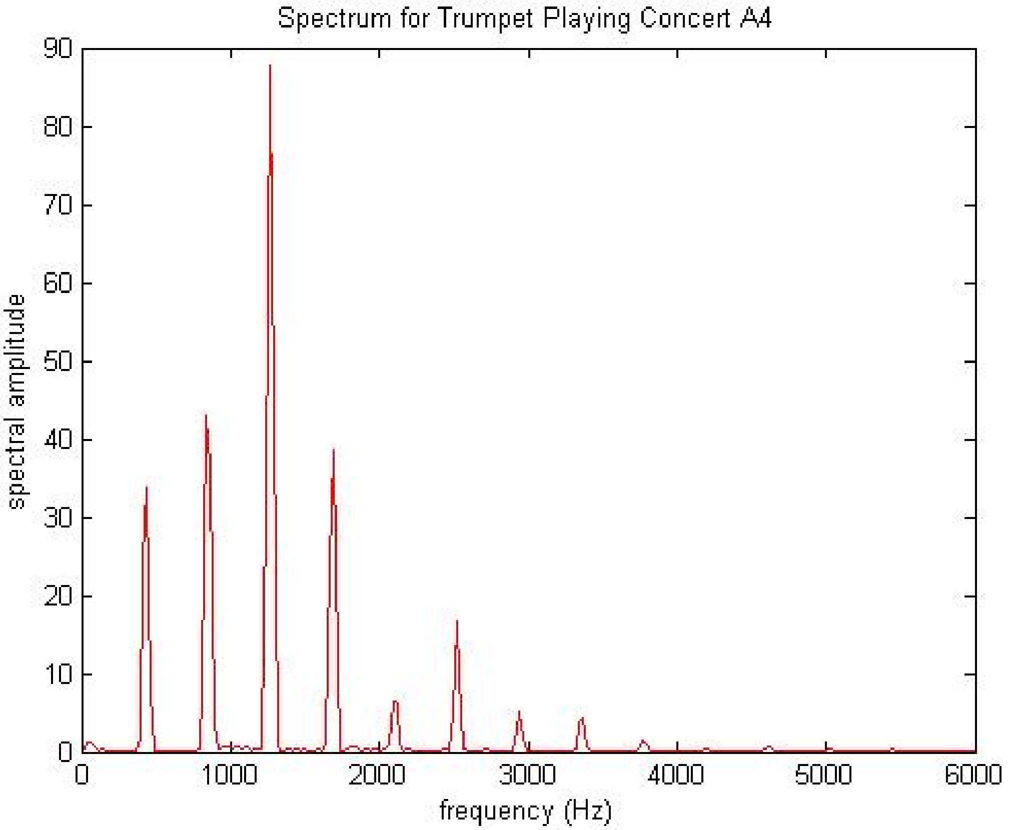


Figure 2.2: Frequence vs. Time for Trumpet playing a concert 'A'=440 Hz

[2] This content is available online at <http://cnx.org/content/m13204/1.2/>.

In the above waveform, we want to find the frequency heard by the human ear as being the fundamental pitch. To do this, we first look at the five highest peaks, which occur at 440, 880, 1320, 1760, and 2640 Hz. From this set of values, we grab the lowest occurring frequency. Hence, the fundamental frequency of the above signal would be stated as being 440 Hz, or a concert 'A'... which is, in fact, the pitch that was played.

2.3 Sinusoidal Harmonic Modeling[3]

2.3.1 Sinusoid Harmonic Modeling

We would like to capture the "typical" spectrum for each instrument, independent of the pitch being produced. This allows us to classify a signal using our model without providing the pitch as another parameter to the model. (We note that this method is not without consequences, as the frequency response of the instrument changes the spectrum depending on the note being played. For example, very low and very high notes are more likely to vary than notes at mid-range. We decided to go with this approach to save time in model training and hopefully reduce the dimensionality of our problem.)

Sinusoidal harmonic modeling (SHM) captures the harmonic envelope of a signal (as opposed to its spectral envelope) and is ideal for tonal sounds produced by wind instruments, as most of the spectral energy is captured in the harmonics. Given a spectrum, SHM finds the fundamental frequency and estimates the harmonics and the harmonic amplitudes, eventually producing a amplitude versus harmonic graph.

[3] This content is available online at <http://cnx.org/content/m13206/1.1/>.

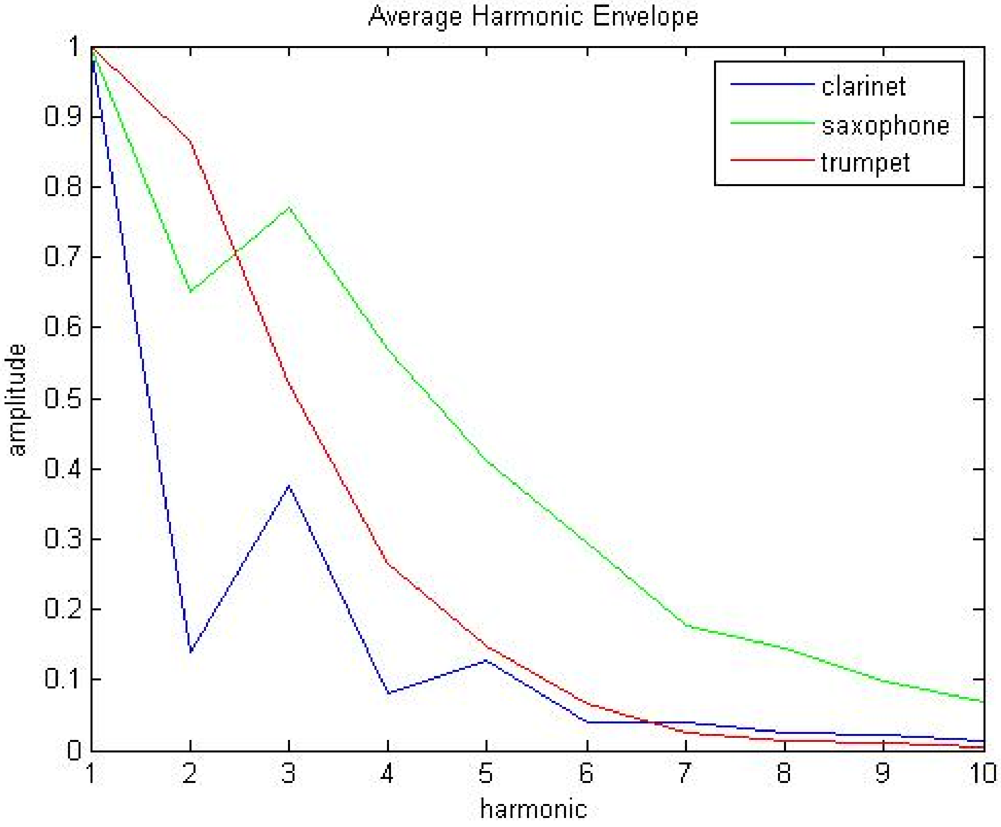


Figure 2.3: Average Harmonic Envelope for Clarinet (Blue), Tenor Sax (Green), and Trumpet (Red)

From this representation, we can then determine characteristic features of the instrument. For example, qualitatively, we can tell that the spectrum of a clarinet declines rather fast, and that most of the energy is in the odd harmonics. Similarly, we can tell that the saxophone declines slower, and that the trumpet has its harmonic energies relatively distributed among the odd and even indices.

2.4 Audio Features[4]

How do we decide what parts of the spectrum are important? The CUIDADO project(2) (p. 9) provided a set of 72 audio features, and research1 has shown that some of the features are more important in capturing the signal characteristics. We therefore decided to implement a small subset of these features:

Cepstral Features

- Mel-Frequency Cepstrum Coefficients (MFCC), k = 2:13

Spectral Features

- Slope

[4] This content is available online at <http://cnx.org/content/m13188/1.3/>.

- Roll-Off
- Centroid
- Spread
- Skew
- Kurtosis
- Odd-to-Even Harmonic Energy Ratio (OER)
- Tristimulus

2.4.1 Definitions

Cepstral coefficients have received a great deal of attention in the speech processing community, as they try to extract the characteristics of the filter and model it independently of the signal being produced. This is ideal, as the filter in our case is the instrument that we are trying to recognize. We work on a Mel scale because it more accurately models how the human auditory system perceives different frequencies, i.e. it gives more weight to changes at low frequencies as humans are more adept at distinguishing low frequency changes.

The centroid correlates to the "brightness" of the sound and is often higher than expected due to the energy from harmonics above the fundamental frequency. The spread, skew, and kurtosis are based on the 2nd, 3rd, and 4th moments and, along with the slope, help portray spectral shape.

Odd-to-even harmonic energy ratio simply determines whether a sound consists primarily of odd harmonic energy, of even harmonic energy, or whether the harmonic energy is equally spread.

The tristimulus measure energy as well and were introduced as the timbre equivalent to the color attributes of vision. Like the OER, it provides clues regarding the distribution of harmonic energy, this time focusing on low, mid, and high harmonics rather than odd and even harmonics. This gives more weight to the first few harmonics, which are perceptually more important.

2.4.2 How We Chose Features

MFCC have shown to work very well in monophonic environments, as they capture the shape of the spectrum very effectively. Unfortunately, they are of less use in polyphonic recordings, as the MFCC captures the shape of a spectrum calculated from multiple sources. Most of the work we have seen on this subject uses MFCC regardless, however. They are particularly useful if only one instrument is playing or is relatively quite salient.

Most wind instruments have their harmonics evenly spread among the odd and even indices, but the clarinet is distinct in that it produces spectra consisting predominantly of odd ratios, with very little even harmonics appearing at all. This makes sense from a physics standpoint, as when played, the clarinet becomes a closed cylinder at one end, therefore allowing only the odd harmonics to resonate. This feature was thus chosen primarily with clarinet classification in mind.

We chose the roll-off and tristimulus as our energy measures, as they were both easy to implement and judged to be important(1) (p. 9). Finally, the first four spectral moments and the spectral slope, in both perceptual and spectral models, were shown to be the top ten most important features in the same study and were therefore some of the first features added to our classification system. We note that we had hoped to implement a perceptual model and thereby nearly double our features, but we could not find an accurate filter model for the mid-ear and thus decided to forgo any features based on perceptual modeling.

For further discussion of these features, along with explicit mathematical formulas, please refer to (1) (p. 9).

2.4.3 References

1. A.A. Livshin and X. Rodet. "Musical Instrument Identification in Continuous Recordings," in Proc. of the 7th Int. Conference on Digital Audio Effects, Naples, Italy, October 5-8, 2004.

2. G. Peeters. "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," 2003. URL: http://www.ircam.fr/anasyn/peeters/ARTICLES/Peeters_2003_cuidadoaudiofeatures.pdf.

2.5 Problems in Polyphonic Detection[5]

The techniques we used for our recognition system have for the most part been applied to monophonic recordings. In moving to polyphonic recordings, we have to subtract out the portion of the sound signal due to the first note and repeat the pitch detection and instrument recognition algorithm for each successive note. We implemented a simple masking function to remove the portions of the spectrum contributed by the detected note. Our first trial used a simple binary mask and removed all the harmonics given a fundamental frequency, but this has the problem of removing potentially significant information if the next note is a harmonic of the first, as their spectrum would therefore overlap. We thus decided to use a more intelligent mask and remove parts of the spectrum using knowledge about the instrument that produced the note. The mask was constructed from the average harmonic envelope and fundamental frequency in a process similar to the inverse of sinusoidal harmonic modeling. However, we only work with the spectral amplitude and not with phase. Because the spectral amplitude of multiple notes is not linear, however, the harmonic peaks in a polyphonic tune cannot simply be subtracted as we have done. We note that accounting for the phase differences is a non-trivial problem, and the simplifying assumption of linearity in spectral amplitude is often used in polyphonic systems.

2.6 Experimental Data and Results[6]

2.6.1 Experimental Data

2.6.1.1 Training

For our training set, we used purely monophonic recordings to ease manual classification. One full chromatic scale was recorded for each of our three instruments. We note that our training is weak, as we only provided one recording for each instrument. By covering the full range of the instrument, we give roughly equal weights to every note, whereas most instruments have a standard playing range and rarely play in the lower or upper limits or their range. Since the spectrum is more apt to skewing effects in the extreme ranges, our average spectral envelope and training features are also negatively affected.

Finally, if we wanted our training set to perform better with polyphonic recordings, we would in practice also provide a few polyphonic recordings as part of our training set. This would allow features unique to polyphonic environments to be modeled as well. For example, a clarinet and trumpet usually cover the melody and are therefore more predominant than a tenor saxophone.

2.6.1.2 Testing

One short monophonic tune per instrument was recorded, as well as two short polyphonic tunes with each instrument combination (clarinet + saxophone, clarinet + trumpet, saxophone + trumpet, all), generating a total of 9 recordings.

2.6.2 Results

2.6.2.1 Self-Validation

We first tested our GMM with the training set to determine how accurate it would be at classifying the data that trained it. The confusion matrix is shown below. (Our confusion matrix shows the actual classification

[5] This content is available online at <http://cnx.org/content/m13207/1.1/>.
[6] This content is available online at <http://cnx.org/content/m13201/1.1/>.

at the left, and the predicted classification at the top.)

	Clarinet	Saxophone	Trumpet
Clarinet	90.0%	7.5%	2.5%
Saxophone	2.9%	92.3%	4.9%
Trumpet	0.9%	11.5%	87.5%

Table 2.1: **Table 1:** Confusion matrix for instrument recognition with training data.

2.6.2.2 Monophonic Recordings

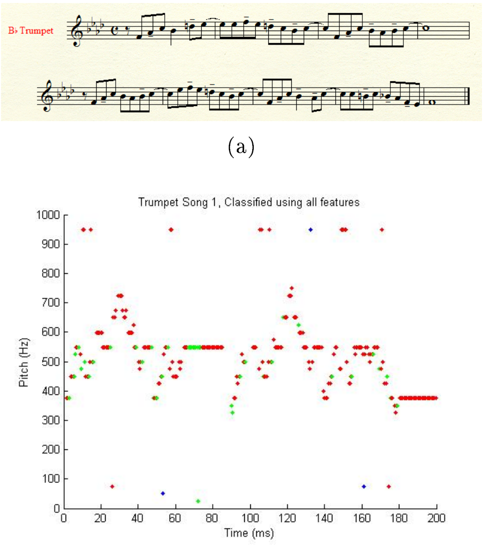
Satisfied that our GMM could classify the training data accurately, we then tested it on a new set of monophonic recordings.

	Clarinet	Saxophone	Trumpet
Clarinet	67.0%	15.1%	17.9%
Saxophone	19.7%	73.0%	7.3%
Trumpet	1.0%	14.9%	84.1%

Table 2.2: **Table 2:** Confusion matrix for instrument of single notes from monophonic recordings.

Average instrument identification using our GMM was 75%, whereas pure guessing would land us at 33.3%. We also see that in our test data, the clarinet and saxophone are confused the most often and can therefore be considered the most similar. This makes sense as both belong to the same instrument family (woodwinds), whereas a trumpet is a brass instrument. In contrast, the clarinet and the trumpet were confused the least often, which is also as expected since their spectrum represent the two extremes within our tested instruments. We are unsure of why the clarinet is often mistaken as a trumpet, but a trumpet is not mistaken as a clarinet, but we believe part of the problem may lie again in our training data, as the self-validation tests showed that the clarinet and trumpet were almost exclusive of one another, and our GMM may have started to memorize the training data.

The following figures show the performed piece of music and the results of our detection and classification algorithm. We note that some discrepancies are due to player error (key fumbles, incorrect rhythmic counting, etc). We follow our coloring scheme of blue representing clarinet, green saxophone, and red trumpet.



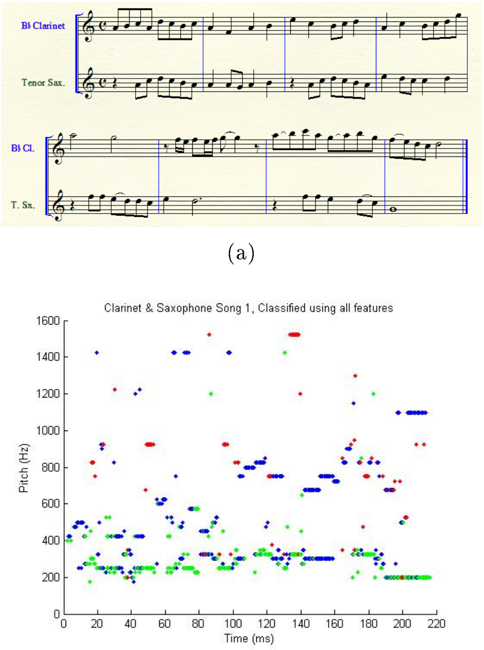
(a)

(b)

Figure 2.4: Original score versus output from our algorithm for a monophonic trumpet tune.

2.6.2.3 Polyphonic Recordings

Finally, we input some polyphonic recordings and compared the experimental outputs to the input music. Quantitative validation is not provided, as it would require us to manually feed into the validation program which instruments at what time. Visually however, we can clearly see that our algorithm correctly separates the melody line, as played by the clarinet, from the lower harmony line, as played by the tenor saxophone.



(a)

(b)

Figure 2.5: Original score versus output from our algorithm for a polyphonic piece.

2.7 Gaussian Mixture Model[7]

2.7.1 Gaussian Mixture Model

A Gaussian Mixture Model (GMM) was used as our classification tool. As our work focused mainly on signal processing, we forgo a rigorous treatment of the mathematics behind the model in favor of a brief description of GMMs and its application to our system.

GMMs belong to the class of pattern recognition systems. They model the probability density function of observed variables using a multivariate Gaussian mixture density. Given a series of inputs, it refines the weights of each distribution through expectation-maximization algorithms.

In this respect, GMMs are very similar to Support Vector Machines[8] and Neural Networks[9] , and all of these models have been used in instrument classification (1) (p. 15). Reported success (2) (p. 15) with GMMs prompted us to use this model for our system.

2.7.2 Recognizing Spectral Patterns

We use 9 features in our recognition program and relied on the GMM to find patterns that would associate these features to the correct instrument. Some of our features consist of a vector (we used 12 MFCC, and

[7] This content is available online at <http://cnx.org/content/m13205/1.1/>.

[8] http://cnx.rice.edu/content/m13131/latest/

[9] http://cnx.rice.edu/content/m11667/latest/

tristimulus has 3 components), so we are actually working in 22 dimension space. For convenience, we focus here on recognizing a pattern between the instrument and two of these dimensions, using the first two MFCC coefficients as an example.

Looking at the distribution of features for the three instruments in figure 1, we clearly see that there are some feature differences based on instrument.

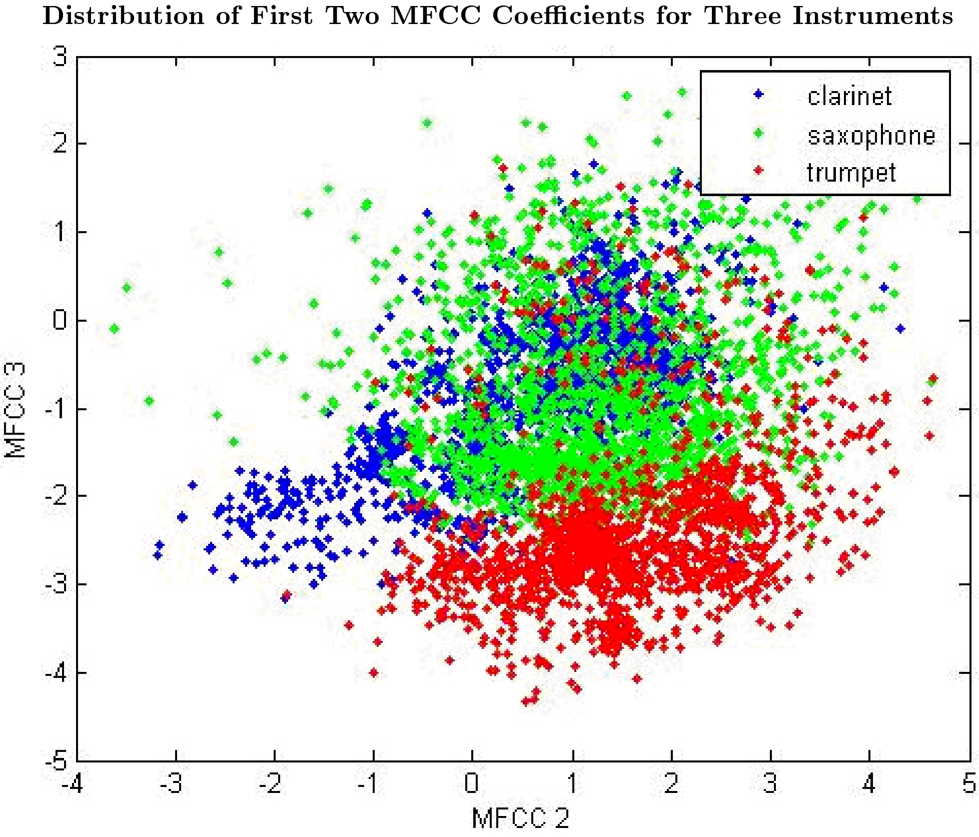


Figure 2.6: Despite the heavy overlap, we see that each instrument dominates different sections of the cepstral space.

GMM detects the patterns in these features and gives us a nice decision rule, as pictured in figure 2. Based on these two features alone, the GMM tells us which instrument most likely played the note, visually represented by the highest peak in the three-dimensional representation.

Two-Parameter Gaussian Mixture Model for Three Instruments

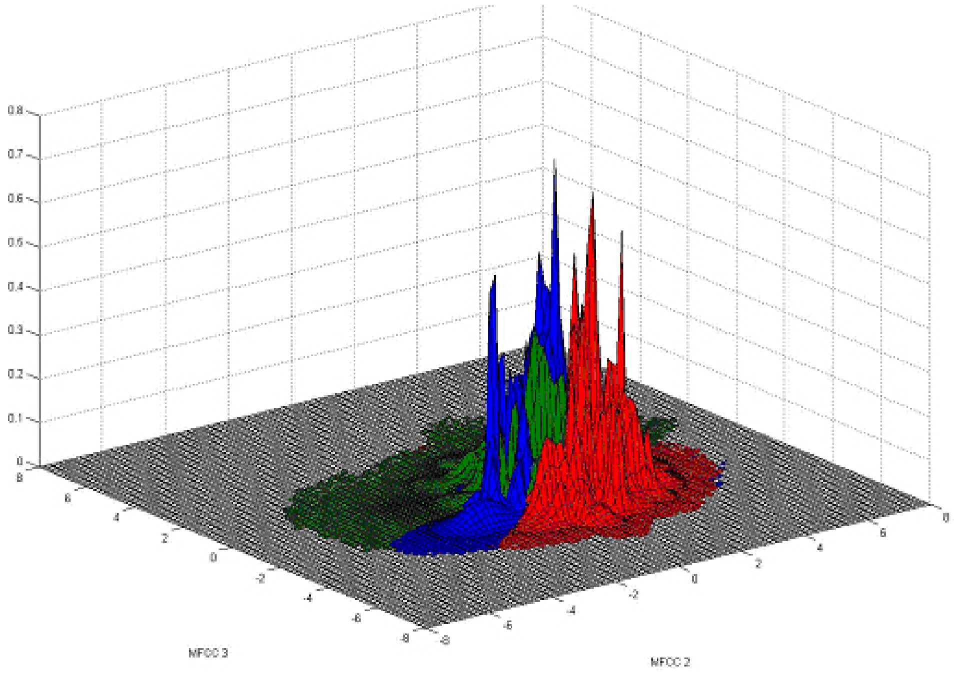


Figure 2.7: Gaussian Mixture Model for Clarinet (blue), Saxophone (green), and Trumpet (red). Signals with features falling in a colored area are classified as a particular instrument. (Gray represents indeterminate instrument.)

Finally, we note that GMMs have been shown to be useful if features are particularly weak or missing (2) (p. 15). This is of particular importance in polyphonic environments, as harmonics may overlap, thus causing some features to be unreliable measures of the instrument.

2.7.3 References

1. A. Brenzikofer. “Instrument Recognition and Transcription in Polyphonic Music.” Term Project. Ecole Polytechnique Federale de Lausanne, June 2004. URL: http://www.brenzi.ch/data/murec-report-web.pdf
2. J. Eggink and G.J. Brown. “A Missing Feature Approach to Instrument Identification in Polyphonic Music,” in IEEE International Conference on Acoustics, Speech, and Signal Processing, Hong Kong, April 2003, 553-556.
3. D. Ellis. Musical Content Analysis: A Practical Investigation of Singing Detection. URL: http://www.ee.columbia.edu/∼dpwe/muscontent/practical/index.html

Chapter 3

Wrap-Up and The Team

3.1 Future Work in Musical Recognition[1]

A number of changes and additions to this project would help it to scale better and be more statistically accurate. Such changes should help the project to handle more complex signals and operate over a larger number of musical instruments.

3.1.1 Improving the Gaussian Mixture Model

To improve the statistical accuracy, the Gaussian Mixture Model used in this project must improve. The features of this model help determine its accuracy, and choosing appropriate additional features is a step towards improving the project. These features may include modeling additional temporal, spectral, harmonic and perceptual properties of the signals, and will help to better distinguish between musical instruments. Temporal features were left out of this project, as they are difficult to analyze in polyphonic signals. However, these features are useful in distinguishing between musical instruments. Articulation, in particular, is useful in distinguishing a trumpet sound, and articulation is by its very nature a temporal feature.

Additionally, more analysis of what features are included in the Gaussian Mixture Model is necessary to improve the statistical accuracy. Too many features, or features that do not adequately distinguish between the instruments, can actually diminish the quality of the output. Such features could respond to the environment noise in a given signal, or to differences between players on the same instrument, more easily than they distinguish between instruments themselves, and this is not desirable. Ideally, this project would involve retesting the sample data with various combinations of feature sets to find the optimal Gaussian Mixture Model.

3.1.2 Improving training data

As training data for this experiment, we used chromatic scales for each instrument over its entire effective range, taken in a single recording session in a relatively low noise environment. To improve this project, the GMM should be trained with multiple players on each instrument, and should include a variety of music - not just the chromatic scale. It should also inlude training data from a number of musical environments with varying levels of noise, as the test data that later is passed through the GMM can hardly be expected to be recorded under the same conditions as the training recordings.

Additionally, the training of the GMM would be improved if it could be initially trained on some polyphonic signals, in addition to the monophonic signals that it is currently trained with. Polyphonic training data was left out of this project due to the complexity of implementation, but it could improve the statistical accuracy of the GMM when decomposing polyphonic test signals.

[1] This content is available online at <http://cnx.org/content/m13196/1.3/>.

3.1.3 Increasing the scope

In addition to training the GMM for other players on the three instruments used in this project, to truly decode an arbitrary musical signal, additional instruments must be added. This includes other woodwinds and brass, from flutes and double reeds to french horns and tubas, to strings and percussion. The GMM would likely need to extensively train on similar instruments to properly distinguish between them, and it is unlikely that it would ever be able to distinguish between the sounds of extremely similar instruments, such as a trumpet and a cornet, or a baritone and a euphonium. Such instruments are so similar that few humans can even discern the subtle differences between them, and the sounds produced by these instruments vary more from player to player than between, say, a trumpet and a cornet.

Further, the project would need to include other families of instruments not yet taken into consideration, such as strings and percussion. Strings and tuned percussion, such as xylophones, produce very different tones than wind instruments, and would likely be easy to decompose. Untuned percussion, however, such as cymbals or a cowbell, would be very difficult to add to this project without modifying it, adding features specifically to detect such instruments. Detecting these instruments would require adding temporal features to the GMM, and would likely entail adding an entire beat detection system to the project.

3.1.4 Improving Pitch Detection

For the most part, and especially in the classical genre, music is written to sound pleasing to the ear. Multiple notes playing at the same time will usually be harmonic ratios of one another, either thirds, or fifths, or octaves. With this knowledge, once we have determined the pitch of the first note, we can determine what pitch the next note is likely to be. Our current system detects the pitch at each window without any dependence on the previously detected note. A better model would track the notes and continue detecting the same pitch until the note ends. Furthermore, Hidden Markov Models have been shown useful in tracking melodies, and such a tracking system could also be incorporated for better pitch detection.

3.2 Acknowledgements and Inquiries[2]

The team would like to thank the following people and organizations.

- Department of Electrical and Computer Engineering, Rice University
- Richard Baraniuk, Elec 301 Instructor
- William Chan, Elec 301 Teaching Assistant
- Music Classification by Genre. Elec 301 Project, Fall 2003. Mitali Banerjee, Melodie Chu, Chris Hunter, Jordan Mayo
- Instrument and Note Identification. Elec 301 Project, Fall 2004. Michael Lawrence, Nathan Shaw, Charles Tripp.
- Auditory Toolbox.[3] Malcolm Slaney
- Netlab.[4] Neural Computing Research Group. Aston University

For the Elec 301 project, we gave a poster[5] presentation on December 14, 2005. We prefer not to provide our source code online, but if you would like to know more about our algorithm, we welcome any questions and concerns. Finally, we ask that you reference us if you decide to use any of our material.

[2] This content is available online at <http://cnx.org/content/m13203/1.3/>.

[3] http://rvl4.ecn.purdue.edu/~malcolm/interval/1998-010/

[4] http://www.ncrg.aston.ac.uk/netlab/intro.php

[5] http://cnx.org/content/m13203/latest/poster.pdf

3.3 Patrick Kruse[6]

3.3.1 Patrick Alan Kruse

Figure 3.1: Patrick Kruse

Patrick is a junior Electrical Engineering major from Will Rice College at Rice University. Originally from Houston, Texas, Patrick intends on specializing in Computer Engineering and pursuing a career in industry after graduation, as acadamia frightens him.

[6] This content is available online at <http://cnx.org/content/m13189/1.1/>.

3.4 Kyle Ringgenberg[7]

3.4.1 Kyle Martin Ringgenberg

Figure 3.2: Kyle Ringgenberg

Originally from Sioux City, Iowa... Kyle is currently a junior electrical engineering major at Rice University. Educational interests rest primarily within the realm of computer engineering. Future plans include either venturing into the work world doing integrated circuit design or remaining in academia to pursue a teaching career.

Outside of academics, Kyle's primary interests are founded in the musical realm. He's performs regularly on both tenor saxophone and violin under the genres of jazz, classical, and modern. He also has a strong interest in 3d computer modeling and animation,which has remained a self-taught hobby of his for years. Communication can be established via his personal website, www.KRingg.com[8] , or by the email address listed under this Connections course.

[7] This content is available online at <http://cnx.org/content/m13185/1.2/>.

[8] http://www.KRingg.com/

3.5 Yi-Chieh Jessica Wu[9]

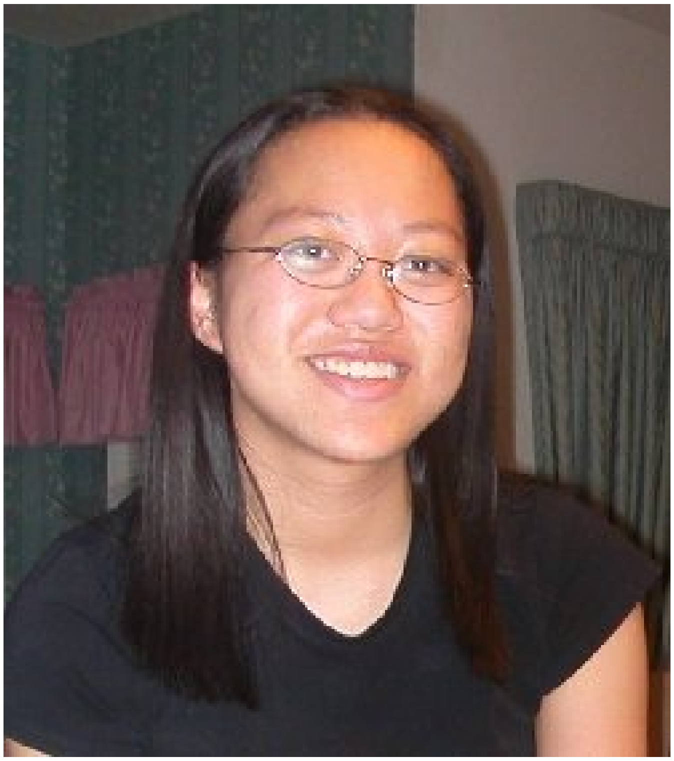


Figure 3.3: Jessica Wu

Jessica is currently a junior electrical engineering major from Sid Richardson College at Rice University. She is specializing in systems and is interested in signal processing applications in music, speech, and bioengineering. She will probably pursue a graduate degree after Rice.

[9] This content is available online at <http://cnx.org/content/m13202/1.1/>.

Index of Keywords and Terms

Keywords are listed by the section with that keyword (page numbers are in parentheses). Keywords do not necessarily appear in the text of the page. They are merely associated with that section. *Ex.* apples, § 1.1 (1) **Terms** are referenced by the page they appear on. *Ex.* apples, 1

Attributions

Collection: *Musical Instrument Recognition*
Edited by: Kyle Ringgenberg, Yi-Chieh Wu, Patrick Kruse
URL: http://cnx.org/content/col10313/1.3/
License: http://creativecommons.org/licenses/by/2.0/

Module: "Introduction"
By: Patrick Kruse, Kyle Ringgenberg
URL: http://cnx.org/content/m13195/1.3/
Page: 1
Copyright: Patrick Kruse, Kyle Ringgenberg
License: http://creativecommons.org/licenses/by/2.0/

Module: "Simple Music Theory as it relates to Signal Processing"
Used here as: "Mathematics of Music Theory"
By: Michael Lawrence
URL: http://cnx.org/content/m12461/1.5/
Pages: 1-2
Copyright: Michael Lawrence
License: http://creativecommons.org/licenses/by/1.0

Module: "Common Music Terms"
By: Kyle Ringgenberg
URL: http://cnx.org/content/m13197/1.2/
Page: 2
Copyright: Kyle Ringgenberg
License: http://creativecommons.org/licenses/by/2.0/

Module: "Matched Filter Based Detection"
By: Kyle Ringgenberg
URL: http://cnx.org/content/m13186/1.3/
Pages: 2-3
Copyright: Kyle Ringgenberg
License: http://creativecommons.org/licenses/by/2.0/

Module: "System Overview"
By: Yi-Chieh Wu
URL: http://cnx.org/content/m13200/1.1/
Page: 5
Copyright: Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Module: "Pitch Detection"
By: Yi-Chieh Wu, Kyle Ringgenberg
URL: http://cnx.org/content/m13204/1.2/
Pages: 6-7
Copyright: Yi-Chieh Wu, Kyle Ringgenberg
License: http://creativecommons.org/licenses/by/2.0/

Module: "Sinusoidal Harmonic Modeling"
By: Kyle Ringgenberg, Yi-Chieh Wu
URL: http://cnx.org/content/m13206/1.1/
Pages: 7-8
Copyright: Kyle Ringgenberg, Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Module: "Audio Features"
By: Yi-Chieh Wu
URL: http://cnx.org/content/m13188/1.3/
Pages: 8-10
Copyright: Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Module: "Problems in Polyphonic Detection"
By: Yi-Chieh Wu
URL: http://cnx.org/content/m13207/1.1/
Page: 10
Copyright: Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Module: "Experimental Data and Results"
By: Yi-Chieh Wu
URL: http://cnx.org/content/m13201/1.1/
Pages: 10-13
Copyright: Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Module: "Gaussian Mixture Model"
By: Yi-Chieh Wu
URL: http://cnx.org/content/m13205/1.1/
Pages: 13-15
Copyright: Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Module: "Future Work in Musical Recognition"
By: Patrick Kruse, Kyle Ringgenberg, Yi-Chieh Wu
URL: http://cnx.org/content/m13196/1.3/
Pages: 17-18
Copyright: Patrick Kruse, Kyle Ringgenberg, Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Module: "Acknowledgements and Inquiries"
By: Yi-Chieh Wu
URL: http://cnx.org/content/m13203/1.3/
Page: 18
Copyright: Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Module: "Patrick Kruse"
By: Patrick Kruse
URL: http://cnx.org/content/m13189/1.1/
Page: 19
Copyright: Patrick Kruse
License: http://creativecommons.org/licenses/by/2.0/

Module: "Kyle Ringgenberg"
By: Kyle Ringgenberg
URL: http://cnx.org/content/m13185/1.2/
Page: 20
Copyright: Kyle Ringgenberg
License: http://creativecommons.org/licenses/by/2.0/

Module: "Yi-Chieh Jessica Wu"
By: Yi-Chieh Wu
URL: http://cnx.org/content/m13202/1.1/
Page: 21
Copyright: Yi-Chieh Wu
License: http://creativecommons.org/licenses/by/2.0/

Musical Instrument Recognition
To detect the pitch and instrument of a monophonic signal. To decompose polyphonic signals into their component pitches and instruments by analyzing the waveform and spectra of each instrument. Elec 301 Project Fall 2005.