Technical University of Crete
School of Electrical and Computer Engineering

Time and Frequency Estimation in Guitar Signals

by:
Tazes Athanasios

Supervisor:
Associate Professor Georgios Karystinos

Thesis Committe:
Associate Professor Michail Lagoudakis
Associate Professor Aikaterini Mania

31 January 2019

Abstract

In this thesis, a complete note onset estimation and pitch detection system is presented, tuned for guitar music signals. The goal of this system is to first estimate the point in time that a single note was played and then attribute a frequency to it. This task is then extended to a sequence of played notes. Various onset and pitch estimation algorithms were tested and compared, taking into account both their efficiency to complete the task, accurate transciption of the signal, as well as their computational complexity timewise, thus achieving processing time less than or equal to the duration of the music signal.

Contents

List of Figures

1 Introduction

The goal of this thesis is to perform time and frequency estimation on guitar signals. This translates to finding the points in time that notes wree played in the signal, and then finding which notes were played. The whole system was designed under the constraint that the processing time is less than or equal to the duration of the music signal.

In Chapter 2, the mechanics of generating a sound through a string will be studied, followed by a short analysis of the ways a musical signal can be represented in the Time and Frequency field.

In Chapter 3 some frequently used techniques on sound proccessing will be described, individually, so that they can be viewed as "black boxes" for the rest of the thesis.

- Frequecy Filtering : Attenuation, amplification or complete removal of subsets of frequencies.
- Short Time Fourier Transform : A simple alteration of the Fourier Transform which allows monitoring of the spectral changes of the signal over time.
- Linear Least Squared : A popular technique used to fit a mathematical linear model to a set of real data.

In Chapter 4 the complete system is presented, which performs the following tasks.

- Preprocessing of the signal, using techniques from Chapter 3.
- Onset Detection : Creation of the Onset Detection Function, followed by Peak-Picking the correct onset times.
- Pitch Detection : Estimation of the fundamental frequency of picked onsets, followed by an octave correction method.

Lastly, in Chapter 5 the results of the system applied on a guitar signal I recored are shown, followed by possible future work and improvements.

2 The Guitar Signal

2.1 Physics of the instrument

A guitar string is a string fixed at both ends which is elastic and can vibrate. These vibrations are called standing waves, and they satisfy the relationship between wavelength and frequency that comes from the definition of waves:

$$v = f\lambda$$

where v is the speed of the wave, f is it's frequency and λ is the wavelength.

Once the string is plucked, it will freely vibrate at it's fundamental frequency, which is a function of the tension of the string. The fundamental frequency is only one of the ones that coexist at the string, and is actually the lowest one in frequency. Any wave with wavelength that satisfies the wave equation, with nodes at it's end can exist at the string.

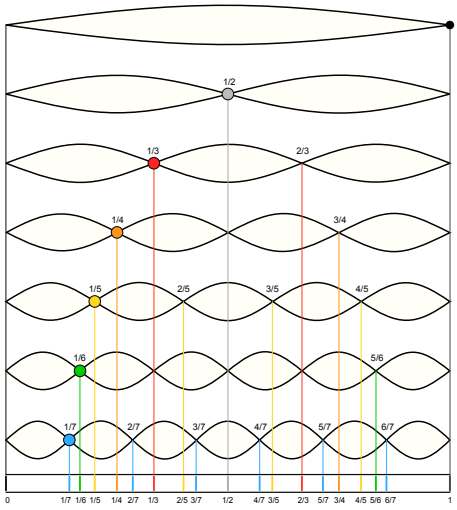


Figure 2.1: String Overtones [1]

Figure 2.1 shows some of the coexisting harmonics on a guitar string. The first wave, the fundamental, comes from the string vibrating with one big arc from bottom to top, satisfying the condition $I = \lambda/2$ where I is the length of the freely vibrating string, and λ the wavelength. The first harmonic comes from vibration with a node in the center wich satisfies $I = \lambda$ and each higher frequency wave will fit an additional half wavelength on the string, satisfying

the conditions $I = {}^{3\lambda}/_{2}$, $I = {}^{4\lambda}/_{2}$ etc, with a theoretical infinite number of harmonic waves.

These waves coexists on the string, and the magnitude of each harmonic, as well as the ratios between them follow no pattern. The content of ratios between them will change between instruments, strings, even the same string plucked at a different way. This seemingly random content of harmonics, along with the timbre of the instrument, is what gives an instrument it's characteristic sound. Timbre is the perception of the sound, depending to the way it was created, such as through strings, percussion, choir etc. The sum of these waves constitutes a musical note, which are most commonly are represented in a music sheet.

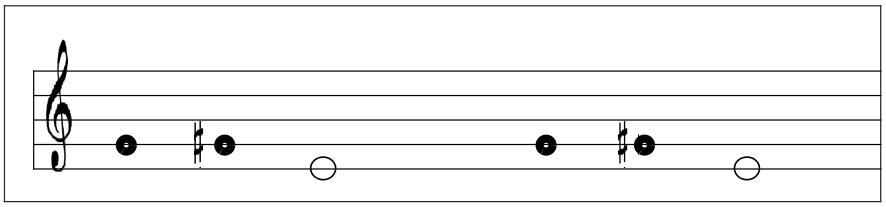


Figure 2.2: Music Sheet

2.2 Signal Representations

2.2.1 Real-Time Representation

The guitar signal can be viewed as an amplitude-varying function of time in the real-time field. Figure 2.3 shows the temporal evolution of a few notes, along with an ideal segmentation of one of them.

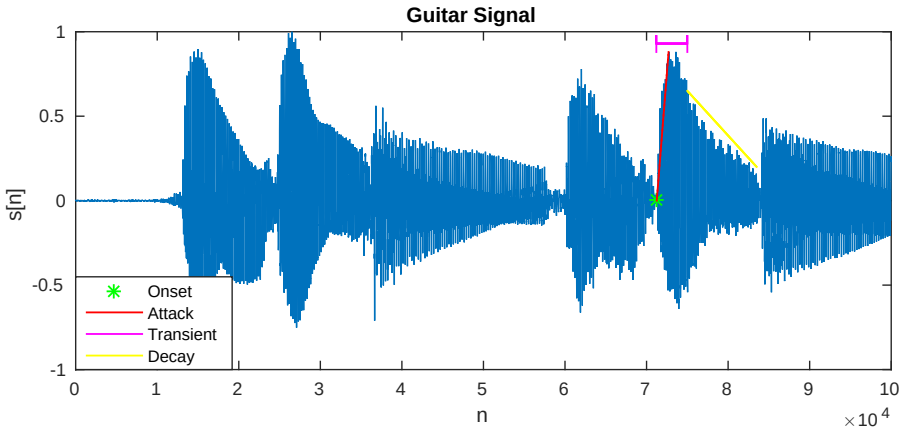


Figure 2.3: Music Signal in real time

This is the most common representation of a music signal. In the current state of music signals processing, this will most probably going to be a digital signal, sampled at 44.1kHz.

$$s[n] = \begin{bmatrix} s_1 & s_2 & s_3 & \dots & s_n \end{bmatrix}, f_s = 44.1\text{kHz}$$

At this point it is important to precisely define the terms onset, attack, transient and decay, since they lack precise definition, mostly due to the fact that they change in sense according to the needs of the applications. For the rest of this thesis, these terms will be defined as:

- Onset: The moment chosen to mark the beginning of a musical event, i.e. the pluck of a guitar string.
- Attack: The time interval during which the amplitude of the signal is still increasing without having reached it's maximum value.
- Transient: The most defining characteristic of a music signal. A short interval of time beginning with the onset and ending in an arbitrary

manner, during which the signal behaves in a temporally quick, unpredictable way.

- Decay: The part of the signal starting after the transient. It's end can be defined either as the point after which the energy of the music signal is comparable to the energy of the noise, or more realistically as the time interval during the previous note hardly being audible and the next one being played.

2.2.2 Frequency Field Representation

Studying a music signal in the field of frequencies is a very common practice. Since the harmonics of notes exist and specific and expected frequencies, a lot of information can be easily extracted by studying them in this field, which can be accessed by the Fourier Transformation.

The Fourier Transform decomposes any function into a sum of sinusoidal basis functions. Each of these basis functions is a complex exponential of a different frequency. The Fourier Transform therefore offers a unique way of viewing any function - as the sum of simple sinusoids, and provides the defining characteristics of these sinusoids, amplitude and phase at the corresponding frequencies. It is the extension of the Fourier Series theory, which will not be analysed further, on non-periodic signals, under the assumption that the non-periodic signal being analysed is actually periodic outside the window of observation.

Music signals fall well inside this category, since their assumed period is hardly stable after the first few repetitions of the note, due to decay of the force pulsing the string, as well as the undefinable state of the transient of the note.

The mathematical expression for Fourier Transform is

$$S(f) = \mathcal{F}\{s[n]\} = \sum_{n=0}^{N-1} s[n]e^{-j2\pi nk/N}$$

where $s[.]$ is the original signal, N the number of samples, $S[.]$ the transformed signal.

A musical note is a summation of harmonic waves, which means individual sinusoidals of different phases and magnitude at different frequencies.

Therefore the Fourier Transform decomposition offers directly the information of each harmonic individually, making it an ideal tool for extracting information from musical notes.

3 Tools

3.1 Frequency Filtering

3.1.1 High-Pass Filtering

A very usefull practice when processing music signals, is high-passing the lowest frequency content out of it.

The lowest harmonic that is expected to be present on a guitar signal is the fundamental of the bassiest note. This frequency is 82Hz, the fundamental of e2 note. Any spectral component bellow that is just noise, so it can be filtered out.

To perform High-Pass on a signal:

1. Perform FFT of the signal to get it's power spectrum, with fft-points next power of two of the samples of the signal for time efficiency.
2. Creating the frequencies vector corresponding the this N-fft point transform.
3. Locate the index of the frequency corresponding the the cut-off frequency.
4. Creating a new signal with completely cut-off frequencies below the cut-off, and the same as the original above.
5. Create a conjugate symmetric vector of amplitudes needed for inverse transform.
6. Perform inverse FFT.

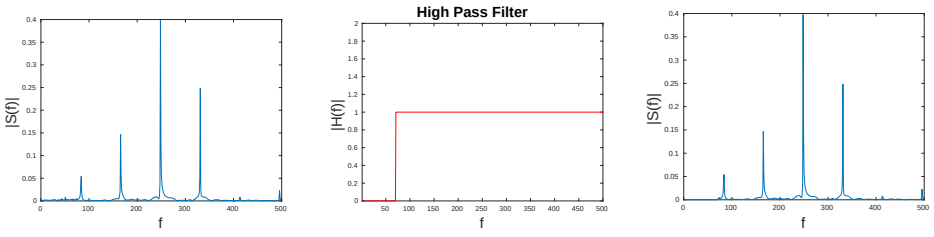


Figure 3.1: Power Spectrum Before/After high-passing.

Figure 3.1 shows the power spectrum of a signal before and after it was high-passed at a cut-off frequency of 70.

3.1.2 Low-Pass Filtering

Low-Pass filter is the complimentary filter to High-Pass. It works the same way, with the difference that it removes the frequencies above a picked frequency.

It was mentioned earlier that the digital sound signals are most likely going to be sampled at 44.1kHz. The reason behind this is that human hearing spectrum is capped at approximately 22kHz. So according to Nyquist Theorem, the sampling frequency must be the double of the maximum audible frequency. Regardless of that, a big portion of these frequencies are practically useless for the signal generated by guitar strings. As was explained in Section 2, each note has a fundamental frequency, as well as the harmonics at integer multiples of it. A guitar will cover about four octaves at average, and the highest fundamental frequency of such is going to be at the 1.3kHz area. Also, even though theoretically the harmonics are infinite, the spectral component of them will rapidly decrease. This results is no more than six or seven harmonics having an affecting spectral component, or be audible at all. That makes any frequencies above 9kHz, as stated, practically useless, therefore they are completely removed.

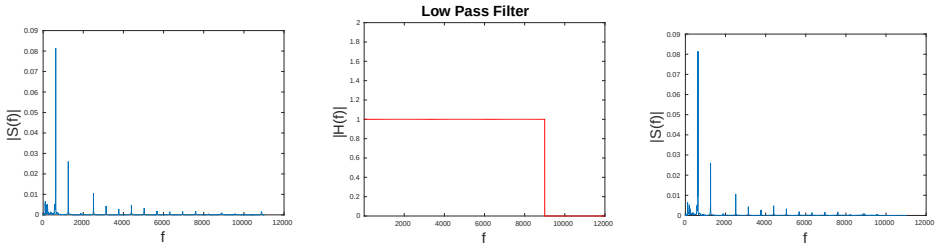


Figure 3.2: Power Spectrum Before/After low-passing.

Figure 3.2 shows the power spectrum of a signal before and after it was low-passed at a cut-off frequency of 9kHz.

3.2 Short Time Fourier Transform

The properties of Fourier Transform were discussed in Section 2 along with how it's well fitted for use on music signals. While it is a very suitable tool for processing segments of the original signal, especially for pitch detection, it hardly offers any usefull information for the overall signal. Music signals are heavily time-varying, corresponding to the notes played at specific times, and knowing the spectral characteristics of it in overall doesn't really help any aspect of the transcription problem. It does give an intuition of the notes played throughout the signal, but neither the time or frequency information on the individual notes. We are interested in the time-varying spectral characteristics of the signal, not the overall.

The Short Time Fourier Transform, STFT from now on, is a simple alteration of the Fourier Transform that provides this option. STFT is a time-frequency representation, essentially it divides a longer time signal into shorter segments of equal length, overlapping or not, and then compute the Fourier Transform separately on each segment. This way, the magnitude and phase content of local sections of a signal can be monitored as they changes over time.

STFT of s[n] is given by

$$S_k(m) = \sum_{n=0}^{\infty} s[n]w(mh - n)e^{(-j2\pi nk)/N}$$

where k = 0,1...N-1 is the frequency bin index, N the number of fft points, w(n) a finite-length sliding window of observation and h the hop size between consecutive transforms, and is shown in the following figure.

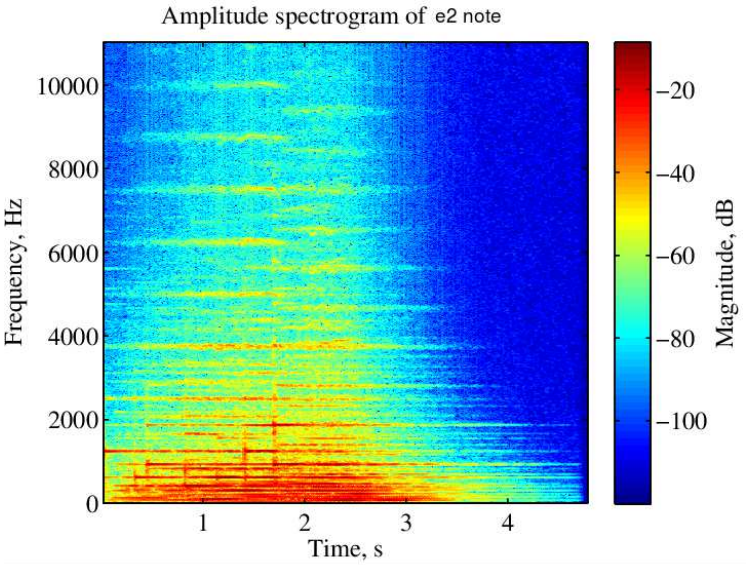


Figure 3.3: STFT performed on a music signal.

Figure 3.3 shows the STFT of a music signal. The frequency content corresponding to time, along with the colour describing the magnitude of the signal in these frequencies.

The implementation of STFT is quite simple and the challenge of it is picking the optimal parametres. Time and frequency information are inversely proportional, since picking a small window of observation will give better time resolution but worse frequency resolution, and vice versa, for a fixed computational capacity. This reflects heavily on the transcription algorithm, since the two main issues to solve, onset detection and pitch detection require good time resolution and good frequency resolution respectively. A visual interpretation of this trade-off can be simply viewed as:

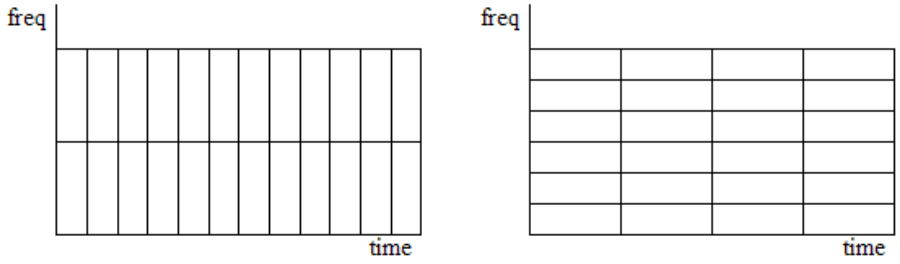


Figure 3.4: STFT Trade-off

Figure 3.4 shows the difference in time/frequency resoultion of same computantional capacity, varying with the length of observation window.

The steps to perform STFT are:

1. Choose appropriate window of observation length and hop size. In this case, a window of 256 samples window with hop size 128 is used, meaning overlapping consecutive windows of transforms.

2. Perform FFT on the first window, then repeat for every window per hop.

3. Update STFT matrix's column per consecutive transform.

The result of this is a matrix where each column contains the Fourier Transform of a 5.8msec segment of the signal.

3.3 Linear least squared

Linear Least Squares, LS from now on, is another popular technique, used to fit a mathematical or statistical, linear model to a set of real data, as shown in the figure below.

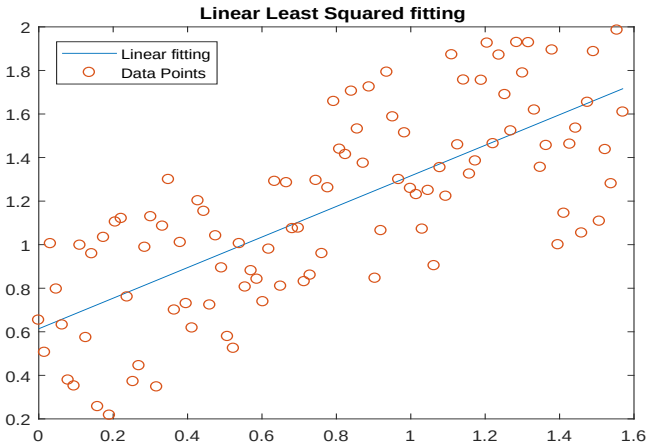


Figure 3.5: Linear Least Squares

As the name implies, this technique finds the line which most accurately fits the blue data points by minimizing the overall square of the distances of the points to the line. The distance between a data point and the linear model can be either perpendicular of vertical, but in practice the vertical is always minimized.

From a mathematical standpoint, if $y = ax+b$ describes the linear model, y_i corresponds to the y coordinates of the data, and R^2 is the 2-D Cartesian Field , LS minimizes the equation

$$R^2(a,b) = \sum_{i=1}^{n} [y_i - (a + bx_i)]^2$$

This translates to finding the conditions for which the derivative of the value we seek to minimize becomes zero, in reference to the variables of our model, i.e, a,b, which in turn translates to sovling the following system of equations

$$\frac{\delta R^2}{\delta a} = 0 \Rightarrow -2 \sum_{i=1}^{n} [y_i - (a + bx_i)] = 0$$

$$\frac{\delta R^2}{\delta b_i} = 0 \Rightarrow -2 \sum_{i=1}^{n} [y_i - (a + bx_i)] x_i = 0$$

which is easily solved into

$$na + b\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$

$$a\sum_{i=1}^{n} x_i + b\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i$$

and into a single array problem

$$\left[\begin{array}{cc} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{array}\right] \times \left[\begin{array}{c} a \\ b \end{array}\right] = \left[\begin{array}{c} \sum y_i \\ \sum x_i y_i \end{array}\right] \Rightarrow \left[\begin{array}{c} a \\ b \end{array}\right] = \left[\begin{array}{cc} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{array}\right]^{-1} \times \left[\begin{array}{c} \sum y_i \\ \sum x_i y_i \end{array}\right]$$

which in turns gives up the optimal variables a and b for the linear model.

$$a = \frac{\sum x_i y_i - \frac{\sum x_i \sum y_i}{n}}{\sum x_i^2 - \frac{\sum x_i^2}{n}} \ , \quad b = \sum y_i - \frac{a\sum x_i}{n}$$

It is worth mentioning that although this technique is mostly used to fit linear models into non linear phenomena, in this case it will be used to extract the "slope function" of another function.

4 System

4.1 Onset Detection

The Onset Detection subproblem has 2 phases. First a Detection Function is derived from proccessing of the input signal. This function's desired property is to have high values at the points in time an onset occurs, while low on the rest. After this function is formulated, the next task is to correctly pick the points in time that correspond to onsets.

4.1.1 Preprocessing for Onset Detection

To create the Detection Function, some preprocessing is performed. Specifically the signal is high-passed and low-passed, as described in the previous section. The low cut frequency picked is 70Hz and the high cut at 9kHz. Afterwards STFT is performed on the music signal. It was noted earlier that the important aspect of performing STFT is the choice of parameters. At this point, STFT is going to segment a 44.1Khz sampled signal into segments of samples of our choice. So this will directly impact the localisation of the onsets to be detected, or, in overall, the accuracy of the system. In this case, 256 sample segments are picked. This allows for a localisation accuracy of approximately 6msecs on a 44.1kHz sampled signal. Also, the hop size of consecutive transforms is picked at 128 samples, half the window size. This means that there is a w/2 overlapse of samples between consecutive transforms, allowing even better localisation of the events

Matrix **s** is defined, where each column holds 256 samples of the original signal s[n] with an overlap of 128 samples. Then Fourier Transform is performed on each column resulting in matrix **S**.

$$\mathbf{s} = \begin{bmatrix} s_1 & s_{129} & s_{257} & \dots & s_{n-w+1} \\ s_2 & s_{130} & s_{258} & \dots & s_{n-w+2} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ s_{256} & s_{384} & s_{512} & \dots & s_n \end{bmatrix}$$

$$\downarrow \mathcal{F}$$

$$\mathbf{S} = \begin{bmatrix} S_{11} & S_{12} & S_{13} & \dots & S_{1q} \\ S_{21} & S_{22} & S_{23} & \dots & S_{2q} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ S_{p1} & S_{p2} & S_{p3} & \dots & S_{pq} \end{bmatrix} \downarrow \text{Frequency}$$
$$\longrightarrow$$
$$\text{Time}$$

Each element of matrix $\mathbf{S}$ is a complex number describing the sinusoidal at frequency k for frame of time m.

4.1.2 Onset Detection Function

Once the signal is ready for processing, the next goal is to create the Detection Function. The algorithm proposed makes use of both spectral and phase content of the signal. Onsets are always going to create a burst in the local spectral characteristics of the signal, but since guitar onsets are being detected, it is not going to be enough. It's a guitar's desired property to have good sound sustain, which translates to rich spectral component throughout the signal. That is why the algorithm is also going to take advantage of the phase discontinuations that occur during the transient part of the signal.

I. Spectral Component

Having performed STFT, the local spectral characteristics of the whole signal can be monitored. By taking the difference between two consecutive segment's magnitude, the Spectral Difference Function can be defined.[2]

$$\delta S_k(m) = |S_k(m)| - |S_k(m-1)| \Rightarrow$$
$$\delta\mathbf{S} = \begin{bmatrix} |S_{11}| - 0 & |S_{12}| - |S_{11}| & \dots & |S_{1q}| - |S_{1,q-1}| \\ |S_{21}| - 0 & |S_{22}| - |S_{21}| & \dots & |S_{2q}| - |S_{2,q-1}| \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ |S_{p1}| - 0 & |S_{p2}| - |S_{p1}| & \dots & |S_{pq}| - |S_{p,q-1}| \end{bmatrix}$$

This function is a measure of spectral changes in the signal, and since they are directly assossiated with presence of an onset, it constitutes a detection

function by itself, but not an efficient one. The following figure shows the Spectral Difference function of a music signal at an arbitrary frequency.

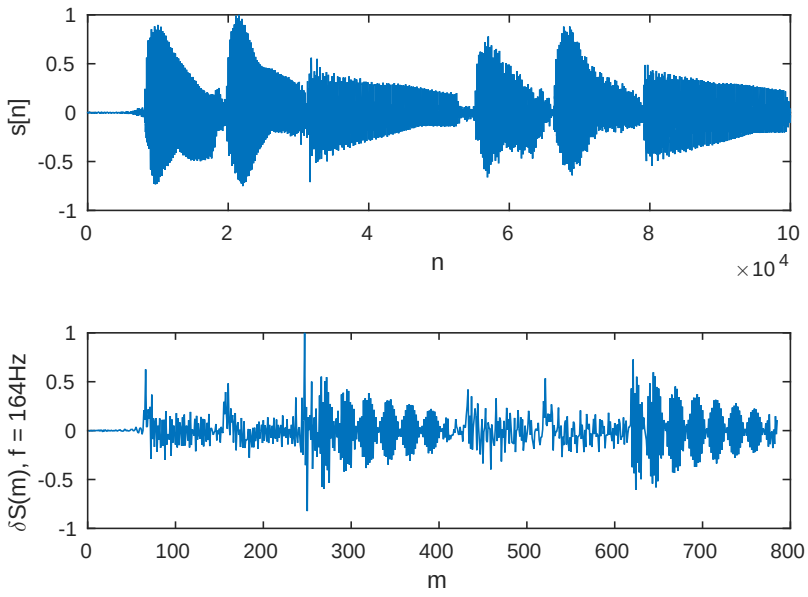


Figure 4.1: Spectral Difference - Guitar

It is noticeable that the moments of attacks on the music signal are loosely related to peaks of Spectral Difference function, but still in no way resembles a well defined, and well observable detection function.

Spectral Difference alone is inadequate to perform onset detection, since the guitar signal has high spectral component throughout the duration of a note.

II.Phase Component

The STFT also provides the phase offset of the sinusoidals of which the signal consists.

$$\mathbf{\Phi} = \angle\underline{\boldsymbol{S}} = \begin{bmatrix} \angle\underline{S}_{11} & \angle\underline{S}_{12} & \dots & \angle\underline{S}_{1m} \\ \angle\underline{S}_{21} & \angle\underline{S}_{22} & \dots & \angle\underline{S}_{2m} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \angle\underline{S}_{k1} & \angle\underline{S}_{k2} & \dots & \angle\underline{S}_{km} \end{bmatrix} = \begin{bmatrix} \Phi_{11} & \Phi_{12} & \dots & \Phi_{1m} \\ \Phi_{21} & \Phi_{22} & \dots & \Phi_{2m} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \Phi_{k1} & \Phi_{k2} & \dots & \Phi_{km} \end{bmatrix}$$

This is the wrapped phase, meaning that all phase points are constrained to the range $-\pi <$ Phase Offset $< \pi$ radians, according to their offset. When the actual phase is outside this range, the phase value is increased or decreased by a multiple of 2π radians to put the phase value within this range. This can be seen at the following figure.

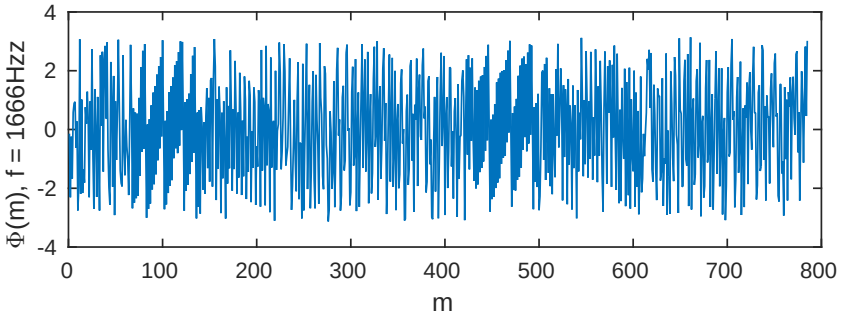


Figure 4.2: Wrapped Phase

The algorithm makes use of a transformation of wrapped phase, the unwrapping of it. Unwrapping the phase means that instead of increasing/decreasing by 2π in order to fit in a defined range, the phase jumps are instead increased over a constant, π in the case. The following figure shows this transformation, alongside the original guitar signal.

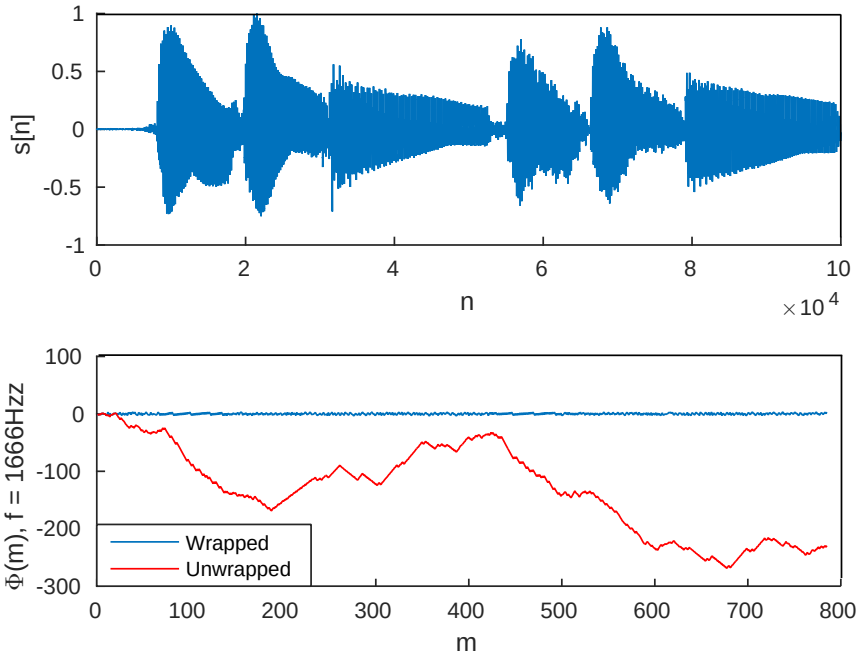


Figure 4.3: Wrapped/Unwrapped Phase

During the steady state of a guitar note, or whenever the are no events occuring in the signal, the unwrapped phase difference between consecutive sets of frames is expected to be close to equal, as expressed by the equation below.[3]

$$\Phi(k,m) - \Phi(k,m-1) \simeq \Phi(k,m-1) - \Phi(k,m-2)$$

By moving all the terms on one side of the equation, Phase Deviation function is defined as

$$\delta\Phi(k,m) = \Phi(k,m) - 2\Phi(k,m-1) + \Phi(k,m-2)$$

which for steady state parts of the signal is close to zero.

$$\delta\Phi(k,m) \simeq 0 \Rightarrow$$

$$\Phi(k,m) \simeq 2\Phi(k,m-1) - \Phi(k,m-2)$$

This suggests that the phase of a sinusoidal can be calculated from the phases of the 2 previous frames, given that the signal is on steady-state.

This function monitors the phase discontinuations, by taking for each frame into consideration the previous two as well. On contrary to being close to zero during steady parts of the signal, this function is going to take large values during the frames of the transient part of the note, since the sinusoidals creating it are by no means steady during that period. The randomness of transient's behaviour is going to create considerable differences in the expected phase of consecutive frames of the same sinusoidal.

This way the Phase Deviation matrix is defined.

$$\delta\mathbf{\Phi} = \begin{bmatrix} \Phi_{11} & \Phi_{12} & \Phi_{13} & \dots & \Phi_{1m-1} \\ \Phi_{21} & \Phi_{22} & \Phi_{23} & \dots & \Phi_{2m-1} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \Phi_{k1} & \Phi_{k2} & \Phi_{k3} & \dots & \Phi_{km-1} \end{bmatrix} - 2 \begin{bmatrix} 0 & \Phi_{11} & \Phi_{12} & \dots & \Phi_{1m-1} \\ 0 & \Phi_{21} & \Phi_{22} & \dots & \Phi_{2m-1} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \Phi_{k1} & \Phi_{k2} & \dots & \Phi_{km-1} \end{bmatrix} + \begin{bmatrix} 0 & 0 & \Phi_{11} & \dots & \Phi_{1m-2} \\ 0 & 0 & \Phi_{21} & \dots & \Phi_{2m-2} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \Phi_{k1} & \dots & \Phi_{km-2} \end{bmatrix}$$

Matrix $\delta\mathbf{\Phi}$ calculates the deviation between the expected phase of a sinusoidal based on previous frames and the actual measurement, given the signal is on steady-state.

III. Combined Approach.[3]

During the steady state segments of the signal, the magnitude of the frequencies is expected to stay approximately constant, and the phase, while not constant is of expected value. By monitoring magnitude and phase simultaneously we can quantify the stationarity of the signal, and localize the disruptions of it during the unexpectedness of the onset transient.

Assuming polar form, the value of the k^{th} bin of the STFT is predicted to be

$$\hat{S}(k,m) = |S(k,m-1)|e^{j\hat{\Phi}(k,m)}$$

where

$$\hat{\Phi}(k,m) = 2\Phi(k,m-1) - \Phi(k,m-2)$$

and in matrix form:

$$\hat{\mathbf{S}} = \begin{bmatrix} |\hat{S}(1,1)|e^{j\hat{\Phi}(1,1)} & \dots & |\hat{S}(1,q)|e^{j\hat{\Phi}(1,q)} \\ \vdots & \vdots & \vdots \\ \vdots & |\hat{S}(k,m)|e^{j\hat{\Phi}(k,m)} & \vdots \\ \vdots & \vdots & \vdots \\ |\hat{S}(p,1)|e^{j\hat{\Phi}(p,1)} & \dots & |\hat{S}(p,q)|e^{j\hat{\Phi}(p,q)} \end{bmatrix}$$

So, $\hat{S}_k(m)$ is the predicted state of the k^{th} bin for if there are no transients present, taking into consideration the expected stability of the magnitude, and the estimation of the phase according to the previous frames.

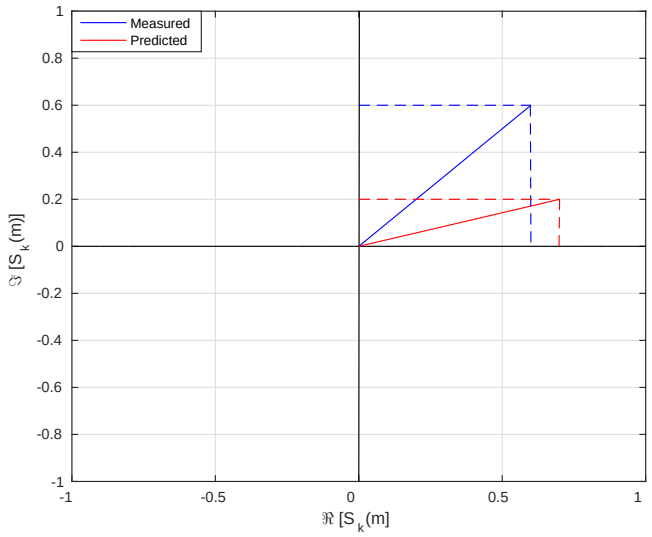
We may then consider the measured values from $\mathbf{S}$ which in polar form are

$$S_k(m) = |S_k(m)|e^{j\Phi_k(m)}$$

where $|S_k(m)|$ and $\Phi_k(m)$ correspond to the magnitude and phase of current frame.

$$\mathbf{S} = \begin{bmatrix} |S(1,1)|e^{j\Phi(1,1)} & \dots & |S(1,q)|e^{j\Phi(1,q)} \\ \vdots & \vdots & \vdots \\ \vdots & |S(k,m)|e^{j\Phi(k,m)} & \vdots \\ \vdots & \vdots & \vdots \\ |S(p,1)|e^{j\Phi(p,1)} & \dots & |S(p,q)|e^{j\Phi(p,q)} \end{bmatrix}$$

The predicted and measured values are shown in the next figure.



The difference between $\hat{S}_k(m)$ and $S_k(m)$ is the stationarity deviation of the signal, and the best way to measure it is the Euclidean Distance between them, as defined by

$$\Gamma(k,m) = \left|\hat{S}(k,m) - S(k,m)\right|$$

$$= \left||S(k,m-1)|e^{j\hat{\Phi}(k,m)} - |S(k,m)|e^{j\Phi(k,m)}\right|$$

$$= \left||S(k,m-1)|\cos(\hat{\Phi}(k,m)) + |S(k,m-1)|j\sin(\hat{\Phi}(k,m)) - |S(k,m)|\cos(\Phi(k,m)) - |S(k,m)|j\sin(\Phi(k,m))\right|$$

$$= \sqrt{[|S(k,m-1)|\cos(\hat{\Phi}(k,m)) + |S(k,m-1)|j\sin(\hat{\Phi}(k,m))]^2 - [|S(k,m)|\cos(\Phi(k,m)) - |S(k,m)|j\sin(\Phi(k,m))]^2}$$

$$= \sqrt{|S(k,m-1)|^2 - 2|S(k,m-1)||S(k,m)|(\cos(\hat{\Phi}(k,m))\cos(\Phi(k,m)) + \sin(\hat{\Phi}(k,m))\sin(\Phi(k,m))) + |S(k,m)|^2}$$

$$= \sqrt{|S(k,m-1)|^2 - 2|S(k,m-1)||S(k,m)|\cos(\hat{\Phi}(k,m) - \Phi(k,m)) + |S(k,m)|^2}$$

$$= \sqrt{|S(k,m-1)|^2 - 2|S(k,m-1)||S(k,m)|\cos(\delta\Phi(k,m)) + |S(k,m)|^2}$$

In case of $\delta\Phi_k(m) = 0$

$$\Gamma(k,m) = \sqrt{|S(k,m-1)|^2 - 2|S(k,m-1)||S(k,m)| + |S(k,m)|^2} \Rightarrow$$

$$\Gamma(k,m) = \big||S(k,m-1)| - |S(k,m)| = |\delta S(k,m)|$$

which means that when the phase prediction is good, then the spectral difference alone is taken into account.

Once the Euclidean Distances have been calculated, the Onset Detection Function as summation of Γ_k across frequencies k.

$$\Gamma(m) = \sum_{k=1}^{p} \Gamma(k,m)$$

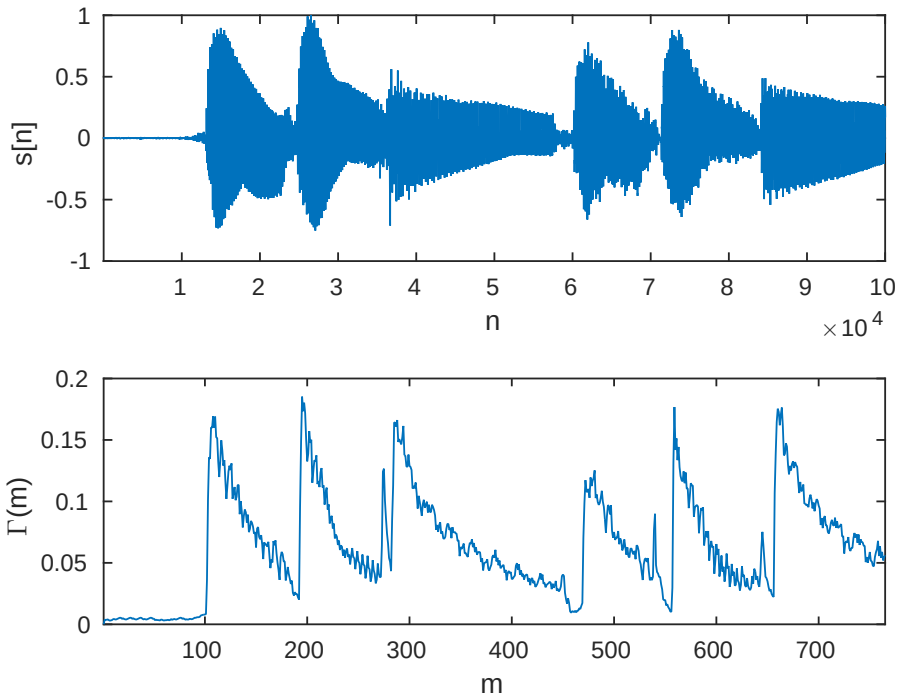
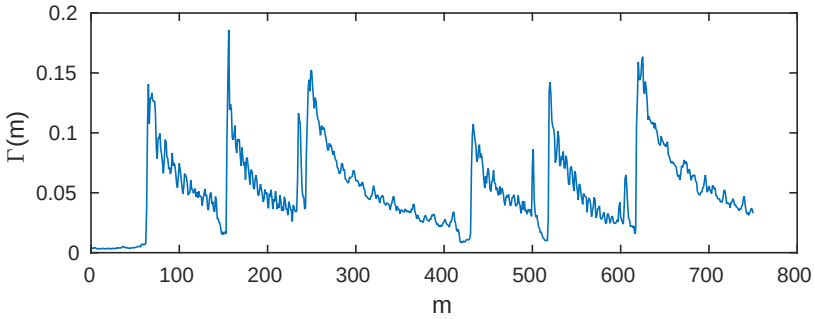


Figure 4.4: Detection Function

4.1.3 Peak-Picking

Once the detection function $\Gamma(m)$ has been formed, the next step is to decide which values of it correspond to onsets. Apparently, that translates to picking the peaks of the detection function. To do so effectively, some processing of the Detection Function will be performed.

First a median filter is applied to the detection function.



$$\Gamma(m) \rightarrow \text{median}(\mathbf{\Gamma}(m : m+5)), m \in \{1, 2.....q\}$$

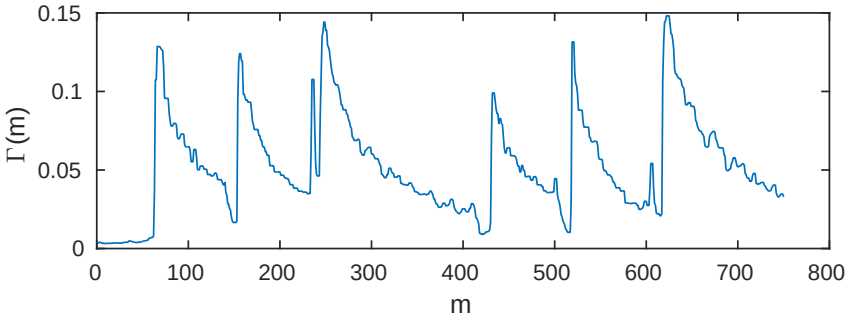


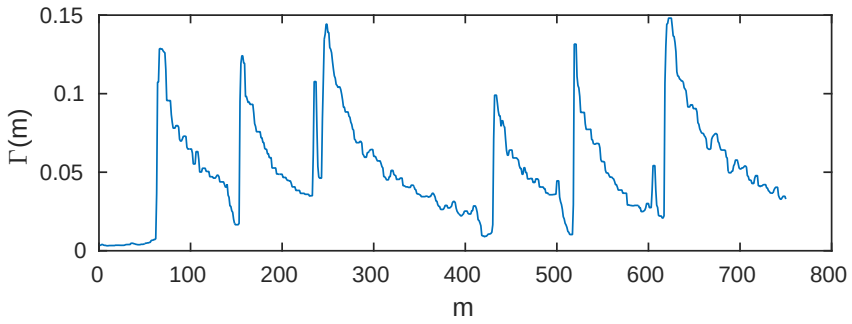
Figure 4.5: Median Filtering

Figure 4.5 shows the detection function before (up) and after (bottom) median filtering.

resulting in a smoothed Detection Function. The length of the median filter is 5 frames, approximately 30msec.

Afterwards, LS which was described in Section 2 is going to be applied in the smoothed Detection Function on length of 7 frames, approximately 40msec.

The offset factor is ignored, and the slope of the regressed line is used as the new Detection Function.



$$\Gamma(m) \rightarrow \mathrm{LS}(\mathbf{\Gamma}(m : m + 6)), m \in \{1, 2.....q\}$$

$$\Gamma(m) = b(m)$$

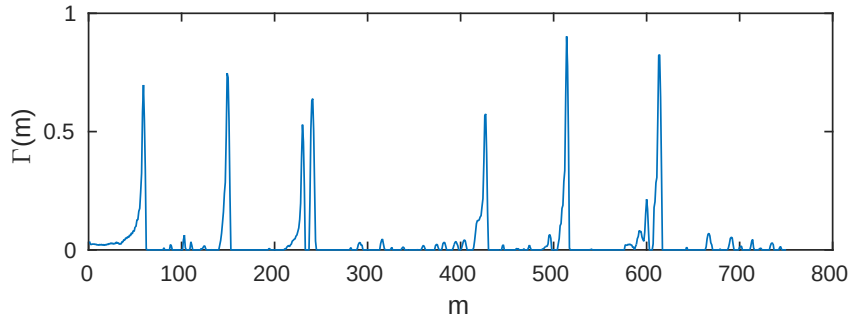
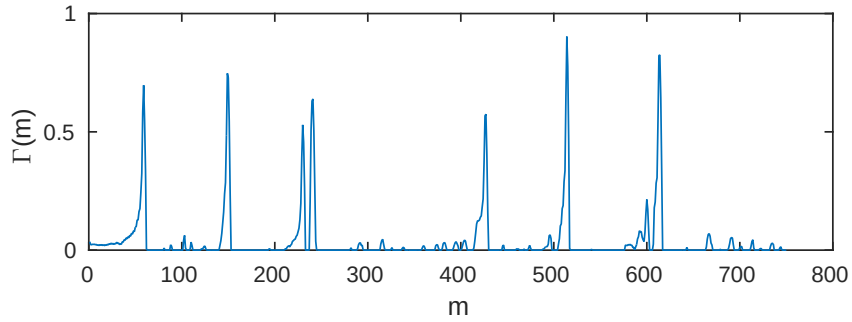


Figure 4.6: Slope of Detection Function

Figure 4.6 shows the smoothed Detection Function before (up) and after (bottom) applying LS.

Comparing it the original Detection Function $\Gamma(m)$, it's obvious that the peaks of real onsets are heavily attenuated while the rest of the function is close to zero or noise level.

The next step is to apply a small threshold to the function, removing the low values and keeping only the highest among the ones in close temporal positions.



$$\Gamma(m) = \begin{cases} \Gamma(m) & \text{if } \Gamma(m) > \text{threshold} \\ 0 & \text{if } \Gamma(m) \leq \text{threshold} \end{cases}$$

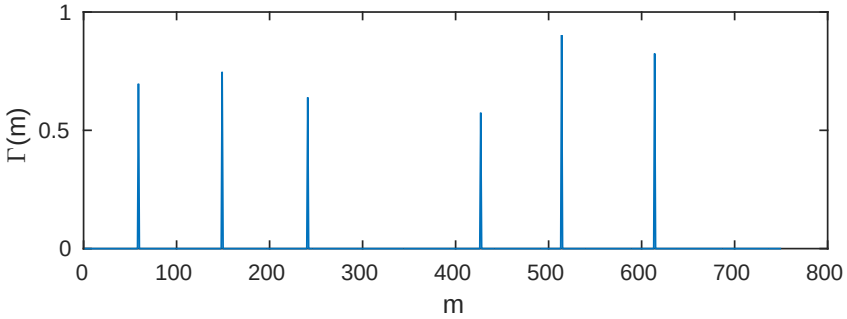


Figure 4.7: Picked Peaks

Figure 4.7 shows the Detection Function before (up) and after (bottom) applying the threshold.

The resulting function has values different than 0 at the indexes of frames that are decided to be onsets, and 0 on the rest. Afterwards, the number of frames get transformed back to their corresponding samples, resulting at knowing exactly the time at which the notes were played.

The result of peak picking can be seen in the following figure

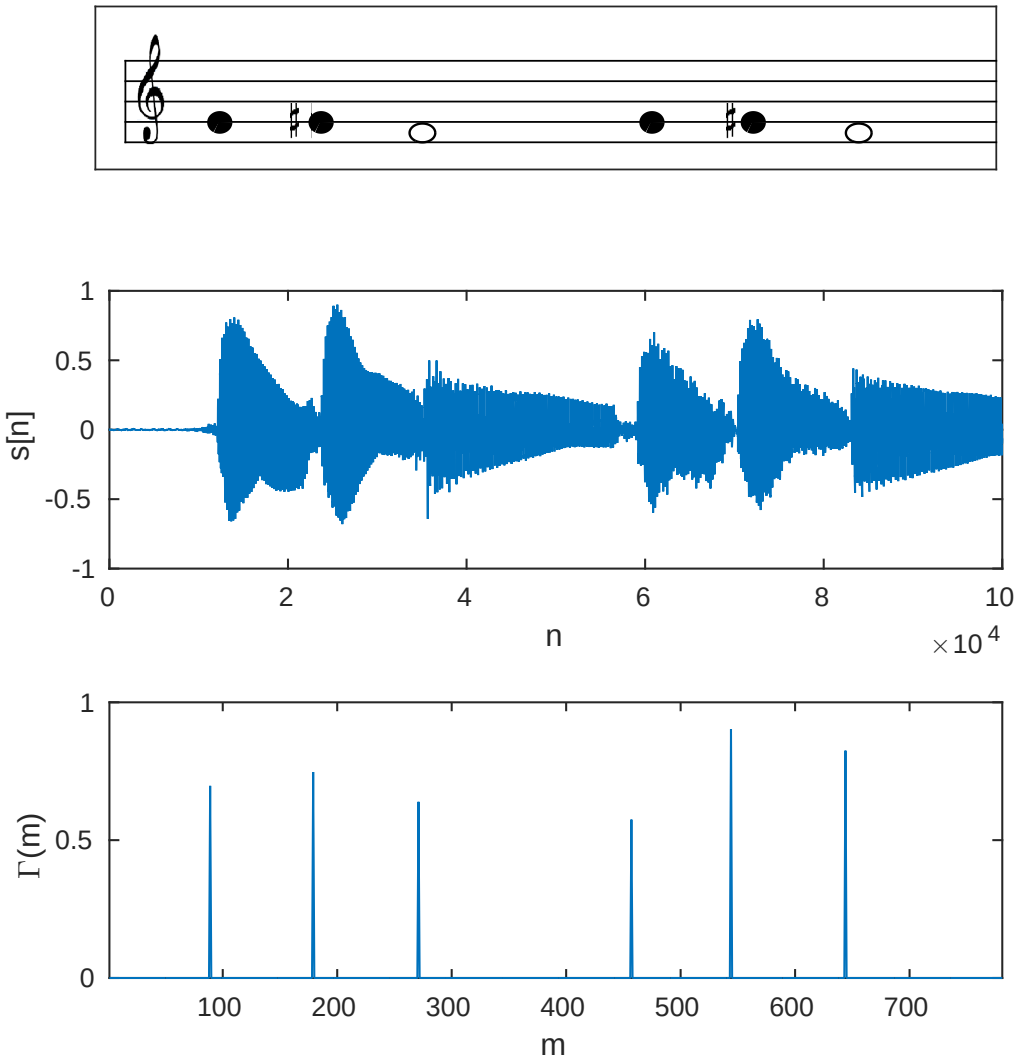


Figure 4.8: Picked Peaks

Figure 4.8 shows the original signal (above) and the picked peaks decided to be onsets(bottom).

4.2 Pitch Detection

Once the points in time where a note was played, i.e an onset, have been estimated, the next task is to attribute a frequency to them, to define which note was played.

4.2.1 Pitch Detection

For each onset picked, a vector containing $fs/10 = 4410$ samples is created.

$$\forall i \in \text{onsets}, \quad \mathbf{x} = \mathbf{s}[i : i + fs/10]$$

While a guitar note will most probably longer, a segment of 4410 samples is enough for calculating the fundamental frequency. The pick of this length also decides how many notes/second can be effectively identified, in this case 10notes/second.

Autocorrelation function, ACF from now on, takes an input function, $x(t)$, and cross-correlates it with itself; that is each element is multiplied by a shifted version of $x(t)$, and the results summed to get a single autocorrelation value. [4]

$$r_{xx}(\tau) = \sum x[j]x[j + \tau]$$

A guitar note is not a periodic signal, but strong periodicity is expected around the fundamental frequency of the note.

The problem that arises with ACF is it's computational difficulty. This function has to compare itself with all the delayed versions of itself for different values of τ, for every onset. Although ACF is a time domain method, frequency transformation can be used for computational efficiency, tackling the above problem.

According to the Wiener-Khinchin theorem, the ACF of a signal can be defined as the inverse Fourier Transform of the Power Spectral Density as

$$r_{xx}(\tau) = \sum_{\text{PSD}}(f)e^{j2\pi\tau f} \tag{1}$$

$$\text{PSD}(f) = \mathcal{F}\{x[n]\}\mathcal{F}^*\{x[n]\} = |X^2(f)| \tag{2}$$

$$(1) \xrightarrow{(2)} r_{xx}(\tau) = \sum_{f=1}^{fs/20} |X^2(f)| e^{j2\pi\tau f}$$

which suggests that the autocorrelation of the signal is the inverse Fourier Transform of the signal's power spectrum.

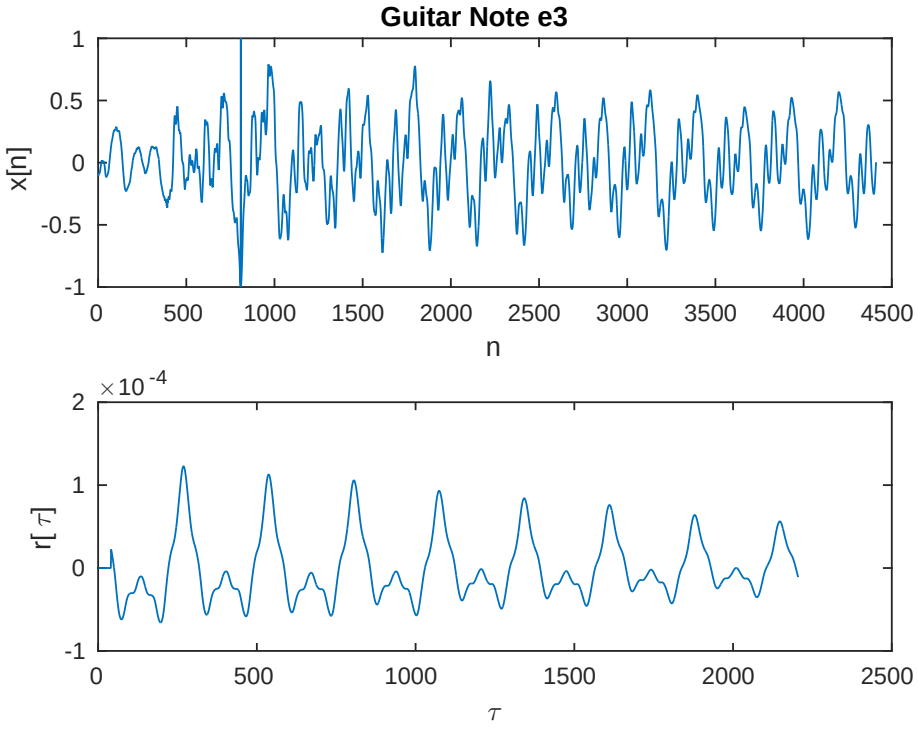


Figure 4.9: ACF of a guitar note.

Figure 4.9 shows 10msec of a e3 note (above)and it's autocorrelation(below). The ACF takes maximum value at for $\tau = 268$, meaning that the fundamental of the signal needs that number of samples to complete it's first period. By dividing the sampling rate fs with this number of samples, the fundamental frequency is calculated, which in this case is 164Hz, the fundamental of e3.

$$f_f = fs/\text{argmax}\{r_{xx}\}$$

4.2.2 Pitch Refinement

While ACF is a very reliable way to calculate the fundamental frequency, there are cases where it may fail, such as bad localisation of an onset or, in rare cases, stronger periodicity observed around a harmonic instead of the fundamental. While it's very unlikely that a wrong note will be decided, there is the possibility of picking wrong octave of a correct note. To tackle this problem, a Pitch Refinement algorithm is applied on every onset.

First an array of 36 prerecorded notes is created.

$$\mathbf{y} = \begin{bmatrix} y_{11} & y_{12} & \dots & y_{1,36} \\ y_{21} & y_{22} & \dots & y_{2,36} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ y_{4410,1} & y_{4410,2} & \dots & y_{4410,36} \end{bmatrix}$$

Each column is of $\mathbf{y}$ holds 0.1sec of a guitar note. Then Fourier Transform is applied to each of those notes, resulting at a Spectral Envelop $\mathbf{Y}$

$$\mathbf{Y} = |\mathcal{F}\{\mathbf{y}\}| = \begin{bmatrix} Y_{11} & Y_{12} & \dots & Y_{1,36} \\ Y_{21} & Y_{22} & \dots & Y_{2,36} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ Y_{4410,1} & Y_{4410,2} & \dots & Y_{4410,36} \end{bmatrix}$$

Afterwards, Fourier Transform is performed for each onset and compared with the transforms of the prerecorded notes, each being a column on $\mathbf{S}$.

$$\forall x, \quad X = |\mathcal{F}\{x\}|$$
$$\mathbf{Y}_{\text{diff}}(:, n) = |\mathbf{Y}(:, n) - X|, n \in 1, 2....36$$

When compared with the correct note's spectrum, the spectral component of $\mathbf{Y}_{\mathbf{diff}}$ is expected to be minimum, while having a rich component in the rest cases. This can be seen in the following figures.

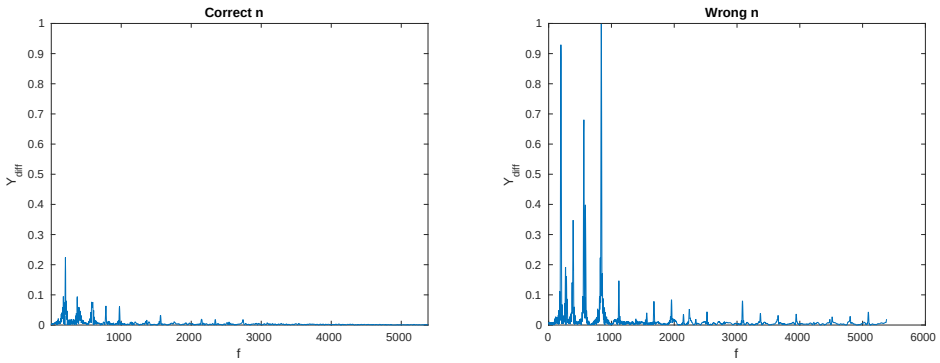


Figure 4.10: Spectral Envelop comparison

So, the correct match is choosen to be

$$\mathbf{Y}_{\text{result}} = \min\{\sum_{k \in f_n} \mathbf{Y}_{\text{diff}}(k, :)\}$$

The summation is performed on a subset of frequencies, f_n instead of the whole spectrum. f_n is the subset of frequencies where note harmonics are expected to appear and is calculated as

$$f_n = f_0 a^n$$

where f_0 is a base frequency, a is the $12 - th$ root of 2, and n the number of semitones away from f_0. The result can be seen in the following figure.

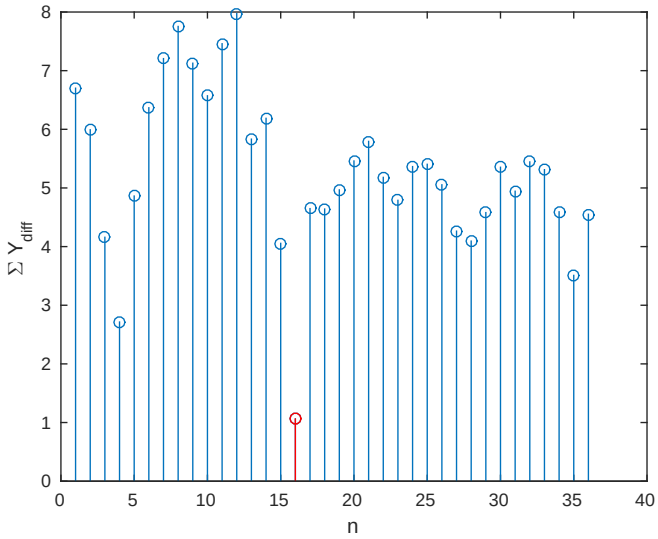


Figure 4.11: Spectral Envelop Result

At this point, 2 possible results have been calculated for each onset, so a decision rule must be applied. ACF is in overall more reliable, so it is going to be the decisive metric, while Spectral Envelop comparison is more reliable in the treble octave.

Octaves are defined as
octave1 : $80\text{Hz} < f < 160\text{Hz}$
octave2 : $161\text{Hz} < f < 320\text{Hz}$
octave3 : $321\text{Hz} < f < 640\text{Hz}$

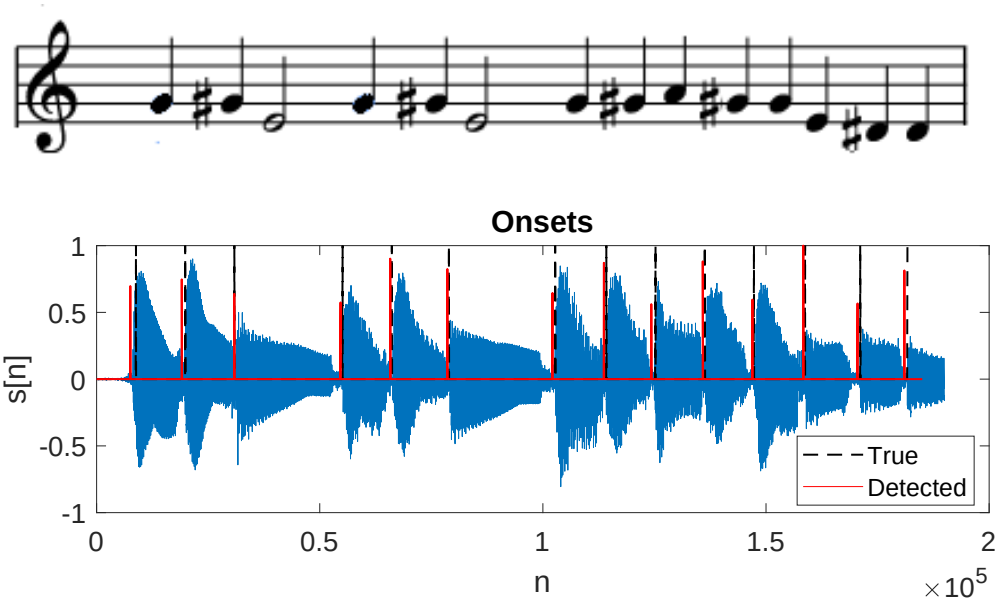
and the rule is

$$\text{note} = \begin{cases} \text{ACF} & \text{if ACF} \in \text{octave1} \wedge \text{ACF} \neq \mathbf{Y}_{\text{result}} + -12 \\ \mathbf{Y}_{\text{result}} & \text{if ACF} \in \text{octave1} \wedge \text{ACF} = \mathbf{Y}_{\text{result}} + -12 \\ \text{ACF} & \text{if ACF} \in \text{octave2} \\ \mathbf{Y}_{\text{result}} & \text{if ACF} \in \text{octave3} \end{cases}$$

5 Results and Future work

5.1 Results

At this point, both Onset Detection and Pitch Detection have been completed. The onsets of the input signal have been have been decided, and a final note has been applied to each one of them. The results are seen in the following figures, where the signal contains the 6 initial notes studied followed by a few more.



5.2 Future work

While this system guarantees accurate transcription of the guitar signal, Onset and Pitch Detection is just the basis of a realistic sound application. Any guitar signal will most probably contain chords as well, while this system is limited to detection of melodies consisting of single notes. Music signals are also highly susceptible to noise which can greatly impact both the localisation of the Onset and Pitch detection. Further work can be made, including but not limited at:

- Addition of a noise removal module : Music signals should always be recorded with proper equipment, in a suitable environment. A guitar microphone and a decent sound card are the minimum requirements, with the recording taking place in a sound insulated space. Noise removal is welcome even when these conditions are met, mandatory when not.

    1. Jashanpreet Kaur, Seema Baghla, Sunil Kumar. A Review: Audio Noise Reduction and Various Techniques. *International Journal of Advances in Science Engineering and Technology, ISSN: 2321-9009, Volume-3, Issue-3*, July 2015
    2. Guoshen Yu, Emmanuel Bacry, St´ephane Mallat. Audio Signal Denoising with complex wavelets and Adaptive Block Attenuation. *Conference Paper in Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference* May 2007

- Chord analysis : The randomness of the harmonic content of a note makes pitch estimation of a single note a challenging task. This problem extends and is amplified in the case of chord estimation. Further frequency analysis, as well as machine listening training has been used for this purpse, albeit more research is required.

    1. Nathan Lenssen. Applications of Fourier Analysis to Audio Signal Processing: An Investigation of Chord Detection Algorithms. *Claremont McKenna College Senior Thesis* April 2013.
    2. Alexander Sheh and Daniel P.W. Ellis. Chord Segmentation and Recognition using EM-Trained Hidden Markov Models. *International Symposium on Music Information Retrieval* June 2012

- Further Information Extraction : While onset times and pitch of any music signal is the minimum information required to transcribe a track, more characteristics of the signal can be extracted such as tempo, note duration (value), song classification etc. Knowledge of this information is vital to extending the transcription to polyphonic signals, containing more instruments than a single guitar. Pattern recognition algorithms have effectively been used for this purpse.

    1. Makarand Velankar, Dr. Parag Kulkarni. Pattern recognition for computational music.*Conference: Frontiers for research in speech*

and music. NIT, Rourkela. December 2017

2. Dan Ellis. Pattern Recognition Applied to Music Signals. *Laboratory for Recognition and Organization of Speech and Audio, Columbia University, New York.* July 2003

These additions applied to the current system would allow for transcription of a polyphonic music signal in a noisy environment, with no limitations to the musical content of the signal.

References

[1] Wikipedia. Harmonic. *https://en.wikipedia.org/wiki/Harmonic*, August 2018.

[2] Juan Pablo Bello Laurent Daudet Samer Abdallah Chris Duxbury Mike Davies Mark B. Sandler. A tutorial on onset detection in music signals. *IEEE*, September 2005.

[3] Juan Pablo Bello Laurent Daudet Samer Abdallah Chris Duxbury Mike Davies Mark B. Sandler. On the use of phase and energy for musical onset detection in the complex domain. *IEEE SIGNAL PROCESSING LETTERS, VOL. 11, NO. 6,*, June 2004.

[4] Philip McLeod. Fast, accurate pitch detection tools for music analysis. *Doctor of Philosophy at the University of Otago, Dunedin, New Zealand.*, May 2008.