DISCOVERY OF SYLLABIC PERCUSSION PATTERNS IN TABLA SOLO RECORDINGS

Swapnil Gupta* swapnil.gupta01@estudiant.upf.edu
Ajay Srinivasamurthy* ajays.murthy@upf.edu
Manoj Kumar† manojpamk@gmail.com
Hema A. Murthy† hema@cse.iitm.ac.in
Xavier Serra* xavier.serra@upf.edu
*Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
†DONlab, Indian Institute of Technology Madras, Chennai, India

ABSTRACT

We address the unexplored problem of percussion pattern discovery in Indian art music. Percussion in Indian art music uses onomatopoeic oral mnemonic syllables for the transmission of repertoire and technique. This is utilized for the task of percussion pattern discovery from audio recordings. From a parallel corpus of audio and expert curated scores for 38 tabla solo recordings, we use the scores to build a set of most frequent syllabic patterns of different lengths. From this set, we manually select a subset of musically representative query patterns. To discover these query patterns in an audio recording, we use syllable-level hidden Markov models (HMM) to automatically transcribe the recording into a syllable sequence, in which we search for the query pattern instances using a Rough Longest Common Subsequence (RLCS) approach. We show that the use of RLCS makes the approach robust to errors in automatic transcription, significantly improving the pattern recall rate and F-measure. We further propose possible enhancements to improve the results.

1. INTRODUCTION

In many music cultures, music is sometimes transmitted partly through speech, using what are variously called vocables, oral mnemonics, solfège, etc [12]. In the case of several percussion traditions, the choice of vowels and consonants is such that the syllables closely represent the underlying acoustic phenomenon they represent. The term *acoustic-iconic mnemonic* systems coined by Hughes [12] explains this mnemonic based syllable systems where the core aspect is the similarity of the phonetic features of the syllables with the acoustic properties of the sounds they represent. A well studied example of such a system is the tabla, where the repertoire and technique is transmitted with the help of a system based on onomatopoeic oral syllables [18]. In this paper, we explore the use of the mnemonic syllable system of tabla for the discovery of percussion patterns. The use of these mnemonics allows us to work with a musically relevant representation that truly reflects the underlying timbre, articulation and dynamics of the patterns played.

Automatic discovery of patterns is a relevant Music Information Retrieval (MIR) task. It has applications in enriched and informed music listening, enhanced appreciation for listeners, in music training, and in aiding musicologists working on such music cultures. We use the onomatopoeic oral mnemonic syllables to represent, transcribe and search for patterns in audio recordings of tabla solos. We first build a set of query patterns from the corpus of scores in our dataset. Given an audio recording, we automatically transcribe it into a sequence of syllables. We then propose a method for searching the query patterns in the automatically transcribed score using approximate string search. We also propose several extensions to improve the search performance. We first provide a brief introduction to tabla.

1.1 Tabla and its solo performances

Tabla is the main rhythm accompanying instrument in Hindustani music, the art music tradition from North India. It consists of two drums: a left hand bass drum called the *bāyān* or *diggā* and a right hand drum called the *dāyān* that can produce a variety of pitched sounds [15]. To showcase the nuances of the *tāl* (the rhythmic framework of Hindustani music) as well as the skill of the percussionist with the tabla, Hindustani music performances feature tabla solos. A tabla solo is intricate and elaborate, with a variety of pre-composed forms used for developing further elaborations. There are specific principles that govern these elaborations [10, p. 42]. Musical forms of tabla such as the *ṭhēkā, kāyadā, palaṭā, rēlā, pēśkār* and *gaṭ* are a part of the solo performance and have different functional and aesthetic roles in a solo performance.

Playing a tabla is taught and learned through the use of onomatopoeic oral mnemonic syllables called the *bōl*, which are vocal syllables corresponding to different timbres that can be produced on the tabla. However, several bōls correspond to the same stroke played on the tabla, creating a many bōl to same timbre mapping, which can be exploited to discover acoustically similar patterns. Though the primary function of the bōls is to provide a representation system, a rhythmic vocal recitation of the bōls, which requires high skills, is inserted into solo performances for music appreciation.

Tabla has different stylistic schools called *gharānās*. The

Sym.	bōls	Sym.	bōls
DA	D, DA, DAA	NA	N, NA, TAA, TU
KI	KA, KAT, KE, KI, KII	DIN	DI, DIN, DING, KAR, GHEN
GE	GA, GHE, GE, GHI, GI	KDA	KDA, KRA, KRI, KRU
TA	TA, TI, RA	TIT	CHAP, TIT

Table 1: The bōls used in tabla, their grouping, and the symbol we use for the syllable group in this paper. The symbols DHA, DHE, DHET, DHI, DHIN, RE, TE, TII, TIN, TRA have a one to one mapping with a syllable of the same name and hence not shown in the table.

repertoires of major gharānās of tabla differ in aspects such as the use of specific bōls, the dynamics of strokes, ornamentation and rhythmical phrases [4, p. 60]. But there are also many similarities due to the fact that the same forms and standard phrases reappear across these repertoires [10, p. 52]. This enables in creation of a library of standard phrases or patterns across compositions of different gharānās.

1.2 Previous Work

Early research related to tabla focused mainly on stroke transcription, as seen in the work of Gillet [9]. Chordia [6] extended the work adding additional features and classifiers, using a larger and more diverse dataset. The use of tabla syllables in a predictive model for tabla stroke sequence was also demonstrated recently by Chordia et al. [7]. Recent work in transcription has been reported for Mridangam, the percussion accompaniment used in South Indian Carnatic music, by Kuriakose et al. [13] and Anantapadmanabhan et al. [1]. The transcription task has a definite analogy to speech recognition and we can apply several tools and knowledge from this well explored research area with many state of the art algorithms and systems [11].

There is significant literature on pattern search and retrieval from percussion solos. Nakano et al. [16] address the problem of drum pattern retrieval using an HMM based approach using onomatopoeia as the representation for drum patterns, retrieving known fixed sequences from a library of drum patterns with snare and bass drums. We use a similar approach, the main difference being that we use a musically well grounded syllabic representation. Recently, Srinivasamurthy et al. [20] demonstrated the use of syllable level HMM followed by a string edit distance to transcribe and classify percussion patterns in Beijing Opera. Tsunoo et al. [21] also demonstrated a music classification task using K-means clustering of bar-long percussive patterns and bass lines extracted using one-pass dynamic programming. While the last two mentioned approaches aim at classification of patterns, we address the general task of retrieving patterns from recordings of full length solo compositions.

Transcription is often inaccurate with many errors, and any pattern search on transcribed data needs to use approximate string search algorithms. There are several attempts to deal with search in symbolic sequences [22]. Well explored techniques such as longest common subsequence (LCS) do not consider the local correlation while searching for a subsequence [14]. To overcome this limitation, Lin et al. [14] proposed a novel Rough Longest Common Subsequence (RLCS) method for music matching. Dutta et al. [8] used a modified version of RLCS for motif spotting in ālāpanas of Carnatic music. We propose to use a similar approach with minor modifications to suit the symbolic domain specific to our use case. To the best of our knowledge, this is the first work to explore syllabic pattern discovery as applied to tabla solos in Hindustani music.

2. PROBLEM FORMULATION

We formulate the problem of discovery of percussion patterns in tabla solo recordings. We present a general framework for the task, while outlining some of the challenges. The approach we explore in this paper is to use syllables to define, transcribe, and eventually search for percussion patterns. We build a fixed set of syllabic query patterns. Given an audio recording, we obtain a time-aligned syllabic transcription using syllable level timbral models. For each of the query patterns in the set, we then perform an approximate search on the output transcription to obtain the locations of the patterns in the audio recording. We describe each of the steps in detail.

We first compile a comprehensive set of syllables in tabla. Although, the syllables vary marginally within and across gharānās, several bōls can represent the same stroke on the tabla. To address this issue, we grouped the full set of 41 syllables into timbrally similar groups resulting into a reduced set of 18 syllable groups as shown in Table 1. Though each syllable on its own has a functional role, this timbral grouping is presumed to be sufficient for discovery of percussion patterns. For the remainder of the paper, we limit ourselves to the reduced set of syllable groups and use them to represent patterns. For convenience, when it is clear from the context, we call the syllable groups as just syllables and denote them by the symbols in Table 1. Further, we use bōls and syllables interchangeably. Let the set of syllables be denoted as $\mathcal{S} = \{S_1, S_2, \cdots S_M\}$, $M = 18$.

A percussion pattern is not well defined and varied definitions can exist. Here, we use a simplistic definition of a pattern, as a sequence of syllables. A pattern is defined as $P_k = [s_1, s_2, \cdots, s_{L_k}]$ where $s_k \in \mathcal{S}$ and L_k is the length of P_k. Though, for defining patterns, it is important to consider the relative and absolute durations of the constituent syllables, as well as the metrical position of the pattern in the tāl, we use a simple definition and leave a more comprehensive definition for future work. In this paper, we take a data driven approach to build a set of K query patterns, $\mathcal{P} = \{P_1, P_2, \cdots P_K\}$.

Given an audio recording $x[n]$, it is first transcribed into a sequence of time-aligned syllables, $T_x = [(t_1, s_1), (t_2, s_2), \cdots, (t_{L_x}, s_{L_x})]$, where t_i is the onset time of syllable s_i. The task of syllabic transcription has a significant analogy to connected word speech recognition using word models. Syllables are analogous to words and a percussion pattern to a sentence - a sequence of words. Finally, given a query pattern P_k of length L_k, we search for the pattern in the output syllabic transcription T_x, to retrieve the subsequences $p_k^{(n)}$ in T_x ($n = 1, \cdots, N_k$) that match the query, where

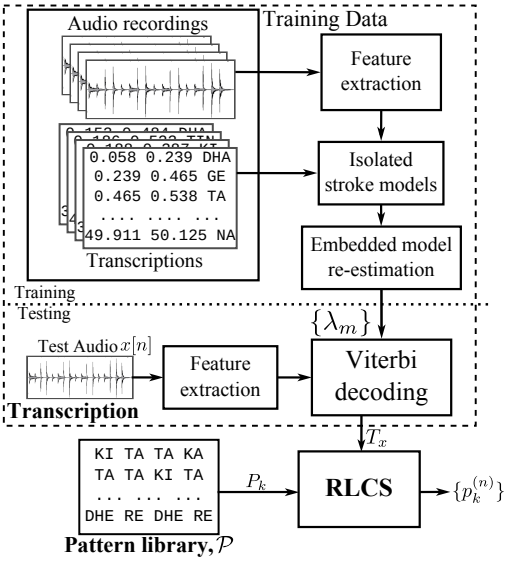

Figure 1: The block diagram of the approach

N_k is the number of retrieved matches for P_k. We use $p_k^{(n)}$ and the corresponding onset times from T_x to extract audio segments corresponding to the retrieved syllabic patterns. Syllabic transcription is often not exact and it can have common transcription errors such as insertions, substitutions and deletions, to handle which we need an approximate search algorithm.

3. DATASET

To evaluate our approach to percussion pattern discovery, we need a parallel corpus with time-aligned scores and audio recordings. These are useful both for building isolated stroke timbre models and for a comprehensive evaluation of the approach. We built a dataset comprising audio recordings, scores and time aligned syllabic transcriptions of 38 tabla solo compositions of different forms in *tīntāl* (a metrical cycle of 16 time units). The compositions were obtained from the instructional video DVD *Shades Of Tabla* by Pandit Arvind Mulgaonkar [1]. Out of the 120 compositions in the DVD, we chose 38 representative compositions spanning all the gharānās of tabla (Ajrada, Benaras, Dilli, Lucknow, Punjab, Farukhabad). The booklet accompanying the DVD provides a syllabic transcription for each composition. We used Tesseract [19], an open source Optical Character Recognizer (OCR) engine to convert printed scores to a machine readable format. The scores obtained from OCR were manually verified and corrected for errors, adding the the *vibhāgs* (sections) of the tāl to the syllabic transcription. The score for each composition has additional metadata describing the gharānā, composer and its musical form.

We extracted audio from the DVD video and segmented the audio for each composition from the full audio recording. The audio recordings are stereo, sampled at 44.1 kHz and have a soft harmonium accompaniment. A time aligned syllabic transcription for each score and audio file pair was obtained using a spectral flux based onset detector [3] followed by manual correction by the authors. The dataset contains about 17 minutes of audio with over 8200 syllables. The dataset is freely available for research purposes through a central online repository [2].

ID	Pattern	L	Count
1	DHE, RE, DHE, RE, KI, TA, TA, KI, NA, TA, TA, KI, TA, TA, KI, NA	16	47
2	TA, TA, KI, TA, TA, KI, TA, TA, KI, TA, TA, KI, TA, TA, KI, TA	16	10
3	TA, KI, TA, TA, KI, TA, TA, KI	8	61
4	TA, TA, KI, TA, TA, KI	6	214
5	TA, TA, KI, TA	4	379
6	KI, TA, TA, KI	4	450
7	TA, TA, KI, NA	4	167
8	DHA, GE, TA, TA	4	97

Table 2: Query Patterns, their ID (k), length (L) and the number of instances in the dataset (Total instances: 1425)

4. APPROACH

The block diagram in Figure 1 shows us the overall approach. It comprises three major steps: *building a set of query patterns, transcription, and search*. In the following sections, we describe each of these in detail.

4.1 Building a set of query patterns

A data driven approach is taken to create a set of query patterns of length $L = 4, 6, 8, 16$. These lengths were chosen based on the structure of tīntāl for different *layas* (tempo classes) [4, p. 126]. Using the simple definition of a pattern as a sequence of syllables, we use the scores of the compositions to generate all the L length patterns that occur in the score collection. We sort them by their frequency of occurrence to get an ordered set of patterns for each stated length. We then manually choose musically representative patterns from this ordered set of most commonly occurring patterns to form a set of query patterns. Table 2 shows the chosen patterns, their length and their count in the dataset, leading to a total of 1425 instances. We want a diverse collection of patterns to test if the algorithms generalize. Hence we choose patterns that have a varied set of syllables that have different timbral characteristics, like syllables that are harmonic (DHA), syllables played with a flam (DHE, RE) and syllables having bass (GE).

4.2 Transcription

Some bōls of tabla may be pronounced with a different vowel or consonant depending on the context, without altering the drum stroke [5]. Furthermore, the bōls and the strokes vary across different gharānās, making the task of transcription of tabla solos challenging. To model the timbral dynamics of syllables, we build an HMM for each syllable (analogous to a word-HMM). We use these HMMs along with a language model to transcribe an input audio solo recording into a sequence of syllables.

[1] http://musicbrainz.org/release/220c5efc-2350-43dd-95c6-4870dc6851f5

[2] http://compmusic.upf.edu/tabla-solo-dataset

The stereo audio is converted to mono, since there is no additional information in stereo channels. We use the MFCC features to model the timbre of the syllables. To capture the temporal dynamics of syllables, we add the velocity and the acceleration coefficients of the MFCC. The 13 dimensional MFCC features (including the 0^{th} coefficient) are computed from the audio with a frame size of 23.2 ms and a shift of 5.8 ms. We also explore the use of energy (as measured by the 0^{th} MFCC coefficient) in transcription performance. Hence we have two sets of features, MFCC_0_D_A, the 39 dimensional feature including the 0^{th}, delta and double-delta coefficients, and MFCC_D_A, the 36 dimensional vector without the 0^{th} coefficient.

Using the features extracted from training audio recordings, we model each syllable S_u using a 7-state left-to-right HMM $\{\lambda_u\}$, $1 \leq u \leq U(=18)$, including an entry and an exit non-emitting states. The emission density of each emitting state is modeled with a three component Gaussian Mixture Model (GMM) to capture the timbral variability in syllables. We experimented with higher number of components in the GMMs, but with little performance improvement. We use the time aligned syllabic transcriptions and the audio recordings in the parallel corpus to do an isolated HMM training for each syllable. We then use these HMMs further in an embedded model Baum-Welch re-estimation to get the final syllable HMMs.

Tabla solos are built hierarchically using short phrases, and hence some bōls tend to follow a bōl more often than others. In such a scenario, a language model can improve transcription. In addition to a flat language model with uniform unigram and transition probabilities, i.e. $p(s_1 = S_u) = 1/U$ and $p(s_{i+1} = S_v/s_i = S_u) = 1/U$, with $1 \leq u, v \leq U$ and i being the sequence index, we explore the use of a bigram language model learned from data.

For testing, we treat the feature sequence extracted from test audio file to have been generated from a first order time-homogeneous discrete Markov chain, which can consist of any finite length sequence of syllables. From the extracted feature sequence, we use the HMMs $\{\lambda_u\}$ and a syllable network constructed from the language model to do a Viterbi (forced) alignment, which aims to provide the best sequence of syllables and their onsets T_x. All the transcription experiments were done using the HMM Toolkit (HTK) [23].

4.3 Pattern Search

The automatically transcribed output syllable sequence T_x is used to search for the query patterns. Transcription is often inaccurate in both the sequence of syllables and in the exact onset times of the transcribed syllables. We need to handle both these errors in a pattern search task from audio. We primarily focus on the errors in syllabic transcription in this paper. We use the syllable boundaries output by the Viterbi algorithm, without any additional post processing. We can improve the output syllable boundaries using an onset detector [3], but we leave this task to future work.

There are three main kinds of errors in the automatically transcribed syllable sequence: Insertions (I), Deletions (D), and Substitutions (B). Further, the query pattern is to be searched in the whole transcribed composition, where several instances of the query can occur. Rough Longest Common Subsequence (RLCS) method is a suitable choice for such a case. RLCS is a subsequence search method that searches for roughly matched subsequences while retaining the local similarity [14]. We make further enhancements to RLCS to handle the I, D and B errors in transcription.

We use a modified version of the RLCS approach as proposed by Lin et al. [14] with changes proposed by Dutta et al. [8] to handle substitution errors. We propose a further enhancement to handle insertions and deletions, and explore its use in the current task. We first present a general form of RLCS and then discuss different variants of the algorithm.

Given a query pattern P_k of length L_k and a reference sequence (transcribed syllable sequence) T_x of length L_x, RLCS uses a dynamic programming approach to compute a score matrix (of size $L_x \times L_k$) between the reference and the query with a rough length of match. We can use a threshold on the score matrix to obtain the instances of the query occurring in the reference. We can then use the syllable boundaries in the output transcription and retrieve the audio segment corresponding to the match.

For the ease of notation, we index the transcribed syllable sequence T_x with i and the query syllable sequence P_k with j. We compute the rough and actual length of the subsequence matches similar to the way computed by Dutta et al. [8]. At every position (i, j), a syllable is included into the matched subsequence if $d(s_i, s_j) < \delta$, where $d(s_i, s_j)$ is the timbral distance between the syllables at positions i and j in the transcription and query, respectively. δ is the threshold distance below which the two syllables are said to be equivalent. The matrices of rough length of match ($\mathbf{C}$) and the actual length of match ($\mathbf{C}^a$) are updated as,

$$\mathbf{C}(i,j) = \mathbf{C}(i-1,j-1) + (1 - d(s_i, s_j)).\mathbb{1}_d \quad (1)$$

$$\mathbf{C}^a(i,j) = \mathbf{C}^a(i-1,j-1) + \mathbb{1}_d \quad (2)$$

where, $\mathbb{1}_d$ is an indicator function that takes a value of 1 if $d(s_i, s_j) < \delta$, else 0. The matrix $\mathbf{C}$ thus contains the length of rough matches ending at all combinations of the syllable positions in reference and the query. The rough length and an appropriate distance measure handles the substitution errors during transcription. To penalize insertion and deletion errors, we compute a "density" of match using two measures called the Width Across Reference (WAR) and Width Across Query (WAQ), respectively. The WAR ($\mathbf{R}$) and WAQ ($\mathbf{Q}$) matrices are initialized to $\mathbf{R}_{i,j} = \mathbf{Q}_{i,j} = 0$ when $i.j = 0$, and propagated as,

$$\mathbf{R}_{i,j} = \begin{cases} \mathbf{R}_{i-1,j-1} + 1 & d(s_i, s_j) < \delta \\ \mathbf{R}_{i-1,j} + 1 & d(s_i, s_j) \geq \delta,\ \mathbf{C}_{i-1,j} \geq \mathbf{C}_{i,j-1} \\ \mathbf{R}_{i,j-1} & d(s_i, s_j) \geq \delta,\ \mathbf{C}_{i-1,j} < \mathbf{C}_{i,j-1} \end{cases} \quad (3)$$

$$\mathbf{Q}_{i,j} = \begin{cases} \mathbf{Q}_{i-1,j-1} + 1 & d(s_i, s_j) < \delta \\ \mathbf{Q}_{i-1,j} & d(s_i, s_j) \geq \delta,\ \mathbf{C}_{i-1,j} \geq \mathbf{C}_{i,j-1} \\ \mathbf{Q}_{i,j-1} + 1 & d(s_i, s_j) \geq \delta,\ \mathbf{C}_{i-1,j} < \mathbf{C}_{i,j-1} \end{cases} \quad (4)$$

Here, $\mathbf{R}_{i,j}$ is the length of substring containing the subsequence match ending at the i^{th} and the j^{th} position of the reference and the query, respectively. $\mathbf{Q}_{i,j}$ represents a simi-

lar measure in the query. When incremented, $\mathbf{R}_{i,j}$ and $\mathbf{Q}_{i,j}$ are incremented by 1 similar to the way formulated by Lin et al. [14]. At the same time, the increment is done based on the conditions formulated by Dutta et al. [8].

Using the rough length of match ($\mathbf{C}$), actual length of match ($\mathbf{C}^a$), and width measures ($\mathbf{R}$ and $\mathbf{Q}$), we compute a score matrix $\boldsymbol{\sigma}$ that incorporates penalties for substitutions, insertions, deletions, and additionally, the fraction of the query matched.

$$\boldsymbol{\sigma}_{i,j} = \begin{cases} \left[\beta \cdot f\left(\frac{\mathbf{C}_{i,j}}{\mathbf{R}_{i,j}}\right) + (1-\beta)\cdot f\left(\frac{\mathbf{C}_{i,j}}{\mathbf{Q}_{i,j}}\right)\right]\cdot\frac{\mathbf{C}(i,j)}{L_k} & \text{if } \frac{\mathbf{C}^a(i,j)}{L_k} \geq \rho \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where $\boldsymbol{\sigma}_{i,j}$ is the score for the match ending at the i^{th} and the j^{th} position of the reference and the query, respectively. f is a warping function for the rough match length densities $\frac{\mathbf{C}_{i,j}}{\mathbf{R}_{i,j}}$ in the reference and $\frac{\mathbf{C}_{i,j}}{\mathbf{Q}_{i,j}}$ in the query. The parameter β controls their weights in the convex combination for score computation. The term $\frac{\mathbf{C}^a_{i,j}}{L_k}$ is the fraction of the query length matched and is used for thresholding the minimum fraction of the query to be matched.

Starting with all combinations of i and j as the end points of the match in the reference and the query, respectively, we perform a traceback to get the starting points of the match. RLCS algorithm outputs a match when the score is more than a score threshold ψ. However, with a simple score thresholding, we get multiple overlapping matches, from which we select the match with the highest score. If the scores of multiple overlapping matches are equal, we select the ones that have the lowest width (WAR). This way, we obtain a match that has the highest score density. We use these non-overlapping matches and the corresponding syllable boundaries to retrieve the audio patterns.

4.3.1 Variants of RLCS

The generalized RLCS provides a framework for subsequence search. The parameters ρ, β, ψ and δ can be tuned to make the algorithm more sensitive to different kinds of transcription errors. The variants we consider here use different distance measures $d(s_i, s_j)$ in Eqn (1) to handle substitutions and different functions $f(.)$ in Eqn (5) to handle insertions and deletions. We explore these variants for the current task and evaluate their performance.

In a default RLCS configuration (RLCS$_0$), we only consider exact syllable matches. We set $\delta = 1$ and use a binary distance metric based on the syllable label, i.e. $d(s_i, s_j) = 0$ if $s_i = s_j$, and 1 otherwise. Further, an identity warping function, f(y) = y is used.

The rough length match densities can be transformed using a non-linear warping function to penalize low density values more than the higher ones, leading to another variant of RLCS (RLCS$_\kappa$). In this paper, we only explore warping functions of the form,

$$f(y) = \frac{e^{\kappa y} - 1}{e^{\kappa} - 1} \quad (6)$$

where $\kappa > 0$ is a parameter to control warping, larger values of κ lead to more deviation from an identity transformation. RLCS$_0$ is a limiting case of RLCS$_\kappa$ when $\kappa \to 0$.

We hypothesize that the substitution errors in transcription are due to the confusion between timbrally similar syllables. A timbral similarity (distance) measure between the syllables can thus be used to make an RLCS algorithm robust to specific kinds of substitution errors. In essence, we want to disregard and give a greater allowance for substitutions between timbrally similar syllables during RLCS matching. Computing timbral similarity is a wide area of research and has many different proposed methods [17], but we restrict ourselves to a basic timbral distance measure: the Mahalanobis distance between the cluster centers obtained using a K-means clustering of MFCC features (with 3 clusters) from isolated audio examples of each syllable [2]. We call this variant of RLCS as RLCS$_\delta$ and experiment with different thresholds δ. For better reproducibility of the work in this paper, an implementation of the different variants of RLCS described is available [3].

5. EXPERIMENTS AND RESULTS

We experiment with different sets of features and language models for transcription. With the best performing transcription configuration, we experiment with different RLCS variants and report their performance. We first describe the evaluation measures used in this paper.

5.1 Evaluation measures

We use the ground truth time aligned syllabic transcriptions to evaluate both the transcription and pattern search algorithms. We evaluate transcription performance using the measures often used in speech recognition, Correctness (Corr.) and Accuracy (Accu.). Given the ground truth transcription T_x^* of length N, the transcribed sequence T_x, and the number of insertions, deletions and substitutions as N_I, N_D, and N_B, respectively, we compute Corr. $= (N - N_D - N_B)/N$ and Accu. $= (N - N_D - N_B - N_I)/N$. The Correctness measure penalizes deletions and substitutions, while Accuracy measure additionally penalizes insertions.

For pattern retrieval, we don't evaluate the accuracy of boundary segmentation. However, we call a retrieved pattern from RLCS as *correctly retrieved* if it has at least a 70% overlap with the pattern instance in ground truth. To evaluate pattern search performance, we use the standard information retrieval measures precision (the ratio between the number of correctly retrieved patterns and all retrieved patterns) and recall (the ratio between number of correctly retrieved patterns and the patterns in the ground truth). The harmonic mean of precision and recall, called the F-measure is also reported.

5.2 Results and Discussion

The transcription results shown in Table 3 are the mean values in a leave-one-out cross validation over the dataset. We experimented with the two different MFCC features (MFCC_D_A and MFCC_0_D_A) and two language models (a flat model and a bigram learnt from data). Overall, we see a best Accuracy of 53.13%, which justifies the use of a robust approximate string search algorithm for pattern retrieval. The use of a bigram language model learned from data improves the transcription performance. We see that

[3] `http://compmusic.upf.edu/ismir-2015-tabla`

	Feature	Corr.	Accu.
Flat language model	MFCC_D_A	64.07	45.01
	MFCC_0_D_A	64.26	49.27
Bigram language model	MFCC_D_A	**65.53**	49.97
	MFCC_0_D_A	**66.23**	**53.13**

Table 3: Transcription results showing the Correctness (Corr.) and Accuracy (Accu.) measures (in percentage) for different features and language models. In each column, the values in bold are statistically equivalent to the best result (in a paired-sample t-test at 5% significance levels).

the Accuracy measure is lower than the Correctness measure, which shows that there are a significant number of insertion errors in transcription. We use the output transcriptions from the best performing combination (MFCC_0_D_A and a bigram language model) to report the performance of the RLCS variants.

To form a baseline for string search performance with the output transcriptions, we used an exact string search algorithm and report its performance in Table 4 (shown as Baseline). We see that the baseline has a precision that is similar to transcription performance, but a very poor recall leading to a poor F-measure.

To establish the optimum parameter settings for RLCS, we performed a grid search over the values of β, ρ and ψ with RLCS_0 . β and ψ are varied in the range 0 to 1. To ensure that the minimum length of the pattern matched is at least 2, we varied ρ between $1.1/\min(L_k)$ and 1.

β is the convex sum parameter for the contribution of the rough match length density of the reference and the query towards the final score. With increasing β, we give more weight to the reference length ratio, allowing more insertions. We observed a poor true positive rate with larger β, and hence we validate the observation that insertion errors contribute to a majority of transcription errors.

The best average F-measure over all the query patterns in an experiment using RLCS_0 is reported in Table 4. We see that RLCS_0 improves the recall, but with a lower precision and an improved F-measure, showing that the flexibility in approximate matching provided by RLCS comes at the cost of additional false positives. The values of ρ, β and ψ that give the best F-measure are then fixed for all subsequent experiments to compare the performance of the proposed RLCS variants.

It is observed that the patterns composed of smaller repetitive patterns (and hence having ambiguous boundaries) result in a poor precision (e.g. P_2 and P_3 in Table 2 with a precision of 0.108 and 0.239, respectively). P_1 in Table 2, on the contrary, has non-ambiguous boundaries leading to a good precision of 0.692. The effect of the length of a pattern on precision is also evident. Small patterns (with $L = 4$) that have non-ambiguous boundaries (e.g. P_8 in Table 2 with a precision of 0.384) have a poor precision as compared to longer patterns with non-ambiguous boundaries (e.g. P_1 in Table 2). The reason for this is that the smaller patterns are more prone to errors as the search algorithm has to match a lower number of syllables.

The results with other variants of RLCS are also reported in Table 4. The results from RLCS_δ show that the use of a timbral syllable distance measure with higher threshold δ further improves the recall, but with a much lower precision and F-measure. Although we find matches that have substitution errors using the distance measure, we retrieve additional matches that do not have substitution errors contributing to additional false positives. On the contrary, using a non-linear warping function $f(.)$ in RLCS_κ improves the precision with a higher value of κ. The penalties on matches with higher number of insertions and deletions is high and they are left out, leading to good precision at the cost of recall. We observe that both the above mentioned variants improve either precision or recall at the cost of the other measure. They need further exploration with better timbral similarity measures to be combined in an effective way to improve the search performance.

Variant	Parameter	Precision	Recall	F-measure
Baseline	-	0.479	0.254	0.332
RLCS_0	$\delta = 1$	0.384	0.395	0.389
RLCS_δ	$\delta = 0.3$	0.139	0.466	0.214
RLCS_δ	$\delta = 0.6$	0.0837	0.558	0.145
RLCS_κ	$\kappa = 1$	0.412	0.350	0.378
RLCS_κ	$\kappa = 4$	0.473	0.268	0.342
RLCS_κ	$\kappa = 7$	0.482	0.259	0.336
RLCS_κ	$\kappa = 9$	0.481	0.258	0.335

Table 4: Performance of different RLCS variants using the best performing parameter settings for RLCS_0 ($\rho = 0.875$, $\beta = 0.76$ and $\psi = 0.6$).

6. SUMMARY

We addressed the unexplored problem of a discovering syllabic percussion patterns in Tabla solo recordings. The presented formulation used a parallel corpus of audio recordings and syllabic scores to create a set of query patterns, that were searched in an automatically transcribed (into syllables) piece of audio. We used a simplistic definition of a pattern and explored RLCS based subsequence search algorithm, using an HMM based automatic transcription. Compared to a baseline, we showed that the use of approximate string search algorithms improved the recall at the cost of precision. Additionally, proposed variants improved either the precision or recall, but do not provide a significant improvement in the F-measure over the basic RLCS.

For future work, we aim to improve syllable boundaries output by transcription using onset detection. Inclusion of the rhythmic information can be an interesting aspect in defining and discovering percussion patterns, and will help in comprehensively evaluating the task of pattern discovery. The next steps would be to incorporate better timbral similarity measures and inclusion of segment boundaries into the RLCS algorithm that effectively combines the proposed variants.

Acknowledgments

This work is partly supported by the European Research Council under the European Union's Seventh Framework Program, as a part of the CompMusic project (ERC grant agreement 267583). The authors thank Pandit Arvind Mulgaonkar for sharing the DVD of Tabla solo recordings.

7. REFERENCES

[1] A. Anantapadmanabhan, A. Bellur, and H. A. Murthy. Modal analysis and transcription of strokes of the mridangam using non-negative matrix factorization. In *Proc. of the 38th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 181–185, Vancouver, Canada, May 2013.

[2] J. J. Aucouturier and F. Pachet. Music similarity measures: What's the use? In *Proc. of 3rd International Conference on Music Information Retrieval*, pages 157–163, Paris, France, 2002.

[3] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and Mark B. Sandler. A tutorial on onset detection in music signals. *IEEE Transactions on Speech and Audio Processing*, 13(5):1035–1047, Sept 2005.

[4] S. Beronja. *The Art of the Indian tabla*. Rupa and Co. New Delhi, 2008.

[5] A. Chandola. *Music as Speech: An Ethnomusicolinguistic Study of India*. Navrang, 1988.

[6] P. Chordia. Segmentation and recognition of tabla strokes. In *Proc. of the 6th International Conference on Music Information Retrieval (ISMIR)*, pages 107–114, London, UK, September 2005.

[7] P. Chordia, A. Sastry, and S. Şentürk. Predictive tabla modelling using variable-length markov and hidden markov models. *Journal of New Music Research*, 40(2):105–118, 2011.

[8] S. Dutta and H. A. Murthy. A modified rough longest common subsequence algorithm for motif spotting in an alapana of carnatic music. In *Proc. of the 20th National Conference on Communications (NCC)*, pages 1–6, Kanpur, India, February 2014.

[9] O. Gillet and G. Richard. Automatic labelling of tabla signals. In *Proc. of the 4th International Conference on Music Information Retrieval (ISMIR)*, Baltimore, USA, October 2003.

[10] R. S. Gottlieb. *Solo Tabla Drumming of North India: Its Repertoire, Styles, and Performance Practices*. Motilal Banarsidass Publishers, 1993.

[11] X. Huang and L. Deng. An overview of modern speech recognition. In N. Indurkhya and F. J. Damerau, editors, *Handbook of Natural Language Processing*, Chapman & Hall/CRC Machine Learning & Pattern Recognition, pages 339–366. Chapman and Hall/CRC, 2nd edition, February 2010.

[12] D. Hughes. No nonsense: the logic and power of acoustic-iconic mnemonic systems. *British Journal of Ethnomusicology*, 9(2):93–120, 2000.

[13] J. Kuriakose, J. C. Kumar, P. Sarala, H. A. Murthy, and U. K. Sivaraman. Akshara transcription of mrudangam strokes in carnatic music. In *Proc. of the 21st National Conference on Communication (NCC)*, Mumbai, India, February 2015.

[14] H. Lin, H. Wu, and C. Wang. Music matching based on rough longest common subsequence. *Journal Information Science and Engineering*, 27(1):95–110, 2011.

[15] M. Miron. Automatic Detection of Hindustani Talas. Master's thesis, Music Technology Group, Universitat Pompeu Fabra, 2011.

[16] T. Nakano, J. Ogata, M. Goto, and Y. Hiraga. A drum pattern retrieval method by voice percussion. In *Proc. of the 5th International Conference on Music Information Retrieval (ISMIR)*, pages 550–553, October 2004.

[17] F. Pachet and J. J. Aucouturier. Improving timbre similarity: How high is the sky. *Journal of negative results in speech and audio sciences*, 1(1):1–13, 2004.

[18] A. D. Patel and J. R. Iversen. Acoustic and perceptual comparison of speech and drum sounds in the north indian tabla tradition: An empirical study of sound symbolism. In *Proc. of the 15th International Congress of Phonetic Sciences (ICPhS)*, pages 925–928, Barcelona, Spain, 2003.

[19] R. Smith. An Overview of the Tesseract OCR Engine. In *Proc. of the Ninth International Conference on Document Analysis and Recognition (ICDAR)*, volume 2, pages 629–633, Washington, DC, USA, 2007.

[20] A. Srinivasamurthy, R. Caro, H. Sundar, and X. Serra. Transcription and recognition of syllable based percussion patterns: The case of Beijing Opera. In *Proc. of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, pages 431–436, Taipei, Taiwan, October 2014.

[21] E. Tsunoo, G. Tzanetakis, N. Ono, and S. Sagayama. Beyond timbral statistics: Improving music classification using percussive patterns and bass lines. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):1003–1014, 2011.

[22] R. Typke, F. Wiering, and R. C Veltkamp. A survey of music information retrieval systems. In *Proc. of the 6th International Conference on Music Information Retrieval (ISMIR)*, pages 153–160, London, UK, September 2005.

[23] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland. *The HTK Book, version 3.4*. Cambridge University Engineering Department, Cambridge, UK, 2006.