Hidden Markov Model based Recognition of Musical Pattern in South Indian Classical Music

M.S. Sinith, K. Rajeev
Dept. of Electronics and Communication Engineering
College of Engineering, Trivandrum, India
sinith@gmail.com
rajeev.kumaraswamy@gmail.com

***Abstract*— Automatic recognition of musical patterns plays a crucial part in Musicological and Ethno musicological research and can become an indispensable tool for the search and comparison of music extracts within a large multimedia database. This paper finds an efficient method for recognizing isolated musical patterns in a monophonic environment, using Hidden Markov Model. Each pattern, to be recognized, is converted into a sequence of frequency jumps by means of a fundamental frequency tracking algorithm, followed by a quantizer. The resulting sequence of frequency jumps is presented to the input of the recognizer which use Hidden Markov Model. The main characteristic of Hidden Markov Model is that it utilizes the stochastic information from the musical frame to recognize the pattern. The methodology is tested in the context of South Indian Classical Music, which exhibits certain characteristics that make the classification task harder, when compared with Western musical tradition. Recognition of 100% has been obtained for the six typical music pattern used in practise. South Indian classical instrument, flute is used for the whole experiment.**

I. Introduction

Digital sound processing tools offer new possibilities to the analysis of musical structures, the modeling of the acoustic characteristics of an instrument and the musical pattern comparison and recognition. The earliest and most well known survey of digital signal processing techniques for the production and processing of musical sounds was authored by [1] in 1976. Today, the computational efficiency of computers permits the research community to deal with tasks that were unrealistic to face before. Such an important task of great interest to the musicologist is the semi-automated search of specific sound patterns within a large number of stored sound files. These musical patterns have been shaped and categorized through practice and experience in many musical traditions.

This paper proposes a scheme for the recognition of such predefined musical patterns in a monophonic environment in the context of south Indian classical music. Indian classical music instrument Flute is used for the whole experiment. From a large number of types of transistory musical patterns encountered in practice in different instrumental style, we have selected the six typical cases.

Indian classical music is defined by two basic elements. They are Raga (classical mode) and Taal(rhythm). In any Indian classical composition, the music is based on a drone, ie, a continual pitch that sounds throughout the concert, which is a tonic. This acts as a point of reference for everything that follows, a home base that the musician returns to after a flight of improvisation. The result is a melodic structure that is easily recognizable, yet infinitely variable. *A Raga is popularly defined as a specified combination, decorated with embellishments and graceful consonances of notes within a mode which has the power of evoking a unique feeling distinct from all other joys and sorrows and which possesses something of a transcendental element* [2]. In other words, a Raga is a characteristic arrangement or progression of notes whose full potential and complexity can only be realised in exposition. This makes it different from the concept of a scale in Western music. A Raga is characterised by several attributes, like its Vaadi-Samvaadi, Aarohana-Avrohana and Pakad, besides the sequence of notes which denote it. It is important to note here that no two performances of the same Raga, even two performances by the same artist, will be identical.

The recognition scheme that we propose consists of three stages. In the first stage, tempo-tracking stage, the onset of each pitch is detected and its duration calculated. The musical piece to be analyzed is segmented with the duration taken as the window width. In the second stage lies a fundamental frequency tracking algorithm, which generates fundamental frequency from unknown musical pattern. Each extracted fundamental frequency is quantized using the pitch values of South Indian music [3]. Several time-domain and frequency domain algorithm were considered [4]-[7]. In addition a new algorithm is developed which is similar to period histogram [4]. In the third stage the output from the second stage is given to a pre-trained hidden markov models(one for each musical pattern).

II. Feature generation

Many methods (both frequency-domain and time-domain) have been proposed in the literature.Those methods are

Frequency-domain approaches: Schroeder's histogram [4], Schroeder's Harmonic Product Spectrum [4], Piszscalski's method [5] and Brown's pattern recognition method based on the properties of a constant-transform [6].

Time-domain methods: Cooper and Kia's method [8] and Brown's narrowed autocorrelation method [7]. Also Tolonen's

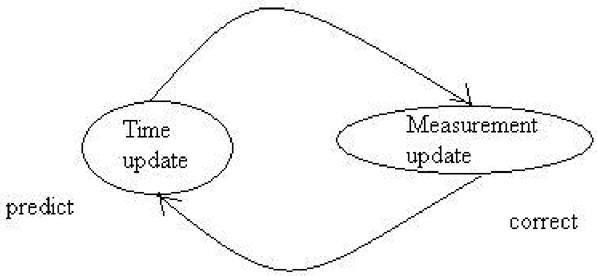

Fig. 1. Discrete kalman filter flow diagram

method was used [9].

We propose a new method, in which first tempo tracking is done to find the transition points of each of the pitches in the music piece, is used to find the window length adaptively.The fundamental frequency in each of these windows is calculated using the method suggested in Schroeder's histogram [4], but with a slight modification that the fundamental frequency is tracked using Greatest Common Divisor method in the specified window.This method can be used for perfect fundamental frequency tracking where the fundamental frequecy component is missing in the fast fourier transform.

A. Tempo tracking

The tempo tracking system is divided into two sections. In the first section, Short Time Fourier Transform of the audio signal with appropriate time-frequency resolution is calculated and used to obtain a rough estimate of the Tempo of the musical Signal[13]. This data is given to a Kalman filter to generate short term predictions of future beats in the audio signal and there by reducing the error adaptively occurred during the observation. The algorithm is evaluated over a range of musical styles by comparing the predicted output to known tempo tracks.The tempo tracking process used in this paper can be subdivided into i)Onset Analysis and ii)Estimation using Kalman filter.

1) Onset Analysis: The aim of the onset analysis stage is not to explicitly detect the locations of note onsets, rather to generate a midlevel representation of the input which emphasizes the onset locations. To reflect this need we choose an onset detection function a continuous signal with peaks at onset positions, as the input to the tempo analysis stage. Each onset detection function is generated from frame based analysis using a window size of 1024 samples and a step increment of 1024 samples, from audio sampled at 22050 Hz, giving a temporal resolution of 46.4 ms. The detection algorithm portrays the complex spectral difference between the discrete Fourier transform (DFT) of the current frame $X_k[n]$ and a predicted target frame,X_{k+1}[n]. Onset detection function peaks are the result of transitions in frequency or pitch changes of the musical signal enabling the approach to detect onsets in a wider range of signals.

2) Estimation using Kalman filter: The detection of the onset of a particular tone or frequency is determined by taking the difference of adjacent Short Time Fourier Transform (STFT) of the audio signal. If this difference is above a predefined threshold, then the frequency transition has occurred in the final block of audio signal. To get precise information of the onset of a particular frequency STFT with 75 percent overlap is used. Thus, the positions of onset of different frequencies are calculated. These calculations are converted into Beats per minute (Bpm) and the average of these values give the exact tempo in Beats per minute.

Mathematically, a dynamical system is characterized by a set of state variables and a set of state transition equations that describe how the system model evolves with time. For example, a perfect metronome can be described as a dynamical system with two state variables: a beat Δ and a period τ Where τ is the time at which the frequency onset or frequency transition occurs and Δ is the duration for which the corresponding frequency persist. Given the values of state variables at $(j-1)^{th}$step as $\widehat{\tau}_{j-1}$ and $\widehat{\Delta}_{j-1}$ and next beat occurs at $\widehat{\tau}_j = \widehat{\tau}_{j-1} + \widehat{\Delta}_{j-1}$.

The period of a perfect metronome is constant so $\Delta_j = \Delta_{j-1}$.By using vector notation we can write a linear state transition model as

$$S_j = [\widehat{\tau}_j, \widehat{\Delta}_j] \tag{1}$$

when the initial state $S_0 = [\widehat{\tau}_0, \widehat{\Delta}_0]$ is given ,the system is fully specified. Since the metronome is perfect the period stays constant. Such a deterministic model is not realistic for natural music performance and can not be used for tracking the tempo in presence of tempo fluctuations and expressive timing deviations. Tempo fluctuations may be modeled by introducing a noise term ν_j that corrupts the state vector

$$S_j = AS_{j-1} + \nu_j \tag{2}$$

where ν is a gaussian random vector with mean zero and diagonal covariance matrix Q.The tempo will drift from the initial tempo quickly if the variance of ν is large. On the other hand when Q→0, we have the constant tempo case. In a music performance, the actual beat $\widehat{\tau}$ and the period $\widehat{\Delta}$ cannot be observed directly. For example, suppose, an expert drummer is tapping along a performance at the beat level. If the task would be repeated on the same piece, we would observe each time a slightly different tempo track. As an alternative, suppose we would know the score of the performance and identify onsets that coincide with the beat. However, due to small scale expressive timing deviations, these onsets will be also noisy, i.e. we can at best observe noisy versions of actual beats. We will denote this noisy beat by τ in contrast to the actual but unobservable beat $\widehat{\tau}$. Mathematically we have

$$\tau_j = \widehat{\tau}_j + w_j \tag{3}$$

The Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated

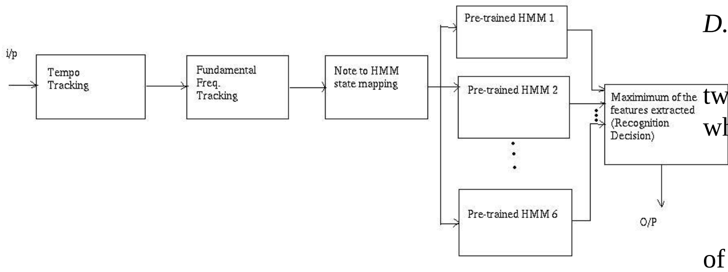

Fig. 2. Raga identification block diagram

error covariance, when some presumed conditions are met. The Kalman filter estimates a process by using a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of (noisy) measurements. As such, the equations for the Kalman filter fall into two groups: time update equations and measurement update equations.

B. Fundamental frequency tracking algorithm

Fundamental frequency and pitch are often used interchangeably, although their values do not always coincide. The perception of pitch is a psychoacoustical phenomenon, whereas the fundamental frequency is a quantity that can be calculated algorithmically for periodic or quasi-periodic signals.The Tempo extracted is utilized to separate the musical signal into several segments each having a single pitch.The fundamental frequency of each of these segments is calculated using a frequency-domain algorithm developed that can be considered as a modification of Schroeder's histogram [4].The algorithm used in [10] is similar to that used in [4] but the drawback of both of these algorithms is that it fails to cope with the problem of missing fundamentals.To compensate this we have used an algorithm in which greatest common divisor of the frequencies is taken, which helps in finding the fundamental frequency, even if it is missing.In the new algorithm derived first the FFT values of indices 513 to 1024 are made zero as they are symmetrical.The remaining 512 are arranged in descending order of their amplitude.Ten of them are selected and Greatest Common Divisor(GCD) of each of each and every pairs of them are taken.Mode of this is taken to obtain the most recurring value.This gives the fundamental frequency.

C. Quantization of extracted frequencies

The fundamental frequency so obtained is passed to a non-uniform quantizer having twelve levels,which are similar to pitch values derived in [3].The actual quantization steps are given by

$$24 * log(JI) \tag{4}$$

where JI is Just Intonation ratio given in [3].

Just Intonation is the ratio of fundamental frequency f_i to the frequency of C note, called "sruthi",in South Indian music.

D. Mapping frequencies to states of Hidden Markov Model

The twelve quantized frequencies so obtained are mapped to twelve states of the hidden markov model which are 1,2,3,...12. which will be discussed in detail in the next section.

III. Hidden Markov Model

Hidden Markov models (HMMs) are mathematical models of stochastic processes, i.e. processes which generate random sequences of outcomes according to certain probabilities. A simple example of such a process is a sequence of coin tosses. More concretely, an HMM is a finite set of states, each of which is associated with a (generally multidimensional) probability distribution. Transitions among the states are governed by a set of probabilities called transition probabilities. In a particular state, an outcome or observation can be generated, according to the associated probability distribution. It is only the outcome not the state that is visible to an external observer. So states are hidden and hence the name hidden Markov model.

In order to define an HMM completely, the following elements are needed
The number of states of the model, N
The number of observation symbols in the alphabet, M.
A set of state transition probabilities

$$A = \{a_{ij}\}$$

$$a_{ij} = P\{q_{t+1} = j/q_t = i\}, 1 \le i, j \le N$$

where q_t denotes the current state.
A probability distribution in each of the states,

$$B = \{b_{jk}\}$$

$$b_{jk} \ge P\{\alpha_t = v_k/q_t = j\}, 1 \le j \le N, j \le N, 1 \le k \le M$$

where v_k denotes the k_{th} observation symbol in the alphabet and t the current parameter vector.
The initial state distribution,

$$\pi = \{\pi_i\}$$

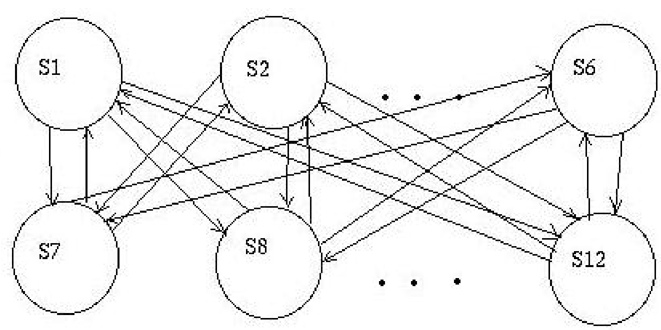

Fig. 3. model of the HMM used

where $\pi_i = p(q_1 = i), 1 \leq i \leq N$
Thus, an HMM can be compactly represented as

$$\lambda = \{A, B, \pi\}$$

Hidden Markov models and their derivatives have been widely applied to speech recognition and other pattern recognition problems [12]. Most of these applications have been inspired by the strength of HMMs, ie the possibility to derive understandable rules, with highly accurate predictive power for detecting instances of the system studied, from the generated models. This also makes HMMs the ideal method for solving Raga identification problems.

A. Selection of Hidden Markov Model

The HMM used in our solution is significantly different from that used in, say word recognition. This HMM, which we call from now on, can be specified by considering each of its elements separately.Each note in each octave represents one state in λ . Thus, the number of states actually required is N=12x3=36 (Here, we are considering the three octaves of Indian classical music, namely the Mandra, Madhya and Tar Saptak, each of which consist of 12 notes) as used in [6]. Instead we have first mapped the frequencies in the Mandra and Tar Saptak into the corresponding frequencies in the Madhya.Hence the total number of states for the HMM is N=12. Pattern recognition in [10] is based on the frequency difference, where the total number of states needed for the HMM is very high compared to the HMM whose states are taken on the basis of absolute frequencies.So we have adopted absolute frequency as the basis for HMM.

The transition probability $A = \{a_{ij}\}$ represents the probability of note j appearing after note i in a note sequence of the Raga represented by λ.

The initial state probability $\pi = \{\pi_i\}$ represents the probability of note i being the first note in a note sequence of the Raga represented by λ .

The outcome probability $B = \{b_{ij}\}$ is set according to the following formula
$B_{ij} = 0, \forall i \neq j,$
$B_{ij} = 1, \forall i = j.$

The last condition takes the hidden character away from, but it can be argued that this setup suffices for the representation of Ragas, as our solution distinguishes between performances of distinct raga on the basis of the exact order of notes sung in them and not on the basis of the embellishments used.Thus, at each state α in λ, the only possible outcome is note α.

B. Training the HMM

There are several algorithms for training a Hidden Markov Model. Main among them are Viterbi algorithm,Baum-Welch algorithm [12]. we have used Viterbi algorithm for the training purpose. The Viterbi algorithm is a dynamic programming algorithm for finding the most likely sequence of hidden states - called the Viterbi path - that result in a sequence of observed events, especially in the context of hidden Markov models. The forward algorithm is a closely related algorithm for computing the probability of a sequence of observed events.

The algorithm makes a number of assumptions. First, both the observed events and hidden events must be in a sequence. This sequence often corresponds to time. Second, these two sequences need to be aligned, and an observed event needs to correspond to exactly one hidden event. Third, computing the most likely hidden sequence up to a certain point t must depend only on the observed event at point t, and the most likely sequence at point $t - 1$. These assumptions are all satisfied in a first-order hidden Markov model.

The Viterbi algorithm operates on a state machine assumption. That is, at any time the system we are modeling is in some state. There are a finite number of states, however large, that can be listed. Each state is represented as a node. Multiple sequences of states (paths) can lead to a given state, but one is the most likely path to that state, called the ”survivor path”. This is a fundamental assumption of the algorithm because the algorithm will examine all possible paths leading to a state and only keep the one most likely. This way the algorithm does not have to keep track of multiple paths, only one per state.

C. Features extracted from HMM

The feature used for the recognition process is the summation of the probability emitted by the HMM.The model producing the highest value is recognized to be the pattern of the wavefile given as input.

IV. Application of the method in the context of South Indian classical music

One model for each of the eight musical pattern selected is constructed and the musical clip whose pattern to be recognized is given as input to each of these six models. The parameter emitted by these models is the summation of the transition probabilities through which the musical clip given as input passes through. The Six typical music patterns that we selected from South Indian classical music are i)Sankarabaranam ii)Nata iii)Natakurunji iv)Kalyani v)Sivaranjini and vi)Abogi.
If the notes present, (as in 12 note representation [3] in the South Indian music) in each of these musical patterns are represented by 1 and those notes which are absent are represented by 0. It can be represented as in the table.

TABLE II
Pattern of Ragas

$Raga$	$pattern$
Nata	111001010011
Sankarabaranam	101011010101
Sivaranjini	101100010100
Abogi	101101000100
Kalyani	101010110101
Natakurunji	101011010110

TABLE I

FEATURES EXTRACTED FOR DIFFERENT RAGAS

Raga	$Nata$	$S.abaranam$	$Abogi$	$sivaranjini$	$Kalayani$	$N.kurunji$	R/NR
Nata	0.2017	0.1846	0.1804	0.1704	0.1789	0.1497	R
sankarabaranam	0.1977	0.2107	0.1932	0.1855	0.2071	0.1877	R
Abogi	0.2055	0.2088	0.2180	0.2063	0.2117	0.2052	R
sivaranjini	0.2030	0.1953	0.2073	0.2195	0.2108	0.2011	R
Kalayani	0.1958	0.1919	0.1876	0.1853	0.2139	0.1791	R
Natakurunji	0.1982	0.1940	0.1845	0.1958	0.2024	0.2181	R

V. RESULTS

A. Tempo tracking

The periods Δ_j obtained are applied to the Kalman filter to track the tempo variations from period measurements. Errors in the measurement and calculation is compensated in the estimation process by the kalman filter. The tracking performance is shown in fig.The tempo obtained after Kalman filtering resembled the exact tempo of the audio input file. To evaluate the effectiveness of the above modeled tempo tracking system, The system is thus shown to be following the tempo variations of the audio effectively. wavefiles with different tempo(BPM) are given as input to the tempo tracker. The pitch time period Δs for these wavefiles are estimated with and without using kalman filter and are plotted in figs.The output of the kalman filter had the tracked tempo most closer to the actual tempo of the music clip given for tempo tracking.

B. Raga Identification

From a large number of types of transistory musical pattern encountered in practice in different instrumental style, we have selected six typical cases. Several musical clips from these six patterns are tested for recognition and a recognition rate of 100% has been obtained.

The results given above are obtained by giving a wavefile to the HMM models of all the six Musical patterns. (Musical pattern) Raga names given in Italics represent the Hidden Markov Model of the respective raga and the others represent the wavefile given as input in the corresponding Raga model.

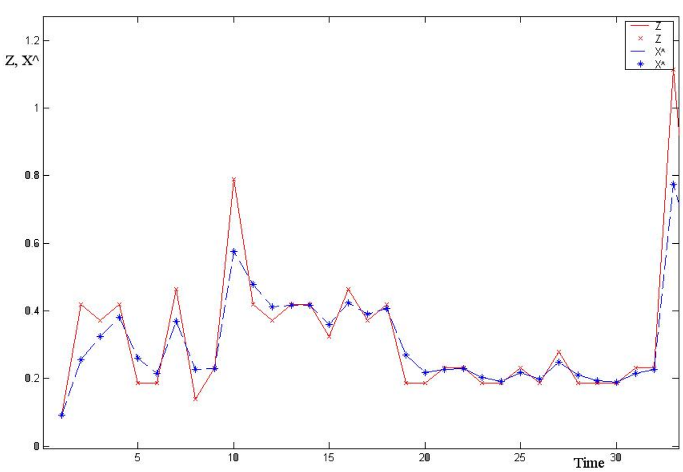

Fig. 4. estimated tempo with and without using kalman filter at 80bpm

The required musical clip to be recognized is given as input to all the eight Hidden Markov Models. The parameters emitted by each of these models are given in the table.The extracted parameter is the summation of the emitted probabilities in each of the state. The model giving the highest value is recognized to be the pattern of the wavefile given as input.For example, in the case of Abogi, The horizontal row for Abogi represent the features extracted from different models for the music clip in the Raga Abogi. In the horizontal row the maximum value is 0.2180 which corresponds to the column Abogi. So the given musical clip is recognized to be of Raga Abogi.

Acknowledgments

First Author Sinith thanks all his friends who have given valuable information relating to the work.we also thank Mr.Thor, University of Sussex, England and Arvindh Krishnaswamy, Stanford University for the valuable insight given by them during the work.

REFERENCES

[1] J. A. Moorer, " Signal processing aspects of computer music A survey," *Proceedings of the IEEE* ,vol. 65, no. 8.

[2] Gaurav Pandey, "Tansen:A system for automatic raga identification," *Indian International conference on Artificial Intelligence 2003*, Hyderabad, India, 2003

[3] Arvindh Krishnaswamy, "Application of pitch tracking to South Indian classical music," *Proceedings of IEEE, ICASSP*, Hong Kong, China, April 2003.

[4] M. R. Schroeder, " Period histogram and product spectrum: New methods for fundamental-frequency measurement", *Journal of the Acoustical Society of America*, , vol. 43, no. 4, 1968.

[5] M. Piszczalski and B. Galler, "Predicting musical pitch from component frequency ratios". *Journal of the Acoustical Society of America,*, vol. 66, no. 3, 1979.

[6] J. C. Brown,, "Musical fundamental frequency tracking using a pattern recognition method", *Journal of the Acoustical Society of America*, vol. 92, no. 3, 1992.

[7] J. C. Brown and B. Zhang, "Musical frequency tracking using the methods of conventional and narrowed autocorrelation",*Journal of the Acoustical Society of America*, vol. 89, no. 5, 1991.

[8] D. Cooper and K. C. Ng, "A monophonic pitch-tracking algorithm based on waveform periodicity determinations using landmark points", *Computer Music Journal*, vol. 20, no. 3, Fall 1996.

[9] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model", *IEEE Transactions on Speech and Audio Processing*,vol. 8, no. 6, November 2000.

[10] Aggelos Pikrakis, " Recognition of isolated musical pattern using context dependent dynamic time warping" *Transaction of speech and audio signal processing,* ,vol. 11, no. 3,May 2003.

[11] Aggelos Pikrakis, " Classification of musical patterns using variable duration hidden hidden Markov model" *Transaction of speech and audio signal processing,* ,issue 99, pages 1-13, 2005.

[12] L.R.Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition" *Proceedings of IEEE*, Vol.77,No. 2, pp. 257-286, February 1989.
[13] Matthew E. P. Davies and Mark D. Plumbley, "Causal Tempo traking of audio" *ISMIR 2004*,Barcelona, Spain, 2004.
[14] Matthew E. P. Davies and Mark D. Plumbley, "On tempo tracking: Tempogram Representation and Kalman filtering" *ICMC 2000*, Berlin, Germany, 2000.
[15] J. Deller, J. Proakis, and J. Hansen, *Discrete-Time Processing of Speech*, McMillan, 1993.