Title
Real time vibraphone pitch and timbre classification :

Permalink
https://escholarship.org/uc/item/5q19z937

Author
Larke, Kevin

Publication Date
2008

Peer reviewed|Thesis/dissertation
Real Time Vibraphone Pitch and Timbre Classification

A thesis submitted in partial satisfaction of the requirements for the degree Master of Arts in Music

by

Kevin Larke

Committee in charge:

Shlomo Dubnov, Chair
Miller Puckette
Richard Moore

2008
The Thesis of Kevin Larke is approved, and it is acceptable in quality and form for publication on microfilm:

University of California, San Diego

2008
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature Page</td>
<td>iii</td>
</tr>
<tr>
<td>Table of Contents</td>
<td>iv</td>
</tr>
<tr>
<td>List of Figures</td>
<td>v</td>
</tr>
<tr>
<td>List of Tables</td>
<td>vi</td>
</tr>
<tr>
<td>Abstract</td>
<td>vii</td>
</tr>
<tr>
<td>1 Overview</td>
<td>1</td>
</tr>
<tr>
<td>1 About the vibraphone</td>
<td>2</td>
</tr>
<tr>
<td>2 Research</td>
<td>4</td>
</tr>
<tr>
<td>A Pitch Tracking</td>
<td>4</td>
</tr>
<tr>
<td>B Percussion Classification</td>
<td>5</td>
</tr>
<tr>
<td>C Onset Detection</td>
<td>6</td>
</tr>
<tr>
<td>2 Vibraphone Tracking</td>
<td>8</td>
</tr>
<tr>
<td>1 The Fourier Transform and the Vibraphone</td>
<td>8</td>
</tr>
<tr>
<td>2 Peak Selection</td>
<td>11</td>
</tr>
<tr>
<td>3 Spectral and Peak Distance Measures</td>
<td>15</td>
</tr>
<tr>
<td>4 Template Representation</td>
<td>21</td>
</tr>
<tr>
<td>5 Onset Detection</td>
<td>27</td>
</tr>
<tr>
<td>6 Template Generator</td>
<td>32</td>
</tr>
<tr>
<td>7 Evaluating the system</td>
<td>34</td>
</tr>
<tr>
<td>8 Future Work</td>
<td>37</td>
</tr>
<tr>
<td>9 Octave Toolbox</td>
<td>40</td>
</tr>
<tr>
<td>10 Conclusion</td>
<td>40</td>
</tr>
<tr>
<td>References</td>
<td>42</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Spectra of several F#4 strikes.</td>
<td>3</td>
</tr>
<tr>
<td>2.2</td>
<td>Spectral whitening for F#4 (369.99 Hertz)</td>
<td>12</td>
</tr>
<tr>
<td>2.3</td>
<td>Peak selection example</td>
<td>13</td>
</tr>
<tr>
<td>2.4</td>
<td>Triangle templates and euclidean distance for all templates</td>
<td>18</td>
</tr>
<tr>
<td>2.5</td>
<td>Mode selection using summed magnitude.</td>
<td>24</td>
</tr>
<tr>
<td>2.6</td>
<td>Mode selection using frequency clustering.</td>
<td>25</td>
</tr>
<tr>
<td>2.7</td>
<td>Vibraphone mode analysis</td>
<td>27</td>
</tr>
<tr>
<td>2.8</td>
<td>Measurements related to onset detection.</td>
<td>28</td>
</tr>
<tr>
<td>2.9</td>
<td>System evaluation interface</td>
<td>36</td>
</tr>
<tr>
<td>2.10</td>
<td>30 pole linear predictive coding on a 512 sample window</td>
<td>39</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Time and frequency for Fourier analysis windows.</td>
<td>10</td>
</tr>
<tr>
<td>2.2</td>
<td>Template Generator Parameters.</td>
<td>35</td>
</tr>
<tr>
<td>2.3</td>
<td>Onset evaluation report.</td>
<td>37</td>
</tr>
<tr>
<td>2.4</td>
<td>System Statistics.</td>
<td>38</td>
</tr>
</tbody>
</table>
This paper examines the problem of classifying vibraphone notes in real-time based only on information from the instrument’s acoustic signal. A system consisting of two basic parts is proposed: an attack identification unit for classifying the attack transients of new notes and a tone follower for tracking the overtones of sustaining notes. Both systems use a k-Nearest-Neighbor algorithm to match the unknown signal to prototype feature vectors. Several alternate feature representations are discussed including magnitude spectra and frequency domain peak information. Finally an informal evaluation of the proposed system is examined.
1 Overview

This thesis discusses issues related to tracking an acoustic vibraphone signal in real-time. It attempts to take a systematic approach to examine how some basic signal measurement techniques succeed and fail when applied to audio generated during a vibraphone performance. As part of this examination a classification based tracking stem is proposed and some of its properties are described. Finally an informal evaluation of the system is given.

Following this introduction the paper is organized into two sections. The first section discusses the vibraphone and existing research related to pitch tracking, onset detection, and percussive signal classification. The second section describes the methods used by the proposed system in more detail.

The primary goal of a vibraphone tracking system is to report the pitch of a sounding bar as quickly as possible following the note onset. In order to accomplish this the follower must be able to notice that an attack occurred and then identify which bar was struck. If the notes were sounded one at a time in an acoustically controlled setting this would be a fairly easy task. In practice however the instrument is played polyphonically and in the presence of other instruments. This leads to the central problem of tracking any polyphonic instrument using a microphone; at any given time multiple notes may be ringing and the signal may be corrupted by input from other sources. As new notes are played they will therefore be obscured by other sounding notes on the same instrument and by other sounds in the environment.

Seen from this perspective the task of polyphonic pitch tracking can be
broadly broken into three sub-tasks: noticing new notes, tracking existing notes, and rejecting other sounds. For the most part this paper deals with the first two issues. The problem of extraneous sounds is either ignored or handled implicitly by the classification techniques.

The complete system described here is implemented as a collection of several hundred octave (Eaton (2002)) functions. The functions are documented here: Octave Toolbox Appendix A gives some useful entry points into the code.

1.1 About the vibraphone

The vibraphone consists of 37 bars with notes ranging from F3 (174.61 Hertz) to F6 (1396.91 Hertz). Each bar has an attached tubular resonator intended to emphasize the fundamental frequency of the bar. Each tube also holds a motor controlled rotating baffle. For the purposes of the system described here the baffles are considered to be turned off. The instrument also has a foot-pedal controlled damper pad. With the damper on (pedal up) each note will ring for a few seconds depending on the force of the strike. With the damper off the bar can easily ring for tens of seconds.

The spectrum of the acoustic signal from a struck vibraphone bar consists of the fundamental, the fourth harmonic of the fundamental, and a collection of significantly non-harmonic overtones (Fletcher and Rossing (1991)). Figure 1.1 shows three views of the spectrum of an F4 (349.23 Hertz). Some sources also claim a third tuned harmonic an octave and a major third above the fourth harmonic (Moore (1970)).

These three graphs show several typical features of a vibraphone spectrum. The vertical lines on the top two graphics show the position of the harmonics of the fundamental. Notice that with the exception of the fourth harmonic the overtones (as indicated by the spectral peaks) do not line up with the location of harmonics of the fundamental.
The spectrum in the top plot was measured from a bar struck with a soft mallet made of wound yarn. The center spectrum comes from a strike made with a hard rubber mallet. This second spectrum is notable in that it contains two characteristics which are contrary to conventional wisdom regarding musical spectra. First the fundamental contains noticeably less energy than many other overtones in the spectrum. Second the overall shape of the spectrum shows a clear tendency to increase with frequency.

The lower plot contains two overlayed magnitude spectrums measured from two different instruments struck with same wound yarn mallet. The red vertical lines are positioned at the fundamental and fourth harmonic.

The overtones of a struck bar result from the vibrational modes of the bar. Differences in the bar material or in the geometry of the bars will therefore shift the height and location of the overtones. Variations between the two instruments shown in the lower plot result in a difference in energy at the fourth harmonic and also slightly different overtone frequencies. As we will see below...
these kinds of variations provide one of the primary challenges to developing an effective instrument follower.

1.2 Research

1.2.A Pitch Tracking

Pitch tracking of digital audio signals has been a major topic in computer music research for more than thirty years. The classical techniques and the principles they rely on continue to have relevance in modern instrument tracking. Pitch following of monophonic signals is often broken into two broad categories: time domain methods and frequency domain methods.

The two most common time domain methods are based on zero-cross measurement (Gold and Rabiner (1969)) and auto-correlation. The zero-cross method is not applicable to polyphonic following where the signals are too complex to be accurately measured this way. Autocorrelation however continues to be used as a fundamental frequency measurement technique for speech and musical instruments (Shimamura and Kobayashi (2001), Tolonen and Karjalainen (2000), Wu et al. (2003)).

The classical frequency domain pitch trackers usually consist of a Fourier transform or auto-regressive time to frequency transform followed by a pitch determination algorithm (PDA). Since pitched signals are generally expected to be quasi-periodic, and therefore mostly harmonic, the PDA often relies on measuring the harmonics of the signal to arrive at the fundamental frequency estimate. Classical examples of PDA’s are harmonic sum, harmonic product and maximum likelihood based algorithms (Noll (1970), Hermes (1988), Logothetis and Saleem (1996)).

Classical techniques tend to be oriented toward monophonic, harmonic and noise free signals. Current techniques have attempted to address less well behaved sources which admit the possibility of multiple simultaneous sounds by
the same or different instruments. These techniques are also more complex in so far as they may employ signal features other than harmonicity.

Klapuri (2006) suggests a mult-pitch follower which bridges the gap between classical frequency following techniques and modern multi-pitch techniques. His system uses harmonic summing to locate the most likely fundamental frequencies in a signal.

In so far as pitch refers to a measurement of auditory perception many current pitch tracking systems are inspired by human auditory models of pitch perception. Two examples of the use of psychoacoustic models of multi-pitch trackers are Tolonen and Karjalainen (2000) and Klapuri (2005).

Multi-pitch analysis is often as much concerned with the problem of source separation as with fundamental frequency estimation. These techniques also rely on machine learning techniques such as Bayesian estimation (Davy and J. (2003)), Support Vector Machines (Poliner and Ellis (2005)) and Non-negative Matrix factorization (Smaragdis and Brown (2003)) to attempt to locate known instrument profiles in a source signal.

### 1.2.B Percussion Classification

Since the vibraphone is a percussion instrument as well as a pitched instrument current research in identifying percussion instruments also applies here. Much of this research centers on the use of supervised learning techniques to match unknown signals to previously learned spectral templates.

Herrera et al. (2002) compares instance based, statistically based and tree based classifiers using a variety of acoustic features to identify various drum sounds. The study attempted to classify individual instruments as well as taxonomies of instruments. Chordia (2005) describes a system for recognizing isolated tabla strokes using a variety of common timbral audio features and machine classification algorithms. Tindale et al. (2004) built a system for following snare gestures in real-time. Steelant et al. (2004) evaluated support vector machines for classifying
polyphonic snare and bass signals.

1.2.C Onset Detection

Like pitch tracking onset detection, has been a major topic of research in recent years. Traditional onset detectors rely on locating energy transients in the spectrum to identify note onsets.

The simplest variety of this style of detector use the first order difference of the RMS envelope of the signal to locate transients. When the difference signal crosses a threshold an onset is reported. Fine tuning the threshold values in schemes like this is critical to the effectiveness of the detector. If the threshold is too high the system will tend to miss onsets. If the threshold is too low then non-attack based transients will trigger onset reports.

When quiet notes follow loud notes the onset of the quiet note can easily be missed by a simple RMS onset detector because the attack transient energy of the second note is relatively small compared to the energy of the preceding note. Some onset detectors have adopted multi-frequency band strategies to avoid this problem.

It is not uncommon for a threshold which is slightly too low to trigger multiple onsets as a result of one transient. Many systems handle this problem by setting a minimum time window between onsets others attempt to smooth the detection function decay thereby making a second onset less likely. The time window has the advantage of giving a guaranteed result at the expense of missing closely spaced onsets. The smoothing function gives a more adaptive result but can still let through spurious secondary triggers when a single transient contains multiple amplitude spikes.

A good example of current research in energy based onsets detectors is outlined in Alonso (2007). This system combines several energy based onset techniques into a single detector. The overall algorithm is based on an improved spectral flux model with a psychoacoustics inspired integration filter and a median
filter to handle spurious secondary transients.

More recently phase based onset detectors have been introduced to better detect instruments which do not have a percussive attack. Bello (2004) suggests that the phase change between steady state portions of a signal should be constant and therefore onsets can be located by finding high phase variation between frames (Bello (2004), Holzapfel and Y. (2008) and Lacoste (2007)).

Question the efficacy of phase deviation as an onset detector but suggest variations of their own. Holzafel tested a group delay based onset detection measurement. Lacoste noticed that while phase deviation between frames did not appear to be useful that phase deviation across bins was effective. The details of phase based onset detectors are discussed in greater detail below.

Although vibraphones have percussive attacks phase based onset detectors are of interest here because they might prove useful for noticing onsets of quiet notes that are masked by previous louder notes.

Several authors (Keiler et al. (2003), Bello (2005)) have also suggested using auto-regressive onset detection models. These onset detectors have several appealing characteristics. First they can potentially achieve higher frequency resolution with shorter time windows and second they naturally produce a residual signal which can be directly used as a detection function.

Others have suggested machine learning based onset detectors. Lacoste (2007) used a neural net based classifier trained on spectral features. Wang (2008) used non-negative matrix factorization to find the time basis function of the short-time Fourier transform of an audio signal. The time basis functions are then used as a detection function much the way a transient measurement would in an energy based detector. As in the case of NNMF pitch detectors the use of the NNMF here suggests the idea of simultaneously separating the overlapping polyphonic notes while identifying their onset location.
2 Vibraphone Tracking

The vibraphone tracking system described here is broken into two sub-systems. The first sub-system attempts to detect note onsets and then identify the bar that produced the onset by analyzing features from the note attack. The second sub-system attempts to follow existing notes as they decay over time.

The attack identification system is characterized as having a high time resolution and a low frequency resolution. The primary purpose of this sub-system is to quickly identify onsets and classify the pitch of the struck bar. The sub-system will work on the hypothesis that the short time spectrum of a bar attack transient and the few milliseconds following it contain a distinguishing signature for each bar.

The tone tracking sub-system will follow the slowly varying periodic signal components. This system is designed to track the ongoing state of the system with less time resolution but higher frequency resolution and thereby give a second perspective on the state of the instrument. These measurements will also be used to improve the onset identification by providing the attack identification system with information about current sounding notes.

2.1 The Fourier Transform and the Vibraphone

Both the attack identification and the tone tracker sub-systems will use short time Fourier transform (STFT) based methods to measure the signal spectrum. This section will review the limitations of the STFT technique relative to
the expected vibraphone signals.

The length of the time window of the Fourier transform determines the frequency resolution of the resulting spectrum. The closest two frequencies to be resolved by the vibraphone tracker are determined by the lowest two notes on the instrument. The low note (F3) is 174.61 Hertz and the semitone just above it (F#3) is at 185 Hertz. At the very least the tracker will therefore need support a resolution of 10.39 Hertz.

Assuming a rectangular window function and a sample rate of 44100 samples per second the minimum window length to distinguish two sinusoids separated by 10.39 Hertz is 44100/10.39 or 4245 samples (96 milliseconds).

In practice the frequency domain artifacts resulting from a rectangle window function might prove to be problematic thereby requiring a hann window function. The hann window however has a main lobe width of twice the rectangle window \cite{Harris}. In order to maintain the frequency discrimination of the rectangle window the window length must therefore be doubled to 8490 samples (192 milliseconds).

In some cases the spectrum will be used to locate magnitude peaks associated with overtones in the spectrum. In order to locate a local maxima at least one local minima (valley) bin will need to be located between the peak bins. In effect this doubles the required frequency resolution and therefore requires another doubling of the required time window to 16980 samples (384 milliseconds).

Each increase in the length of the time window comes with an associated decrease in the time localization ability of the transform. In other words as the analysis window lengthens the spectra will become increasingly contaminated with non-local time information. Developing strategies to deal with this trade-off is one of the fundamental problems which confront systems like the one described here.

Table \ref{tab:time_frequency_tradeoff} summarizes the time frequency trade-off for the Fourier transform given a sample rate of 44100 samples per second.

The period of the lowest note on a vibraphone is 5.72 milliseconds. This
Table 2.1: Time and frequency for Fourier analysis windows.

<table>
<thead>
<tr>
<th>Time (Milliseconds)</th>
<th>Time (Samples)</th>
<th>Resolution (Hertz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.45</td>
<td>64</td>
<td>689.06</td>
</tr>
<tr>
<td>2.90</td>
<td>128</td>
<td>344.53</td>
</tr>
<tr>
<td>5.80</td>
<td>512</td>
<td>172.27</td>
</tr>
<tr>
<td>11.61</td>
<td>512</td>
<td>86.13</td>
</tr>
<tr>
<td>23.22</td>
<td>1024</td>
<td>43.07</td>
</tr>
<tr>
<td>24.08</td>
<td>1062</td>
<td>41.52</td>
</tr>
<tr>
<td>46.44</td>
<td>2048</td>
<td>21.53</td>
</tr>
<tr>
<td>92.88</td>
<td>4096</td>
<td>10.76</td>
</tr>
<tr>
<td>96.25</td>
<td>4245</td>
<td>10.39</td>
</tr>
<tr>
<td>185.76</td>
<td>8192</td>
<td>5.38</td>
</tr>
<tr>
<td>371.52</td>
<td>16384</td>
<td>2.69</td>
</tr>
</tbody>
</table>

is relevant in the sense that it gives an indication of a lower bound on the time required in the very best of circumstances (no noise, no attack transients, no confirmation from a second cycle) to complete a single cycle of the lowest note.

From analysis of the vibraphone notes collected for this study the time between the bar strike and the maximum amplitude of the signal is between one and three milliseconds. Adding another couple of milliseconds for the attack transient noise to settle out might indicate an optimistic lowest latency bound (strike time to pitch report time) of around 10 milliseconds for the F3 bar. Given the constraints of the Fourier transform however is several times longer.

A more optimistic analysis of the problem might set out by not attempting to distinguish the fourth harmonic of each note rather than the fundamental. Attempting to distinguish the fourth harmonic of the lowest two notes relaxes the frequency resolution requirements by four times to 41.56 Hertz. Assuming a rectangular window this will require a 94.2 millisecond (1062 samples) analysis window.

Yet another way around this problem would be to not rely on pitch information for identifying attacks. If signature information can be found in other spectral components then waiting for the fundamental frequency to develop may not be necessary.
2.2 Peak Selection

This section describes the strategies the system uses to extract peaks from the spectrum of a signal. The same process of peak selection is used for locating peaks in both the templates and the unknown signals during classification.

The goal of peak selection on a single spectrum is to locate the magnitude peaks which most likely represent an underlying sinusoid in the signal. The output from the peak selection process is a list of peak magnitudes and frequencies. The system described here has two algorithms which can be independently selected for peak selection. The first is a magnitude only selection process. The second algorithm uses a combination of magnitude and an instantaneous frequency estimation.

The magnitude only process works roughly according to intuition. First the largest spectral magnitude peaks above a certain threshold are selected. Second the frequency location of the peaks is refined through a process of parabolic interpolation [Serra and Smith (1990)]. The interpolation procedure uses the magnitude of the peak bin and the magnitude of the bin just above and below it to refine the location of the underlying sinusoid.

Figure 2.2 displays one possible problem with implementing this technique directly. The horizontal red line in this graph shows a hypothetical magnitude threshold. Only local maxima which are above this threshold will be picked as candidate peaks. Because the height of the peaks rolls off with frequency (as is typical of quasi-harmonic sounds) the low frequency peaks are significantly above the threshold whereas higher frequency peaks fall below it. The high frequency peaks would therefore be rejected as potential sinusoid candidates.

One way to eliminate the $1/F$ roll off and thereby level the spectrum is via spectral whitening [Tolonen and Karjalainen (2000)]. The idea of spectral whitening is to flatten (whiten) the spectrum in order to remove the high frequency roll off. The green trace in Figure 2.2 shows the results of the whitening process when applied to the spectrum shown by the blue trace.
The process of spectral whitening as implemented here is a two step process. First a smoothed spectrum is calculated. Then the difference between the maximum value of the smooth spectrum and each magnitude bin is added to each magnitude bin. The whitening process is only used to pick candidate peaks so the fact that energy is added to the whitened spectrum is immaterial. Once the candidates are selected the unwhitened magnitude is used as the peak magnitude. If power needed to be conserved however the median of the smoothed spectrum could be be used in place of the maximum as described above. For spectrums with a 1/F roll off this would have the effect of decreasing the energy below the median and increasing the energy above it.

The smoothed spectrum is shown as the magenta trace in the top graph of Figure 2.2. The smoothed spectrum is calculated by taking a moving average of the the magnitude spectrum. The width of the moving average window is calculated as the mean of the energy in the octave above and below each bin of
the magnitude spectrum. The moving average window therefore increases (when measured in Hertz) as it moves up the spectrum.

Figure 2.2 shows the whitening process applied to magnitude spectrum on a decibel scale in the upper graph and linear scale in the lower graph. The lower graph is particularly interesting because it shows how the whitening process as implemented here can cause a disproportional increase in the lowest peaks in the spectrum. This distortion is an artifact of the smoothing process. Since the smoothing process uses an octave integration window the number of bins in the octave is relatively small for very low frequencies. This means that for low frequency peaks the peak bin and its immediate neighbors make up the only bins in the octave and thereby reinforce the peak. In contrast the octave window above and below higher frequency peaks will include many bins and the contribution from the peak itself will be proportionally less.

Figure 2.3: Peak selection example.

Figure 2.3 shows the result of peak selection on the same spectrum shown
in Figure 2.2. The peaks selected in this graph were found using the magnitude and instantaneous frequency method. In practice the two techniques tend to select the same peaks since the largest magnitude peaks tend to be the best candidates. The estimation of the peak frequency however appears to be more accurate using the instantaneous frequency method as opposed to the magnitude only approach. The increase in frequency accuracy is particularly important for identifying the frequency of low notes which may be critically sampled in the frequency domain.

The instantaneous frequency estimate approach to peak selection is taken from Brown and Puckette (1993). The technique is based on the insight that the phase change for a one sample advance in a sinusoidal component is both unambiguous (not subject to wrapping) and a good estimate of the frequency because it is very localized in time.

The brute force method to measure frequency using this idea would be to measure the phase of the signal us estimating the phase given a one sample advance. The article also demonstrates that the frequency estimate made using this approach is more accurate than the parabolic interpolation technique mentioned above.

The technique as implemented in Brown and Puckette (1993) effectively applies a hann window to the complex spectrum in the frequency domain. The bins on either side of sinusoids measured with a hann windowed spectrum are in phase with the sinusoid. The instantaneous frequency estimate of the neighboring bins therefore tend to be the same as the peak bin. Searching for blocks of repeating frequencies up the spectrum can therefore act as a method of peak searching which is less reliant on magnitude. This is especially useful because it reduces the reliance of the system on setting a peak level threshold. Hard thresholds for peak selection have the effect of producing noisy data because peaks from low level sines will tend to fall randomly above and below the threshold.

Both the magnitude only and the instantaneous frequency peak selection algorithms used in the system described here include a frequency filter window to remove peaks that fall too close together. This window is generally set to
0.5 semitones above and below the magnitude peak. If two peaks fall inside this window then the lower magnitude peak is rejected. This filter is most useful for removing peaks arising from analysis window side lobes in the frequency domain. A better approach to eliminating this problem would be to subtract the peaks from the spectrum from highest to lowest. The subtraction would tend to eliminate both the peak and the side lobes from the spectrum thereby avoiding the problem of side lobe peak selection.

2.3 Spectral and Peak Distance Measures

The classifier used by both the attack identifier and the tone tracker compares labelled template data vectors to data vectors measured from unknown signals. The templates which best match the measured features determine the systems best guess of the signal source.

Several elements of the system rely on distance measures to evaluate how well a feature measured from an unknown signal matches a template feature. Both the attack identification and tone following system attempt to match measured spectral or frequency peak vectors from an unknown signal to templates extracted from hand labelled signals. Both systems also use a k-Nearest-Neighbor (kNN) algorithm (Duda et al. (2001)) to classify unknown feature vectors given a database of templates. The kNN algorithm works by measuring the distance from the vector to be classified to each of the labelled template vectors. The unknown vector is then classified by taking on the label of the most frequently occurring label from among the k closest templates. The metric used to define the distance between the vectors is one important parameter to this algorithm.

For comparing magnitude spectra the kNN algorithm implemented here uses the Euclidean distance between the template and the unknown vector. Euclidean distance satisfies four properties which make it an effective distance metric (Duda et al. (2001)):
Given a distance function $D$ and three vectors $a$, $b$ and $c$:

**non-negativity** $D(a, b) \geq 0$

**reflexivity** $D(a, b) = 0$ if and only if $a = b$

**symmetry** $D(a, b) = D(b, a)$

**triangle inequality** $D(a, b) + D(b, c) \geq D(a, c)$

While Euclidean distance is an obvious choice for measuring the distance between magnitude spectra the method used for comparing peak frequencies is not as clear. The distance between two sets of peak frequencies cannot be measured directly via Euclidean distance because the peak information is represented by discrete points. Unlike the magnitude spectra the peaks are not regularly sampled across the spectrum.

As implemented the classification system uses two different approaches to measuring the distance between vectors of frequency points. The first and simplest is to convert the points into a magnitude spectrum representation. Once converted the generated spectrum can be compared using Euclidean distance just as if they were measured magnitude spectra.

One direct way to accomplish frequency point to spectrum conversion is to start with a zeroed magnitude vector and place a 1.0 in the bin associated with the location of the peak frequency. The resulting vector would then fulfill the requirements necessary to compare using a Euclidean distance measure. In fact the vector could be compared to either measured magnitude spectra or other spectra generated via the same technique.

The main drawback to generating synthetic spectra with this simple approach is that it will produce very sparse vectors. Most of the bins of the generated vector will be zero and the ones representing the peaks will be spread out in discrete locations. This could mean that two vectors which were actually very close could produce a distance measures equal to much less similar spectra because their
bin locations were offset from one another by a single bin. The solution to this problem is to interpolate a peak shape into the spectral bin containing the peak as well as the bins on either side of it.

The shape of the generated peak depends on the spectrum to which it will be compared. If the synthetic peak will be compared to other synthetic peaks then a simple triangle shape may be adequate. Triangles have the nice quality of being easy to generate with a parametric bandwidth argument. If the generated shape will be compared to other measured spectra then the frequency domain shape of the window function used to measure the target spectra should be used (Harris (1978)).

A second obvious improvement to the peak to spectrum conversion process described above is the use the measured peak magnitude as the magnitude of the generated peak rather than a 1.0. Using the measured peak magnitude has the advantage of recoding the relative magnitudes of the peaks into the spectrum. It is unclear how the frequency envelopes of the peaks may encode information about their source bar but nonetheless it seems unwise to dispose of this information when forming the synthetic spectra.

As implemented here the peaks are interpolated as triangles with a parametric semitone width. The specification of the width in semitones is important because it ensures that the width of the peak will increase with increased center frequency. When viewed on a linear scale the semitone width specification results in asymmetric triangles - the low frequency leg is shorter than the high frequency leg. The top graph of Figure 2.4 shows a collection of triangles synthetically generated to represent the fundamental and fourth harmonic of each vibraphone note.

The bottom graph shows the result of finding the dot product of the spectrum of F#4 with each of the triangle templates

A second approach to measuring the distance between discrete peaks is to directly measure the distance between the two sets of peaks. The algorithm used to implement this approach is based on the intuition that the difference between
Figure 2.4: Triangle templates and euclidean distance for all templates.

two patterns of discrete points is small when the points line up close to one another and large when they are spread apart.

The most direct way to accomplish this measurement would be to associate all peaks in the prototype vector to the closest peak in the unknown vector and vice versa and then find the distance between matched peaks. Where v0 and v1 are vectors of frequency values this method can be accomplished in octave with the following code:

```octave
m0 = ones(length(v1),1) * v0;
m1 = v1' * ones(1,length(v0));
n0 = length(v0);
n1 = length(g1);
d = sum(min(abs(m0 - m1)))./ n0 + sum(min(abs(m0'-m1')))./n1;
```

This calculation produces the average distance between the best possible
match of all points.

This value is probably a good way to measure distance between well defined point vectors (like the template vectors) but may not be ideal for comparing template vectors to unknown vectors. In this system the unknown vectors will in general have far more peaks than the template vectors because the unknown signal will contain multiple simultaneous notes. The distance from unknown points to their randomly selected nearest template point could easily overwhelm the well matched points.

A better measurement would compare the template peaks to the unknown peaks but not the other way around. If the template vector were contained in v0 the last line of the octave code from above could then be reduced to:

\[ d = \frac{\text{sum}(\text{min}(\text{abs}(m0 - m1))))}{\text{length}(v0)} \]

In effect this means unknown peaks which are relatively far from the template peaks would be ignored.

The concept of treating near peaks differently than far peaks could be extended into a more complicated algorithm. By including a distance limit as a parameter the matching algorithm could attempt to measure the distance between near peaks and then add a penalty distance for peaks that were out of range.

This concept was implemented as an alternate to the simpler previous approach. The procedure begins by finding peaks in the template spectrum which are within some semitone range of peaks in the unknown spectrum. It then turns the problem around and finds all peaks in the unknown spectrum that are within range of peaks in the template spectrum. The intersection of these two lists provides a set of unique matches between the two spectra. All peaks in both sets can then be assigned to one of three categories.

**Matching Pairs** Matching pairs are associated sets of measured and template peaks which are within range of one another.

**Extra** Extra peaks are measured peaks with no template peak in range.
Missed Template peaks with no measured peaks in range.

To arrive at the distance between the two peak vectors the absolute value of the distance between the matching peaks is penalized by the number of extra and missed peaks. The maximum allowable distance between peaks is treated as the penalty value. This calculation can be summarized as:

\[ d = \frac{\text{abs}(h_{0} - h_{1}) \times 2 + (Mn \times md) + (En \times md)}{(H0n \times 2) + Mn + En} \times md \]

where:

- \( h_{0} \) and \( h_{1} \) are the frequency peaks on a semitone scale
- \( md \) - is the maximum allowable distance between peaks in semitones
- \( Mn \) - count of missing peaks
- \( En \) - count of extra peaks

The minimal possible value is zero when all peaks match exactly. The maximum possible value is 1.0 when no peaks match at all. In effect this formulation measures the average distance between matching peaks and assigns a penalty distance to all unmatched peaks. The penalty distance is the maximum frequency distance.

Maher and Beauchamp (1994) has outlined a similar approach to matching frequency peak vectors. In his 'two-way mismatch' algorithm he compares measured peaks to ideally positioned synthesized harmonic peaks in an effort to locate the fundamental. He also suggests weighting the peaks directly according to their height and frequency. Stronger peaks also have a better signal to noise ratio and are therefore more reliable than lower peaks. Higher frequency peaks have better fractional frequency resolution and therefore may produce more accurate distance measurements.

A similar approach to a weighted match score could be used for the system described here. As discussed above the window length used by the Fourier
transform dictates the possible resolution of the sinusoid components of the frequency peaks. The contribution from peaks occurring at frequencies below this threshold could be suppressed relative to peaks above it. Likewise the high overtones in the vibraphone spectrum have been observed to be unstable in frequency and so their contribution could also be reduced.

Yet another method of comparing peak frequency vectors would be to break the templates into individual peaks labelled with their source pitch. Each measured peak could then be assigned to its nearest pitch. The pitch with the most matches would then be picked as the most likely pitch. This method has the advantage of tremendous simplicity but eliminates the natural advantage gained by forcing patterns of unknown peaks to match to patterns of template peaks.

The ideal match would attempt to find a combination of templates which can account for the measured signal. In effect each measured peak should be assigned to a peak in a template. The minimal set of template combinations which best accounts for all the peaks could then be defined as the most likely set of sounding notes.

### 2.4 Template Representation

As with any instance based supervised learning system their are several options for preprocessing and representing the template feature vectors. Two of the most common issues center on data normalization and aggregation. Normalization is the process of removing some commonn feature in order to emphasize another. Aggregation is the process of combining templates in an effort to produce a more prototypical templates for each class.

**Normalization**

The magnitude spectrum values may optionally be normalized to remove the effect of the overall magnitude while retaining the shape of the spectrum. This is accomplished by dividing through by the sum of the all magnitude bins.
This operation in effect removes the effect of the force of the mallet strike from the templates. The operation is valid if the bar acts linearly with the force of the strike - that is the force of the strike does not effect the proportional mix of frequency components but rather scales them all equally. This assumption generally seems to hold as will be shown below.

The primary advantage of this normalization is that it produces magnitude spectrum based feature vectors with equal energy (they all sum to 1.0). Comparing the two vectors can then be accomplished by finding the Euclidean distance between the measured and template vector.

The problem with this kind normalization is that in the presence of noise it can significantly distort the shape of the spectrum. This is a particular problem when the spectrum is extracted from a signal with a low signal to noise ratio. When measuring the template spectra it is fairly easy to simply discard low energy templates. The unknown spectra however do not afford this possibility.

In general the distortion caused by the normalization takes the form of large areas of noise in the highest part of the spectrum (above 10k). This noise can easily hold as much energy as the lower frequency signal components. The simplest way to handle the problem is to limit the bandwidth of the compare function to frequencies below this limit. While this may improve the situation somewhat it will work at the expense of significantly limiting the available bandwidth used to represent high pitched notes. It is not unusual on the vibraphone to have clear overtones at well above 10k particularly if the bar is struck with a hard mallet. One possible solution is to zero bands relative to the highest identified peak. The noise peaks are not generally selected by the peak selection process because prior to normalization they are relatively small. This approach would adapt the bandwidth of the feature vector to the contents of the signal and therefore might prove to be a more flexible solution.

*Aggregation*

The simplest way to compare unknown feature vectors to template vectors
is to compare each unknown vector to each template vector. Another way would be to aggregate the template vectors according to their pitch and thereby reduce the number of templates. Decreasing the number of templates has the benefit of decreasing the computational cost of the compare. It may also improve the quality of the templates by averaging out noise from the instances - leaving a more prototypical template. Of course if different templates share the same pitch but have significant differences then aggregating might decrease the quality of the matches. For example aggregating templates recorded from different instruments might eliminate important features necessary to recognize the bar on one or both instruments.

Aggregating the magnitude spectrum feature vectors is very straightforward. The mean magnitude of each bin is taken across all examples for that pitch. How to aggregate the frequency peak vectors however is less obvious.

If the sound produced by a vibraphone were harmonic the problem of aggregating peak data would be much easier. In fact it might not even be necessary to use measured peaks at all. The location of the peaks could be predicted from their fundamental. Synthetic templates could then be generated which had peaks at the location of the fundamental. In fact this is the approach that many pitch trackers take to following harmonic signals ([Klapuri (2006)]).

As discussed above vibraphone bars vibrate in certain natural modes. Two prominent modes produce the fundamental and fourth harmonic. The modes above the fourth harmonic are significantly inharmonic and also appear to vary somewhat between instruments (See Figure 1.1). The following discussion illustrates two methods of attempting to locate the modes.

The black cross markings in Figure 2.5 mark the location of frequency peaks from twenty templates for F#4. Each of the twenty templates in turn has around twenty peaks. The blue trace is the average magnitude spectrum of the twenty instances. The red stars mark the location of the seventeen largest peaks in the spectrum (as determined by the 'magnitude only' peak selection algorithm
described above)

Figure 2.5: Mode selection using summed magnitude.

It is expected that the modes would fall on those frequencies with the greatest number of vertically aligned black crosses. By appearance the technique of choosing the modes based on the peaks in the averaged magnitude vector has identified these locations. From the graph alone however it is not clear that the modes were optimally chosen. For example there is a vertical collection of crosses at around 2500 Hertz which was not selected as a mode whereas an apparently smaller collection was selected at around 3000 Hertz.

The distribution of peaks is also an interesting feature. Some peaks are well clustered in frequency and others are more spread out, or even bi-modal. As expected the peak around the fundamental (at the leftmost red star) and the fourth harmonic are tightly clustered in frequency. The peak near 2250 Hertz however appears to be associated with two separate vertical clusters. Likewise several of the higher frequency peaks appear to have multiple clusters. These sub-clusters
are probably a result of combining templates measured from different instruments.

Figure 2.6 shows two views of an alternate method used to locate the modes of the F#4 bar. This method attempts to locate the modes by forming clusters directly from the position of the peaks as shown by the black cross markers in Figure 2.5. The blue cross markers in Figure 2.6 are at the location of the modes found in Figure 2.5.

The cluster method used a three step operation. First each peak (black cross in 2.5) was assigned a value based on the count of other peaks within 0.5 semitones of it. Second each of the counts in steps one is incremented by the inverse of the distance to all peaks within 0.5 semitones of it. The inverse distance is normalized to a value between 0.0 and 1.0. Peaks with many very close peaks are therefore incremented by about 1.0 and peaks with few or no nearby peaks are not incremented at all. This step has the result of making the peaks closest to the center of a cluster slightly higher than the surrounding peaks. The state of
the peaks after this step is shown in the top graph of Figure 2.6. In the final step each peak is examined and compared to all peaks within 0.5 semitones. The value of the peak under examination is assigned to the largest peak in the range. This has the effect of narrowing the mode peak to a single peak. In light of this last step the second step is also easier to understand; the second step makes two peaks in range of one another have equal heights highly unlikely - whereas it was quite likely after step one. The final state of the clustered peaks is shown in the bottom graph of Figure 2.6. The red stars show the location of the seventeen highest peaks and therefore the best guess modes for the process.

It is interesting to notice that despite the difference in approach both methods produce just about the same result. In most cases the blue crosses representing the position of the modes found in the first method are at the same frequency as the modes found by the clustering technique. The two exceptions are the mode at 7000 Hz in the first method was not repeated in the second method and the mode at around 3250 Hz in the second method was not found by the first method.

Perhaps the most notable insight Figure 2.6 gives however is that the search appears to be finding too many nodes. There seems to be a clear division in the magnitude of the peaks found via the second method. The highest ten modes appear to be much higher than the lowest seven modes. This may suggest that we should be looking for ten or less modes.

The top plot of Figure 2.7 shows ten peaks for each of 943 templates representing all bars. The spectrum used to generate the peaks was 16384 samples long and used a hann window. The sample rate of the template audio was 44100 and so the effective frequency resolution was 5.38 Hertz. The peak frequency has been divided through by the labelled fundamental frequency to shift the fundamental to near 1.0.

The bottom graph shows the clusters reduced using the algorithm described above. The ten largest peaks were found to be at ratios of: 3.79, 0.95,
As expected the fundamental and the fourth harmonic are most prominent. The next three harmonic at 13, 9.6 and 2 however also appear to be significant. The lack of particularly well defined clusters however suggest that they may not be useful for the purposes of recognition.

2.5 Onset Detection

In the system proposed here attack feature measurements are only taken when an onset is detected in the unknown signal. The onset detector is therefore a critical component of the attack identification sub-system. Like other percussive instruments the vibraphone is a fairly easy instrument to onset detect reliably. Nonetheless the onset detector must be well tuned to avoid temporal masking effects.
As part of this project several different onset detection schemes were tested. Figure 2.8 shows four different perspectives on a four note sequence where a loud note overlaps a quieter note. The four notes are F3, F#3, G3 and G#3.

![Graphs showing different perspectives on a four note sequence.](image)

**Figure 2.8: Measurements related to onset detection.**

The green vertical lines in the first three figures show the position of the true onset. In this case the signal was formed synthetically by overlapping samples of the four notes. The onset position could therefore be labelled with high precision.

The black continuous line in all four graphs shows the envelope of the signal. The envelope was formed by taking the RMS of the samples in 64 sample blocks. The audio signal has a sample rate of 44100 samples per second. Where a Fourier transform is used for the frequency domain processes discussed here a 512 sample hann analysis window was used.

The blue trace on Figure 2.8 graph A shows the RMS envelope of the signal converted to a decibel scale then normalized into the 0.0 to 1.0 range. The normalization is intended to allow simple visual comparison between the shape of
the linear envelope and the decibel envelope. The decibel plot gives a better sense of what a listener hears. Notice that there is very little indication of the second note on the decibel scale. This is roughly equivalent to the actual sound. The second note can be heard but is quiet enough to go unnoticed on casual listening.

Figure 2.8 graph B shows the plot of three common audio features used to summarize the spectral content of a signal. HFC (red trace) refers to high frequency content which is defined as.

\[ HFC = \sum_{k=0}^{N-1} |X_k| \]  

(2.1)

where \( X_k \) is the discrete Fourier transform of \( x(n) \) evaluated at \( n_0 \) with a window length of \( N \)

\[ X_k = \frac{1}{N} \sum_{n=0}^{N-1} x(n_0 + n)w(n)e^{-j2\pi kn/N} \quad k = 0, 1 \ldots N - 1 \]  

(2.2)

\( w(n) \) represents an analysis window function.

SFM (green trace) refers to spectral flatness measure. This audio feature captures the flatness versus spikiness of the spectrum. Attack transients are noisy and therefore tend to produce a flat spectrum. Sustained sound however tend to be composed of discrete overtones and therefore to produce spiky spectrums. Spectral flatness measure is defined as the geometric mean of the power spectrum divided by the arithmetic mean.

\[ SFM = \sqrt[\frac{N}{2}]\prod_{k=0}^{N-1} \frac{|X_k|}{N} \]  

(2.3)

ZCR is the zero crossing rate. Zero crossing count is a simple time domain measurement of the number of times the signal crosses zero during a given time window. Noisy sounds tend to have many zero crossing while harmonic sounds
tend to have fewer zero crossings. We would therefore expect the attack transients to generate an increased zero-crossing rate.

All three of these signals were computed using a sliding window of 512 samples and a hop distance of 64 samples. Of the three signals the HFC clearly responds best to the second note onset. In fact however the HFC appears to be only about as useful as the envelope itself. In other words the change in the HFC value is about the same as the change in the RMS envelope in the vicinity of the second onset. As an onset detection function this makes the two about equivalently effective.

The SFM responds very noticeably to the onset of the initial loud note (F3 and G3) but is much less responsive to the second quite note.

The ZCR is too noisy to be useful. While there is increase in the zero crossing count around the onsets the increase is not enough to find a detection threshold which would not produce many spurious onsets.

In the example shown here the quiet note is separated from the preceding louder note by 50 milliseconds. As can be seen in the RMS signal this produces a clear bump in the envelope which alone would be noticed by an onset detector. If the inter-onset time of the two notes was halved to about 25 milliseconds however it is easy to see that the envelopes would begin to merge such that the bump in overall energy would be more difficult to detect. The change in energy would then be small enough that it would not be enough to indicate an onset relative to other small changes in the overall level which do not indicate onsets. The bump in energy following the fourth note would be an example of an apparent increase in energy which is not associated with of a note onset.

If the change in energy is insufficient to trigger the onset some research has indicated that phase deviation may be a useful alternative (Bello (2004)). The principal behind this technique is that the phase of the sinusoidal elements of a complex signal should change predictably while the phase of noisy signals should deviate randomly.
The method used to measure phase deviation using the phase measurements from a Fourier transform is as follows. Each bin of the Fourier transform is assumed to reference an oscillator operating at the bin center frequency. This reference oscillator is useful because it is possible to unambiguously track its unwrapped phase. A constant frequency sinusoidal component in the signal under analysis will produce a phase measurement which can be compared to the unwrapped reference phase. Given that the component frequency is constant the difference between the reference phase and the measured phase should be the same at each hop. A transient in the signal however will create broadband energy in the spectrum and upset the steady state component phase thereby changing the deviation from the center frequency. The summed phase deviation across the spectrum should therefore reflect the presence of a transient in the signal.

The phase measurements from the Fourier transform are only valid when energy exists in the bin under analysis. It has therefore been suggested that the phase deviation should be weighted by the amplitude of each bin. (Dixon (2006)). The blue trace (Phase 0) in Figure 2.8 graph C shows phase deviation with amplitude weighting. In practice however this value appears to reflect nothing more than white noise shaped by the amplitude envelope. Using phase deviation of just spectral magnitude peaks produced a similar plot.

The reason for the phase deviation proving ineffective is not entirely clear. Tests with sustained and frequency modulated signals produced the expected results but real audio tended to produce very noisy transient signals which were not suitable for simple threshold detection. The presence of multiple sinusoids per band might contribute to the problem. Perhaps only the higher bins should be examined since they are less likely to contain multiple components. With the signal shown here it also seems possible that the time required for the phase deviation to stabilize may not have elapsed.

Lacoste (2007) found similar disappointing results using the phase deviation method and suggested using the sum of the difference of phase across the
spectra. Although he offered no explanation for this approach it appears to produce somewhat better results than the phase deviation approach. See the red trace in Figure 2.8 graph C (Phase 1). The second and fourth note show spikes at their onset using this measure. Perhaps this behavior can be explained by noting that the bins on either side of a sinusoid tend to track the phase of the sinusoid. Spectrums consisting of mostly sines would therefore tend to have a lower difference between successive bins. In contrast noisy spectrums would have a random phase distribution which would tend to produce a larger phase differences between adjoining bins.

Figure 2.8 graph D shows the onset detector used by the attack identification system. This detector uses a combination of first order difference of a constant-Q transform and high frequency content. First a constant-Q transform is taken by frequency warping the magnitude spectrum of the signal using a base octave of 16.35 Hertz. Since the bin width is actually wider than this the lowest bands are assigned their own bin until the octave bands are wider than the bin width. This results in a 10 band spectrum. The difference between successive constant Q frames (hops) is taken and the high frequency content technique is then applied to each band (the band magnitude is scaled by the band number). High frequencies are therefore emphasized over low frequencies. The detection function resulting from this process is shown in as the magenta trace (detect) in Figure 2.8 graph D. The large spike at the onsets are easy to detect with a threshold of 1.0. The blue circles are the point where the ascending detection function crosses the threshold and indicate the point of the report onset.

2.6 Template Generator

Both the attack identification sub-system and the tone tracking sub-system use their own instance of a template generator for controlling the template representation.
The raw data used by the template generator is a list of hand segmented vibraphone recordings. In this case the recordings consist of 1020 individual vibraphone notes recorded from four different instruments. The number of strikes is actually somewhat less than this because in some cases the strikes were simultaneously recorded with two microphones at different positions. The strikes vary in intensity and mallet type. The toolbox functions thSegmentReview() and thAnnote() were used to create a label file containing the pitch, mallet type (hard/soft), and dynamic level of the note. The dynamic level was taken as the average RMS level of the note between its onset and offset. The signal offset is taken as the location where the RMS envelope had fallen to less than -30 dB (0 dB being full scale). The location of the onset as determined by the onset detector was also recorded as a means of aligning the attack identification template analysis window.

The template generator amounts to a function for producing templates with different representations. The most fundamental parameters to the template generator control the three basic properties of the template list: template format (magnitude spectra or peak), method of template aggregation (or no aggregation at all), and the length and location of the analysis window in the segment audio.

The format of the templates (magnitude spectra vs. peaks) was discussed in detail above. When a client of the template generator requests a particular format it also provides the parameters for measuring that format. For example when the peak format is requested then the minimum peak level and the minimum peak distances is also specified. The same is true when aggregation is requested.

Control of the size and location of the template analysis window within the template audio is important for generating templates that are specific to either the attack identification system or tone tracking system. The attack identification system needs to measure the template audio just after the onset is detected - which is around the location of the maximum amplitude of the note. The tone tracker uses a much longer analysis window positioned well past the attack. This distinction
is in keeping with the way the unknown signal will also be measured. The attack
id sub-system only measures the unknown signal when an onset is detected using
a short window to capture the spectral shape of the attack. In contrast the tone
tracking system uses a longer window for following the longer term components of
the note with better frequency resolution.

2.7 Evaluating the system

Figure 2.9 shows the visual evaluation tool. This interface is used to
inspect the results of tracking a labelled audio file with a given system configura-
tion. The orientation of the graph shows notes on the vertical axis and time in
milliseconds on the horizontal axis. The example shown here is a series of over-
lapping semitones beginning on F3. While somewhat obscured by other graphic
tokens the labelled notes are displayed as black horizontal lines - piano role style.
The parallel horizontal lines on the upper part of the graph show the position of
the labelled notes transposed by octaves. The actual notes are labelled with their
pitch midway through their duration. The purpose of the octave transpositions is
to clearly indicate the position of octave errors. This type of error is so common
that it is useful to be able to clearly visualize it.

The red, blue and green circle tokens represent the tone trackers calcu-
lation of the most likely active notes. In the example shown in Figure 2.9 the red
and blue markers track the labelled notes fairly well. In some places, particularly
at the point of attacks, however tone tracker tokens can be seen which are well off
the mark. These tokens represent tracking errors.

Onset errors are shown as vertical lines running from the top of the graph
to the bottom. Missed onsets are shown in red and spurious onsets are shown in
cyan or yellow. The onset evaluation system has a programmable time window
which it uses to evaluate onset errors. Labelled onsets with no measured onset
within the time window are considered misses. Measured onsets with no labelled
Table 2.2: Template Generator Parameters.

<table>
<thead>
<tr>
<th>Name</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>wndMs</td>
<td>milliseconds</td>
<td>Analysis window length</td>
</tr>
<tr>
<td>wndOffsMs</td>
<td>milliseconds</td>
<td>Analysis window offset from template onset</td>
</tr>
<tr>
<td>wndCenterFl</td>
<td>boolean</td>
<td>Center window on onset vs. start at onset</td>
</tr>
<tr>
<td>wndType</td>
<td></td>
<td>Analysis window type</td>
</tr>
<tr>
<td>minHz</td>
<td>Hertz</td>
<td>Minimum spectrum or peak frequency</td>
</tr>
<tr>
<td>maxHz</td>
<td>Hertz</td>
<td>Maximum spectrum or peak frequency</td>
</tr>
<tr>
<td>minDST</td>
<td>semitones</td>
<td>Minimum distance between peaks</td>
</tr>
<tr>
<td>peakThreshDb</td>
<td>decibels</td>
<td>Minimum peak power</td>
</tr>
<tr>
<td>smoothST</td>
<td>semitones</td>
<td>Moving avg window width for whitening</td>
</tr>
<tr>
<td>srcType</td>
<td></td>
<td>Feature data source</td>
</tr>
<tr>
<td></td>
<td>synthesize</td>
<td>Sythesize harmonic peaks</td>
</tr>
<tr>
<td></td>
<td>peaks</td>
<td>Use measured peaks</td>
</tr>
<tr>
<td></td>
<td>spectrum</td>
<td>Use magnitude spectrum</td>
</tr>
<tr>
<td>shapeType</td>
<td></td>
<td>Shape for peaks to spectrum conversion</td>
</tr>
<tr>
<td></td>
<td>off</td>
<td>Use peak with no shape</td>
</tr>
<tr>
<td></td>
<td>triangle</td>
<td>Triangle shape (see triWidthST)</td>
</tr>
<tr>
<td></td>
<td>sinc</td>
<td>Sinc shape</td>
</tr>
<tr>
<td></td>
<td>hann</td>
<td>Hann window shape</td>
</tr>
<tr>
<td>triWidthST</td>
<td>semitones</td>
<td>Half width of triangles</td>
</tr>
<tr>
<td>whitenType</td>
<td></td>
<td>Apply whitening to the spectrum</td>
</tr>
<tr>
<td></td>
<td>off</td>
<td>Do not apply whitening</td>
</tr>
<tr>
<td></td>
<td>linear</td>
<td>Apply whitening to the linear spectrum</td>
</tr>
<tr>
<td></td>
<td>db</td>
<td>Apply whitening to the db spectrum</td>
</tr>
<tr>
<td>Aggregate</td>
<td></td>
<td>Aggregate the spectrum</td>
</tr>
<tr>
<td></td>
<td>off</td>
<td>Do not aggregate. Use each instance</td>
</tr>
<tr>
<td></td>
<td>magnitude</td>
<td>Cluster using magnitude of all instances</td>
</tr>
<tr>
<td></td>
<td>frequency</td>
<td>Cluster using frequency distance</td>
</tr>
<tr>
<td>Normalize</td>
<td>boolean</td>
<td>Normalize the spectrum to unit energy</td>
</tr>
</tbody>
</table>
onset within the time window are considered spurious onsets and displayed in cyan. Likewise if multiple measured onsets map to one labelled onset only the closest onset is considered correct - all others are considered spurious and displayed in yellow.

Each measured onset (correct or spurious) has an associated set of most likely attack classifications. As shown in Figure 2.9 these classifications are displayed as black, cyan and yellow stars. The black star is the most likely, followed by cyan and yellow.

Each evaluation run also generates a quantitative report listing the number of correct and erroneous classifications. The listing below shows the report associated with Figure 2.9. For the pitch classifications it breaks the errors out into three categories: octave errors, semitone errors and all other errors. For the onset classifications the errors are broken down between missed, extra and multiple.

Table 2.3 shows an example of the report generated for the same run of
Table 2.3: Onset evaluation report.

<table>
<thead>
<tr>
<th>Onset</th>
<th>Labels</th>
<th>Found</th>
<th>Correct</th>
<th>Missed</th>
<th>Extra</th>
<th>Mult</th>
</tr>
</thead>
<tbody>
<tr>
<td>37</td>
<td>39</td>
<td>36</td>
<td>92.3%</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>5.4%</td>
<td>7.7%</td>
<td>2.6%</td>
<td></td>
</tr>
</tbody>
</table>

the system depicted in Figure 2.9. The labels correspond to the errors described above in the onset detector system. The percentage values for Correct, Extra and Mult are taken as a function of the total number of onsets found. The percentage Missed is taken as a function of the number of labelled onsets.

Table 2.3 shows examples of reports generated for a series of system test. The system was run on the same audio file shown in Figure 2.9. The srcType variable for the attack identifier and tone tracker template generator was changed for each run of the system. The top several lines show the statistics for the best run. For this set of tests the attack identifier and tone tracker were set to analysis window lengths of 1024 and 4096 respectively.

These results suggest that peak information may be more effective than magnitude information for tracking both the attack and sustain portion of the vibraphone notes. More testing on a wider variety of input and with a manipulation of a greater number of variables however will be necessary before coming to any strong conclusion.

2.8 Future Work

The system as proposed here has several fundamental limitations. First the it is limited by the poor frequency resolution of the Fourier Transform at low frequencies. This is particularly problematic for attack identification when the window length must also be short.

Use of an alternative time to frequency transform in the range below the effective note resolution is one possible solution to this problem. In this scenario the signal would be split into bands above and below some cross-over point. The
Table 2.4: System Statistics.

<table>
<thead>
<tr>
<th>Labels</th>
<th>Correct</th>
<th>Missed</th>
<th>ST</th>
<th>Octave</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peak-Peak: Attack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tone</td>
<td>49</td>
<td>24</td>
<td>25</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>49.0%</td>
<td>51.0%</td>
<td>4.0%</td>
<td>12.0%</td>
</tr>
<tr>
<td>Tone</td>
<td>699</td>
<td>513</td>
<td>186</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>73.4%</td>
<td>26.6%</td>
<td>0.5%</td>
<td>14.5%</td>
</tr>
<tr>
<td>Magn-Magn: Attack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tone</td>
<td>49</td>
<td>20</td>
<td>29</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>40.8%</td>
<td>59.2%</td>
<td>6.9%</td>
<td>6.9%</td>
</tr>
<tr>
<td>Tone</td>
<td>699</td>
<td>488</td>
<td>211</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>69.8%</td>
<td>30.2%</td>
<td>0.5%</td>
<td>12.8%</td>
</tr>
<tr>
<td>Peak - Magn: Attack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tone</td>
<td>49</td>
<td>22</td>
<td>27</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>44.9%</td>
<td>55.1%</td>
<td>7.41%</td>
<td>7.41%</td>
</tr>
<tr>
<td>Tone</td>
<td>699</td>
<td>494</td>
<td>205</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>70.7%</td>
<td>29.3%</td>
<td>0%</td>
<td>23.9%</td>
</tr>
<tr>
<td>Magn - Peak: Attack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tone</td>
<td>49</td>
<td>21</td>
<td>28</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>42.9%</td>
<td>57.1%</td>
<td>0%</td>
<td>10.7%</td>
</tr>
<tr>
<td>Tone</td>
<td>699</td>
<td>483</td>
<td>216</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>69.1%</td>
<td>30.9%</td>
<td>1.9%</td>
<td>8.8%</td>
</tr>
<tr>
<td>Synth - Peak: Attack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tone</td>
<td>49</td>
<td>9</td>
<td>40</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>18.4%</td>
<td>81.6%</td>
<td>7.5%</td>
<td>12.5%</td>
</tr>
<tr>
<td>Tone</td>
<td>699</td>
<td>229</td>
<td>470</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>32.8%</td>
<td>67.24%</td>
<td>6.6%</td>
<td>11.7%</td>
</tr>
<tr>
<td>Synth - Magn: Attack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tone</td>
<td>49</td>
<td>8</td>
<td>41</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>16.3%</td>
<td>83.7%</td>
<td>7.3%</td>
<td>14.6%</td>
</tr>
<tr>
<td>Tone</td>
<td>699</td>
<td>346</td>
<td>353</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>49.5%</td>
<td>50.6%</td>
<td>0.5%</td>
<td>24.7%</td>
</tr>
</tbody>
</table>
high frequency band would be transformed by the Fourier transform - as it is now. The low frequency band could be transformed, for example, by linear predictive coding. Figure 2.10 shows an example of a spectrums generated by LPC and a Fourier transform on the same 512 sample vibraphone attack.

![Figure 2.10: 30 pole linear predictive coding on a 512 sample window.](image)

A more complex solution would replace the Fourier transform altogether. Boyer (2002) has described an analysis system based on damped sinusoid decomposition. Vibraphones tones are collections of damped sinusoids and so might be effectively analyzed by a technique like this. The use of a damped sine analysis technique also suggests taking better advantage of time signature information. The current system is entirely frame based. It makes no use of successive frames in attempting to recognize ongoing notes.

Preliminary results for the current system suggest that peak data is more effective than magnitude data for separating simultaneous notes (See Table 2.4). Peak information however takes a relatively long time to form after the note onset.
A system which recognized the note by its attack spectrum would be faster. The current system may not be matching attack spectrums well because the composite spectrum is distorted by the spectral mixtures which are simultaneously sounding. The individual template spectrums can therefore not match it effectively. One possible solution for this problem would be to use Non-negative matrix factorization (Smaragdis and Brown (2003)) to attempt to draw apart the individual spectrums.

2.9 Octave Toolbox

The following functions serve as good entry points into the octave toolbox. The web pages used to document the source code contain 'calls' and 'called by' links to easily navigate to the functions used by these top level routines.

When executed these three program are driven interactively through a keystroke interface. Type 'h' at the program prompt for a list of possible commands.

thSegmentReview is the main program used to segment and label the template data.

thAnnotate is the program used for verifying new algorithms.

vibStFtTrkTest is the program used to test the system.

2.10 Conclusion

This paper has discussed several basic signal measurement techniques relevant for real-time instrument tracking. In particular issues related to the limitations of the Fourier transform and Fourier peak selection were discussed and a variety of solutions suggested. Several methods for selecting peaks in the frequency domain were detailed including magnitude and phase based methods. Spectral whitening as a peak picking pre-process was described. Several onset detection
schemes were evaluated and a constant-Q based high-frequency-content algorithm was arrived at. The proposed system was coded in octave and a preliminary evaluation system was tested.
References


