International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-2 Issue-5, June 2013

Scale Identification of an Audio Input

Alan Shaji Idicula, Kalpana Balani, Preetesh Shetty, Sayali Thalve, T. Rajani Mangala

Abstract: Generally a music piece plays on one single scale. Each scale is uniquely characterized by a set of specific notes. In this paper, a methodology is presented for the recognition of musical scales that are associated with any piece of music. The procedure of recognition is essentially based on the comparison between the combinations of notes occurring in any music piece, with a predefined set of notes denoting a scale. To determine scale from a musical recording, frequency of the note is the characteristic which is used to identify a note and differentiate it from other notes. The proposed methodology led to the development of a system which can be exploited by a beginner learning music. The system performed successful recognition for the 92% of the tested recordings. It should be noted that the proposed system can operate in real time.

Keywords- audio input, frequency, matrix, music, note, scale

I.INTRODUCTION

An approach is presented for identification of musical scale of any audio signal. A musical piece will be taken as input; the scale it is being played on will be identified.

The scale shapes the music. In Western music, the smallest step between adjacent notes is found by making the next higher note have a frequency equal to 1.059435 times the frequency of the lower note. So middle C is 261.63 Hertz and the next higher note, C#, is 261.63 X 1.059435 = 277.18 Hertz. In many songs only some of the possible notes can be used. The scales are comprised of a collection of notes.[1]

In music, an octave is the interval between one musical pitch and another with half or double its frequency. For example, if one note has a frequency of 440 Hz, the note an octave above it is at 880 Hz, and the note an octave below is at 220 Hz. The ratio of frequencies of two notes an octave apart is therefore 2:1. Further octaves of a note occur at 2n times the frequency of that note (where n is an integer), such as 2, 4, 8, 16, etc. and the reciprocal of that series. For example, 55 Hz and 440 Hz are one and two octaves away from 110 Hz because they are 0.5 (or 2 −1) and 4 (or 22) times the frequency, respectively.[2] After the unison, the octave is the simplest interval in music. The human ear tends to hear both notes as being essentially "the same", due to closely related harmonics. Notes in an octave "ring" together, adding a pleasing sound to music. For this reason, notes an octave apart are given the same note name in the Western system of music notation—the name of a note an octave above A is also A.

Manuscript published on 30 June 2013.
* Correspondence Author (s)
Alan Idicula, Electronics & Telecommunication Engg, Mumbai University, Mumbai, India.
Kalpana Balani ,Electronics & Telecommunication Engg, Mumbai University, Mumbai, India.
Sayali Thalve, Electronics & Telecommunication Engg, Mumbai University, Mumbai, India.
Preetesh Shetty , Electronics & Telecommunication Engg, Mumbai University, Mumbai, India.
Mrs. T. Rajani Mangala, . Electronics & Telecommunication Dept, Mumbai University, Mumbai, India.

This is called octave equivalency, the assumption that pitches one or more octaves apart are musically equivalent in many ways, leading to the convention "that scales are uniquely defined by specifying the intervals within an octave".

The conceptualization of pitch as having two dimensions, pitch height (absolute frequency) and pitch class (relative position within the octave), inherently include octave circularity. Thus all C♯s, or all 1s (if C = 0), in any octave are part of the same pitch class.[2]

The frequency is an important component of any music. It is the main differentiating factor between two notes. Thus signal processing for frequency identification is the primary step for scale identification. Every music is made up of several notes. Many specific notes combined together form scales. For example, if the notes are C, C#, D, D#, E, F, F#, G, G#, A, A#, B, then the notes C, D, E, F, G, A, B are used in any combination to generate C scale. Thus, for any music input, if the frequency of each note is identified, the corresponding note can be determined. From this information, the scale of the music can be obtained.

II.LITERATURE SURVEY

Not much work has been done for the recognition of scale of any music piece but work has been done for recognition of chords using neural network approach. Sheh and Ellis proposed a statistical learning method for chord segmentation and recognition [4]. They used the hidden Markov models(HMMs) trained by the Expectation Maximization algorithm, and treated the chord labels as hidden values within the EM framework. In training the models, they used only the chord sequence as an input to the models and applied the forward-backward or Baunch-Welch algorithm to estimate the model parameters. The frame accuracy in percent they obtained was about 76% for segmentation and about 22% for recognition, respectively. The poor performance for recognition may be due to insufficient and unlabeled training data compared with a large set of classes (20 songs for 147 chord types).

Table.1

Frequencies in Hz of the 12 notes in all octaves

NOTES	OCTAVES							
	0	1	2	3	4	5	6	7
C	16.35	32.70	65.41	130.8	261.6	523.3	1047	2093
C#	17.32	34.65	69.30	138.6	277.2	554.4	1109	2217
D	18.35	36.71	73.42	146.8	293.7	587.3	1175	2349

D#	19.45	38.89	77.78	155.6	311.1	622.3	1245	2489
E	20.60	41.20	82.41	164.8	329.6	659.3	1319	2637
F	21.83	43.65	87.31	174.6	349.2	698.5	1397	2794
F#	23.12	46.25	92.50	185.0	370.0	740.0	1480	2960
G	24.50	49.00	98.00	196.0	392.0	784.0	1568	3136
G#	25.96	51.91	103.8	207.7	415.3	830.6	1661	3322
A	27.50	55.00	110.0	220.0	440.0	880.0	1760	3520
A#	29.14	58.27	116.5	233.1	466.2	932.3	1865	3729
B	30.87	61.74	123.5	246.9	493.9	987.8	1976	3951

Table 1 clearly shows that all notes have different frequencies, which makes it possible to uniquely identify each note separately. Notes have different frequencies for different octaves.

III. METHOD TO IDENTIFY THE MUSIC SCALE

The proposed method can be explained as follows:

A. Formatting

The musical piece is first sampled at 6000Hz such that each sample is represented by N bits using Pulse-code Modulation(PCM).The value of N can be 8,16,24 or 32 depending on the range of the output obtained during modulation.[3] The musical piece so generated is monaural or monophonic and has audio in a single channel.

B. Frequency Domain Transformation

The next step is the analysis of this music signal in the frequency domain. This is done by computing the Short Time Fourier Transform of the signal. The short-time Fourier transform (STFT), or alternatively short-term Fourier transform, is a Fourier-related transform used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time [5]. Mathematically Short Time Fourier Transform of a continuous signal is given as

$$\mathbf{STFT}\{\mathbf{x(t)}\} = \mathbf{X(\tau,\omega)} = \int_{-\infty}^{\infty} \mathbf{x(t)w(t-\tau)e^{-j\omega t}dt} \quad (1)$$

where w(t) is the window function, commonly a Hanning window or Gaussian bell centered around zero, and x(t) is the signal to be transformed. X (τ,ω) is essentially the Fourier Transform of x(t)w(t-τ), a complex function representing the phase and magnitude of the signal over time and frequency. The function to be transformed is multiplied by a window function which is nonzero for only a short period of time. The Fourier transform (a one-dimensional function) of the resulting signal is taken as the window is slid along the time axis, resulting in a two-dimensional representation of the signal. For a discrete signal it is given as

$$\text{STFT}\{x[n]\} = \text{X}(\text{m},\omega) = \sum_{n=-\infty}^{\infty} x[n]w[n-m]e^{-j\omega n} \quad (2)$$

with signal x[n] and window w[n]. In this case, m is discrete and ω is continuous. In the continuous-time case, the function to be transformed is multiplied by a window function which is nonzero for only a short period of time. The Fourier transform (a one-dimensional function) of the resulting signal is taken as the window is slid along the time axis, resulting in a two-dimensional representation of the signal. In the discrete time case, the data to be transformed could be broken up into chunks or frames (which usually overlap each other). Each chunk is Fourier transformed, and the complex result is added to a matrix, which records magnitude and phase for each point in time and frequency. Short Time Fourier Transform of the music input is calculated by taking a Hanning window of N samples at a time at sampling frequency (f_s) of 6000Hz.Taking the Fourier Transform produces N complex coefficients. Of these coefficients only half are useful (the last N/2being the complex conjugate of the first N/2 in reverse order, as this is a real valued signal). These N/2 coefficients represent the frequencies 0 to $f_s/2$ (Nyquist) and two consecutive coefficients are spaced apart by f_s /N Hz. To increase the frequency resolution of the window the frequency spacing of the coefficients needs to be reduced. There are only two variables f_s and N. Keeping f_s constant the window size N is increased to have a good frequency resolution appropriate for the application. When a window function is used, little information at the tapered ends is obtained. One way to fix this is to use a sliding window with an overlap. Overlapping transforms works somewhat like a "zoom". It does this by effectively stretching the time scale. Adding overlap does indeed overlap small time events, it enables much greater visibility of frequency changes with time. The resulting sequence is approximated to the original sequence as best as possible. The Short Time Fourier Transform yields information about time, frequency, power spectral density and power by computing a matrix for each of the above entities.

C. Classification

A note is the fundamental unit in any music. There are total twelve notes. Each note has a particular frequency and this frequency doubles with every higher octave irrespective of the instrument. The notes occurring in the music input are extracted using the frequency matrix generated by Short Time Fourier Transform. The number of occurrences of each note in the music input is computed. These numbers of occurrences of each note in music input are given appropriate weights according to the scale they are being compared with, and the scale that yields the maximum sum is the scale in which the music is being played.

IV. EXPERIMENTAL RESULTS

Musical data from all the twelve major musical scales was used in this analysis. The musical data taken as input is comprised of only instrumental data. The output thus obtained is compared with the musical data of all the possible twelve major musical scales.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

A. Choice of Window

Hanning Window was chosen to determine the Short time Fourier transform as it provides minimum power in the side lobes as compared to Hamming window.

B. Choosing Length of FFT Samples

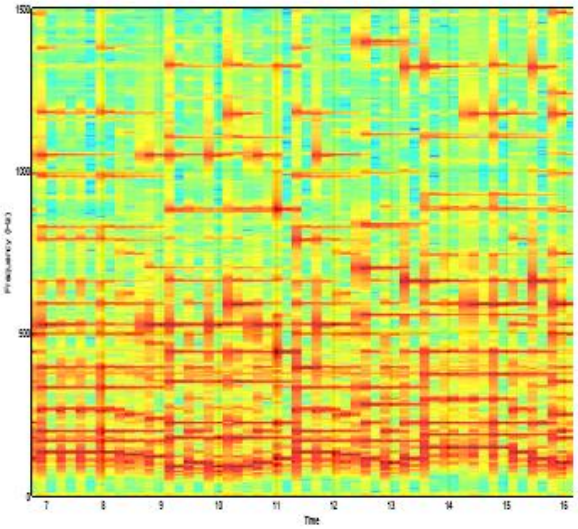

Fig.1 STFT considering 2048 samples

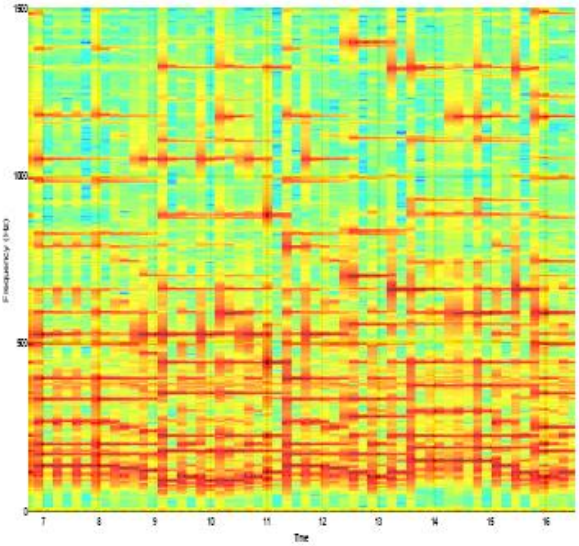

Fig.2 STFT considering 4096 samples

C. Determining the scale of the audio input

The input is first converted into frequency domain using Short-time Fourier Transform and then the occurrence of each note is estimated. Considering 'newyork.wav' as to be the input, the Short-time Fourier Transform of the input and the histogram of the input in the frequency domain can be shown as follows:

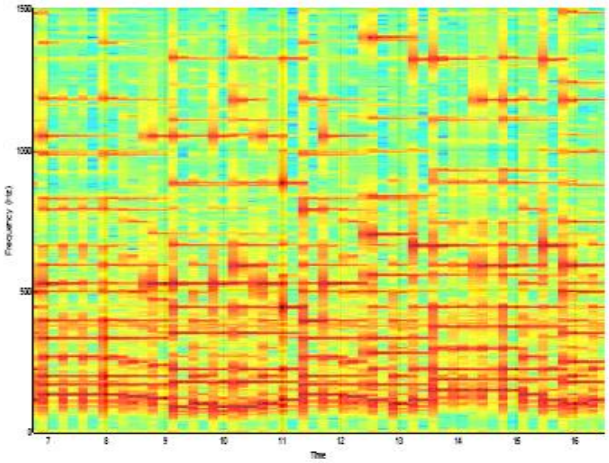

Fig.3 Spectrogram of 'newyork.wav'

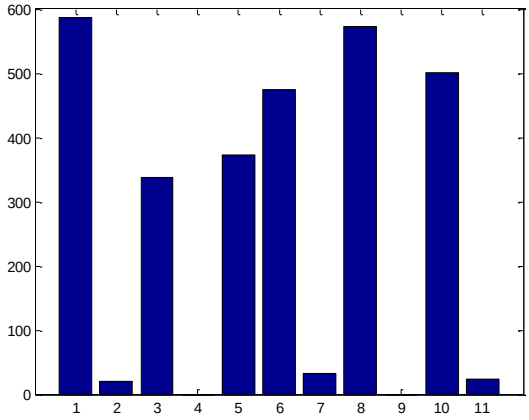

Fig.4 Histogram of the input 'newyork.wav'

The frequency of occurrence of each note in an input audio file at a power level greater than a predefined value is calculated.

Table 2
Frequency of occurrence of each note in the input newyork.wav

Notes	Frequency of occurrence
C	280
C#	10
D	163
D#	0
E	183
F	230
F#	16
G	284
G#	0
A	244
A#	12
B	118

Table 2 shows that some of the notes like C, D, E, F, G, A and B have occurred more frequently than the remaining notes in the input newyork.wav. Hence the notes such as 'C Sharp', 'D Sharp', 'F Sharp', 'G Sharp' and 'A Sharp' are less likely to occur.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Considering appropriate weights for each scale, the scale to which the input belongs is then determined.

Table 3 Sum of occurrence of notes pertaining to each scale

Scale(major)	Sum
C	3090
G	2048
D	2083
A	1511
E	1174
B/C-flat	696
F	2866
b-flat	2494
E-flat	1993
A-flat	1677
D-flat	1138
F-sharp/G-flat	799

Table 3 shows that weights associated with C scale produces the maximum value for newyork.wav. Thus this approach has been used to separate the scale being played.

This method was then applied to inputs from each of the 12 scales and the output obtained was as follows:

Table 4 Outputs obtained for each of the 12 scales

Input	Actual scale	Determined scale
newyork.wav	C	C
majorcradle.wav	A flat	A flat
anger.wav	G flat	D flat
roving.wav	B flat	B flat
wewish.wav	D flat	D flat
childth.wav	D	D
first.wav	E flat	E flat
study.wav	E	E
virgin.wav	F	F
manger.wav	G	G
enfant.wav	A	A
major.wav	B	B

Out of all 12 inputs, only input pertaining to G flat scale gave as wrong scale as D flat. Thus the software provided satisfactory result for 11 of the 12 inputs providing 92% accuracy.

V. CONCLUSION

An approach is presented in this paper to determine the musical scale by taking a music input of approximately 20 seconds. Thus using these features a novice would be able to determine the appropriate chords and scale of the musical data.

REFERENCES

[1] www.music.vt.edu/musicdictionary/appendix/scales/scales/ majorscales. html

[2] Cooper, Paul (1973). Perspectives in Music Theory: An Historical-Analytical Approach, p.16. ISBN 0-396-06752-2.

[3] http://www.mathworks.in/help/matlab/ref/wavread.html

[4] A. Sheh and D. P. Ellis, “Chord segmentation and recognition using EM-trained hidden Markov models,” in Proceedings of the International Symposium on Music Information Retrieval, Baltimore, MD, 2003.

[5] E. Jacobsen and R. Lyons, The sliding DFT, Signal Processing Magazine vol. 20, issue 2, pp. 74–80 (March 2003).

Alan Shaji Idicula, B.E. Electronics&Telecommunication, Mumbai University

Kalpana Balani, B.E. Electronics&Telecommunication, Mumbai University

Sayali Thalve, B.E. Electronics&Telecommunication, Mumbai University

Preetesh Shetty, B.E. Electronics&Telecommunication, Mumbai University

Mrs. T. Rajani Mangala, . Electronics&Telecommunication Dept, VESIT, Mumbai University

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)