iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 Issue: IV Month of publication: April 2019

DOI: https://doi.org/10.22214/ijraset.2019.4617

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

Indian Classical Music Swara/Note Transcription System

Kanviksha Maujeker[1], Sangam Borkar [2]
[1]*Student, Department of Electronics and Telecommunication, Goa College of Engineering, Farmagudi Goa, India*
[2]*Department of Electronics and Telecommunication, Goa College of Engineering, Farmagudi Goa, India*

Abstract: "A trained Hindustani musician is capable of perceiving the notes based on the lead voice but a novice person is unable to decode the notes. Sometimes even the trained musician fails to detect transition notes. This necessitates the development of an automated note transcription system". The automatic system detects and generates the notes present in the music file. In Indian classical music, Harmonium and drone (tanpura) are the most important instruments which accompany vocal signal. For training, purpose tanpura is the only instrument which provides base music and helps to sing correct notes. In this paper, we first prepare a database both for training and testing phase assuming a female vocal monophonic signal with a sampling frequency of 44100 Hz. Recording of all the notes/Swara sung individually by the female singer is used for the training phase. There is a total of 13 audio files for training the system. The testing dataset includes Alankar audio files which have different placement and sequence of the notes/swara or "Aakar (melismatic singing on the syllable /a/)". The system is also tested on Bandish.wav file. For Pitch extraction from both the datasets we have used Autocorrelation function and the classifier with the highest accuracy is chosen using the Classifier Learner Application from MATLAB software for classification. A generalized comparison between Autocorrelation function and the YIN algorithm was made. A trained model in Linear SVM gives the highest accuracy of 93.6%. This system doesn't avoid any transition notes to achieve melody tolerance since the extracted pitch ranges already has incorporated this parameter and can further be used for classification. ANN Classifier has already been used in other paper, for training features like "Mel-frequency cepstral coefficients (MFCC)", Pitch, Jitter and Shimmer for vocal as well as non-vocal signal and accuracy for vocal by their approach achieves 85.59%. By quantitative evaluation, it has been validated that the system proposed in this paper demonstrates superior performance and as such can be used in future applications in music processing.
Keywords: Swara, Autocorrelation, SVM, Alankar, Bandish, Aakar, Monophonic, Pitch.

I. INTRODUCTION

This project is based on the design of a transcription system which transcribes the swara/notes present in Indian classical music. Indian classical music is in raga and tala format [1]. The term raga describes the set of permitted Swara's or notes and characteristic phrases, and tala defines the rhythm. The combination of notes and its ornamentation characterizes raga [1]. The seven Swara's or notes are Shadja (Sa), Rishabh (Re), Gandhar (Ga), Madhyam (Ma), Panchama (Pa), Dhaivat (Dha) and Nishad (Ni) [1].
Out of these seven notes, Shadja and Panchama [1] have no variation and are called Achala or immovable note [1]. However, the rest of the notes exhibit microtonal variation, also called vikrit form [1]. Rishabh, Gandhar, Dhaivat, and Nishad have Komal or flat version [1], moved below their natural place and only Madhyam has Tivra or sharp version [1], higher than the /natural one. Taking these variations into account, there are twelve notes in classical music. It is important to know what type of audio signal is taken into account. Following are the types of musical textures [2][3].

A. Monophony [3]

Monophonic music texture is nothing but a one sound or single note and music only contains a melody line with no harmony. It is usually played by one person their own or by many people can play the same melody.

B. Heterophony [3]

Heterophony music is formed by real-time recording by the different number of singers of different versions of the same tune. It contains parts of the different music of the same melody.

C. Polyphony [3]

Polyphonic music is based on counterpoint which is the simultaneous performance of multiple melodies or tunes that are distinct from each other in notes and rhythm.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

D. Homophony [3]

Homophony is defined as a texture in which we encounter most often multiple voices in which one it can consist of a single dominating melody and is accomplished by chords.

Several mono pitch algorithms based on time domain such as zero crossing and Average magnitude difference function has been proposed. One of the fundamental algorithms in this class is an auto-correlation function (ACF) [2][21], which is further modified to YIN [8] algorithm which enhances its performance.

Autocorrelation [2] is the simplest method to implement. A correlation function [2] is a measure of the degree of similarity between two signals. The autocorrelation [2] measures how well the input matches with a time-shifted version of itself. The pitch is simply inverse of the pitch period and we can find pitch period from the peaks of the autocorrelation function [2].

In the frequency domain, Soft Fourier transforms STFT converts the signal to magnitude spectra [4]. In such representation, the pitch corresponds to peaks in the power spectrum. Plotting techniques being utilized are spectrogram [5][10] and correlogram [7].

Pitch tracking: DNN (Deep Neural Network) [4] [6] and UPDUDP [4] (unbroken pitch determination using dynamic programming) algorithms for separation of singing voice from music and for pitch tracking respectively. KNN (K- Nearest Neighbor) [6] algorithm is trained on a large database of piano tones. Performance accuracy of KNN pitch classifier is examined with 3 different distance measure as well as different values of *K*. Note Quantization [3] [8]is also used to develop vocal training system based on MIDI [10] (Musical Instrument Digital Interface) signal. Artificial Neural Networks (ANNs) [9][19] is used as classifiers to capture the relevant information from the songs [19].

In this paper, we have focused on the monophonic vocal signal of a well-trained female singer in Indian classical music for creating a database. The database focuses on the basic lessons taught to music students like singing affirmative Notes/Swara, Alankar/Ornamentation, Aakar, and Bandish. There are different kinds of Bandish audio files available, but we recorded only one for testing our system. We also tested our system on different Aakar (melismatic singing on the syllable /a/) [6] audio files. Despite the YIN [8] algorithm is a modification of Autocorrelation function and gives optimized output and prevents error [4], our approach uses the simplest version i.e. the most basic, Autocorrelation function neglecting all the complexities. For classification of pitch levels, Linear SVM classifier was chosen from all other classifier models depending on their accuracy to get transcribed output.

II. RELATED WORK

Many methods for Pitch extraction and Pitch tracking have been proposed. Most of the work revolves around the tonic pitch feature extraction Tonic [2] is one of the integral parts of Indian music. It is a base pitch for entire rag or melody [2]. In one paper to overcome the computational complexity and slightly sluggish algorithms for tonic extraction, Harmonic Product Spectrum (HPS) [2] [12] approach was used to exploit the harmonic nature of music for pitch extraction[2]. The production features are derived from the acoustic signal [14], using the signal processing methods such as short-time Fourier transform (STFT) [2][14], linear- prediction analysis [14], and zero-frequency filtering [14]. The music and vocal regions are separated by music-source separation, using STFT with different types of windowing techniques [14] such as Blackman, Hamming, and Kaiser windows. Proposed method performs better than the ACF [2] and YIN algorithm [2]. It was also noted that as frame size changes the accuracy of tonic extraction varies [2].

Hardware system using SHARC Processor [9] to develop an adept vocal training system that perceives and appraises the pitches of both the amateur vocalist [9] and the professional vocalist and displays inaccuracy of the amateur's voice. Subharmonic-to-Harmonic Resolution (SHR) [9] [12]algorithm is used for pitch detection as it deals with alternate pulse cycles or subharmonics in speech [9]. The first step in this algorithm is taking autocorrelation of the signal. Classification of notes is done by Note quantizer based on MIDI (Musical Instrument Digital Interface) signal [8] [9].

An improved method for the detection of the vocal pitch for musical transcription [11] by eliminating harmonics present in the voice signal and then carrying out filtering techniques. The resulting signal is divided into a number of segments based on the input sequence given by a metronome input [11]. The pitch detection process is done using the LPC (Linear Predictive Coding) [11] [14] method. This method uses spectral analysis for the detection of the pitch after finding the main envelope of the vocal signal for a given window duration [11].

Another paper proposes a novel and e□ective two-stage approach to singing pitch extraction [4], which involves singing voice separation and pitch tracking for monaural polyphonic audio music. The first stage extracts singing voice from the songs by using deep neural networks [4] for singing voice separation in a supervised setting. After extracting the singing voice from mixture music, the pitch is determined by using a robust pitch tracking [4] method based on dynamic programming[4]. The input to the DNN [6] [4], consisting of magnitude spectra of the mixture music which is performed by using short time Fourier transform (STFT) [4] [11]

[12][13][14]. The time-domain signals of estimated magnitude spectra [4] are reconstructed by using inverse short time Fourier transform (ISTFT) [4], which uses the phase information obtained from the original input signals. Once the vocal is extracted from the mixture; there is a need to perform pitch tracking [4] to extract the vocal pitch. In this work, authors proposed a new adaptive approach [4] of unbroken pitch determination using dynamic programming (UPDUDP) [4] which is a robust pitch tracking method based on dynamic programming which considers both periodicity and smoothness to derive the final optimum path [4].

An efficient approach to transcription of monophonic melodies [6] from a raw acoustic signal is presented. Two different instancebased pitch classification [6] methods are proposed, the choice of which depends on the size of the available training database. In the first method, the conventional K-Nearest Neighbour algorithm [6] is trained on a large database of piano tones and employed for monophonic pitch detection. For cases where the training database contains only one sample from each possible note, a two-step algorithm, combining semi-KNN [6] pitch candidate selection and note sequence tracking, is suggested. It is demonstrated that in the abundance of training data, the KNN algorithm [6] [15] along with a proper choice of the distance measure and K, yields high-performance accuracy. Nevertheless, the authors report mistakenly transcribed vocal notes as a main source of error and the inaccurate vocal detection as the major limitation of the system performance. In the process of optimization, transition notes are often neglected and eliminated using thresholding [1]. In our approach, these notes are grouped with other notes to form a final note. System accuracy diminishes mainly due to limited training dataset [15]. The initial stage of transcription system follows Ratio to Tonic [2] measure is to find ratio between different notes to the tonic[1] and then melody contour [15] is plotted against Cent scale [1] (Note melody value in Cent) achieved from ratio to tonic values [1], significantly higher performance is achieved when evaluating the same transcription algorithm on a training dataset containing individual fundamental frequency ranges of notes. We, therefore, identify a need for improving the system and design it using autocorrelation function, the simplest technique for pitch extraction and use Classifier Learner Application from MATLAB Software to choose the best classifier with the highest accuracy for pitch classification.

III.PROPOSED ALGORITHM

Our objective is to design a Transcription system for Indian classical music. Considerations made are as follows:

1) Monophonic texture
2) Female Voice (Drone set to Note A)
3) Sampling Frequency: 44100 Hz
4) Frame Size: 40 msec

The block diagram below shows the steps for generalized Automatic Music Transcription System [3] [13].

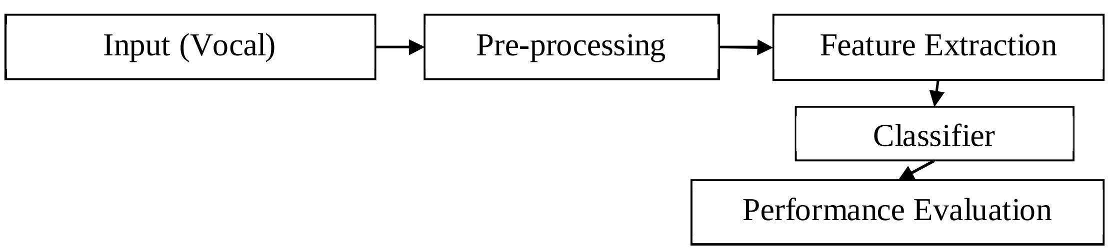

Fig. 1 Block diagram of the Generalized Automatic Music Transcription system

A. Autocorrelation Function for Pitch Estimation

This paper uses the most basic approach for pitch estimation called Autocorrelation function [3]. In this method, the signal is delayed and multiplied with itself for different time lags. The autocorrelation function has peaks at the lags in which the signal is self-similar (when it is periodic).

1) Compare Autocorrelation and YIN Algorithm

a) This step was done to just have a general comparison between the fundamental frequencies f0 frame by frame using autocorrelation and YIN [8] algorithm.
b) Firstly the code for autocorrelation was run on a soundtrack of a clarinet which is a musical instrument whose fundamental frequency f0 (Pitch) was known using the YIN algorithm.
c) Using the YIN algorithm the fundamental frequency f0/pitch of the whole soundtrack is 312.7978 Hz.
d) Using Autocorrelation fundamental frequency f0 of the frames ranges between 310.4 Hz to 315 Hz. Therefore, we conclude that the YIN algorithm gives the optimized result.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

B. Training and Testing Dataset

For preparing both the datasets we hired a Female Singer who is well trained in Indian Classical Music. Recording of all the audio file was done in a Professionally accessorized Music Recording Studio.

Every singer has his/her reference note which they need to sing correctly. Therefore, the singer was told to wear headphones onto which drone music set on Note A (Pandri 6) was played. Audio files don't include drone music or digital interference, the only raw vocal was recorded. The time interval for singing a note wasn't planned, a note can be of any duration depending upon the singer's capability to sing in one breath. As mentioned above in the considerations, the sampling frequency was set to 44100 Hz mono signal at 16 bits/sec.

Training Dataset: Training dataset will comprise of 250 pitch samples extracted using autocorrelation function on each vocal Note/Swara sung by the trained singer. There will be a total of (13 Notes/Swara *250) 3250 samples. Fig. 2. shows a block diagram for the training stage.

Testing Dataset: For testing, dataset different kind of ornamentation/Alankar(Aakar) and Bandish sung by the same singer are evaluated. Each .wav file is run through an Autocorrelation function to estimate frame by frame pitch.

C. MATLAB (Classification learner Application)

Training Database was used to train a number of classifiers available in the MATLAB Classifier learner Application. It trains models to classify data using supervised machine learning Statistics and Machine Learning Toolbox.

The classifier with the highest Accuracy was chosen for the testing phase. SVM (Support Vector Machine) [18] Linear classifier had the highest accuracy of 93.6%, therefore this model is chosen for the testing stage. Fig. 3. Shows the block diagram for the testing stage.

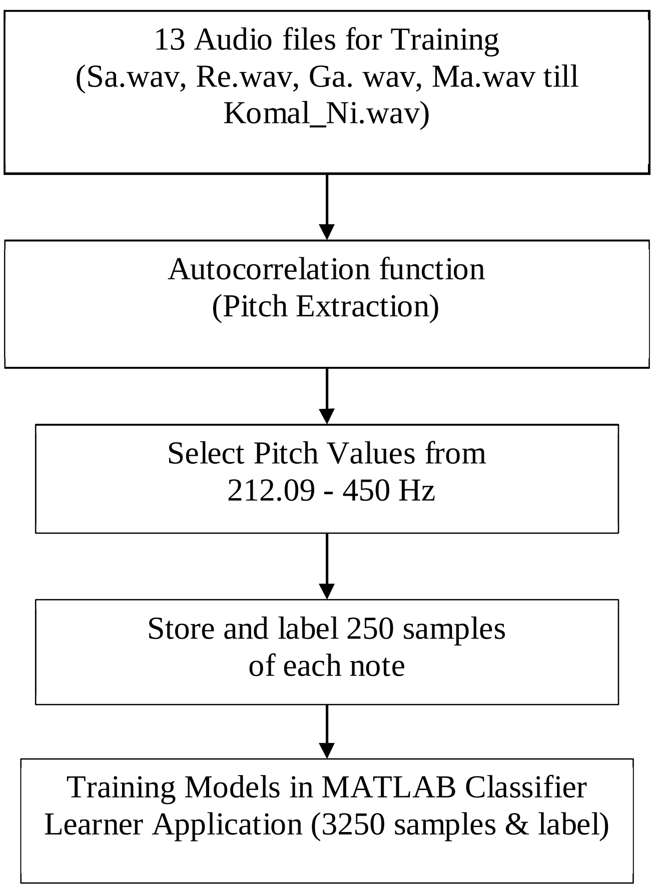

Fig.2 Block diagram for the training stage

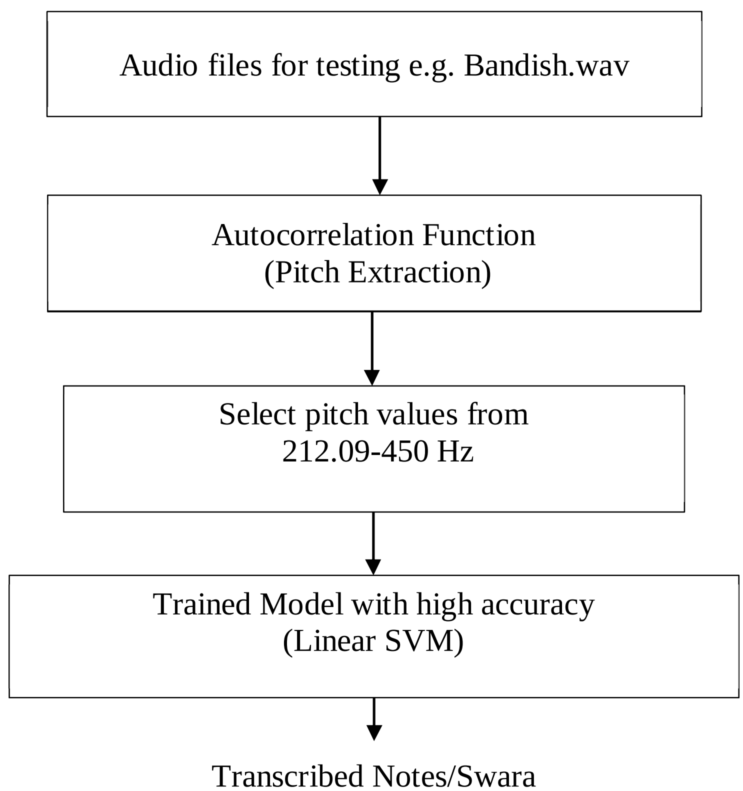

Fig. 3 Block diagram for the testing stage

IV.RESULTS

Table. I summarize the extracted pitch ranges of every note/swara vocal during the training stage. These ranges are from 250 samples of each note.

Fig. 4 shows some of the samples of extracted pitch and its corresponding Note/swara label. Each Note has 250 samples of the extracted pitch.

Total number of notes/swara 'n' =13

A total number of samples:

$$n \times 250 = 325$$

These 3250 samples are used to train the Linear SVM classification model. Fig. 5 shows the accuracy level of different types of models trained and the one with the highest accuracy is exported and testing is carried out.

Confusion Matrix of the Linear SVM Model is shown in Fig. 6. This shows that 41% of samples of Kom_Dha and 77% of samples of Kom_Ga are correctly predicted. This problem occurs due to the slight overlap of Kom_Dha and Kom_Ga pitch ranges used for training.

This model is exported and ran on files called Akar1.wav and Bandish.wav file from testing dataset. Fig. 7 shows two MATLAB windows, at the left hand there is transcript output of Akar1.wav and at the right the shows the transcript output of Bandish.wav file

TABLE I
PITCH RANGES

NOTE/SWARA	EXTRACTED PITCH (Hz)
Sa	212.09-222.72
Re	245-252
Ga	270.55-279
Ma	290.13-300
Pa	329.10-331.57
Dha	370.58-380.17
Ni	408.33-420
Sa#	436.63-450
Komal Re (Kom_Re)	230.89-237.09
Komal Ga (Kom_Ga)	249.15-264.07
Tivra Ma	306.25-315
Komal Dha (Kom_Dha)	259.41-267.27
Komal Ni (Kom_Ni)	380.17-390.26

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

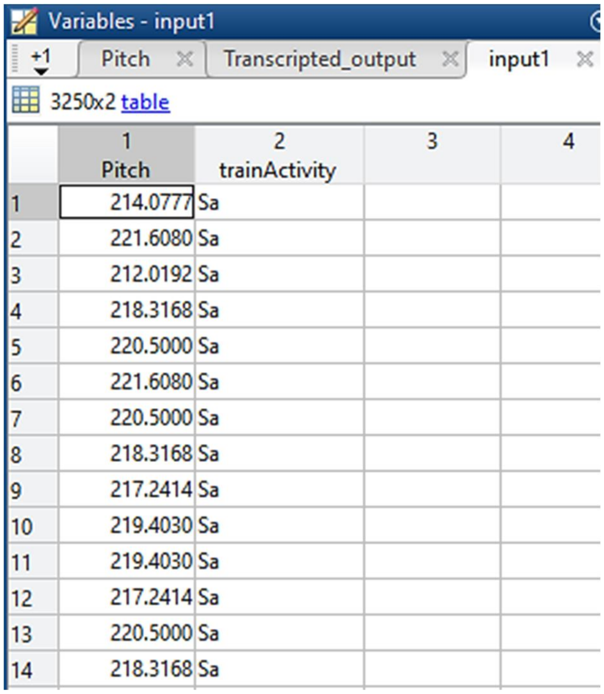

Fig. 4 Training Data including pitch values and notes/swara

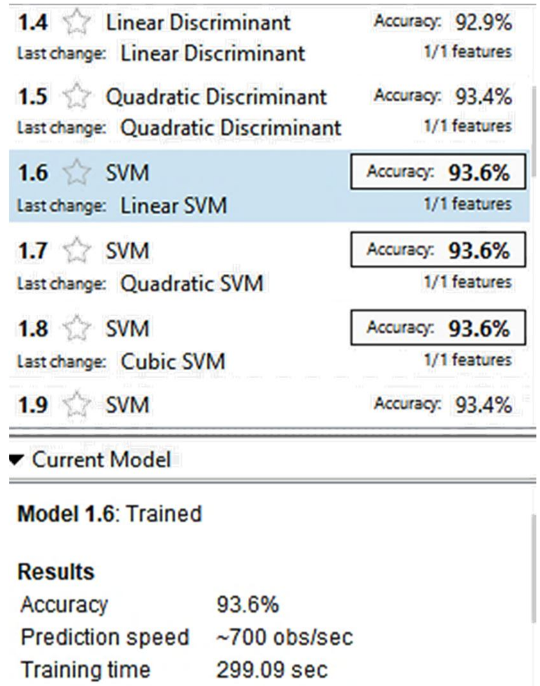

Fig. 5 The accuracy of the trained model

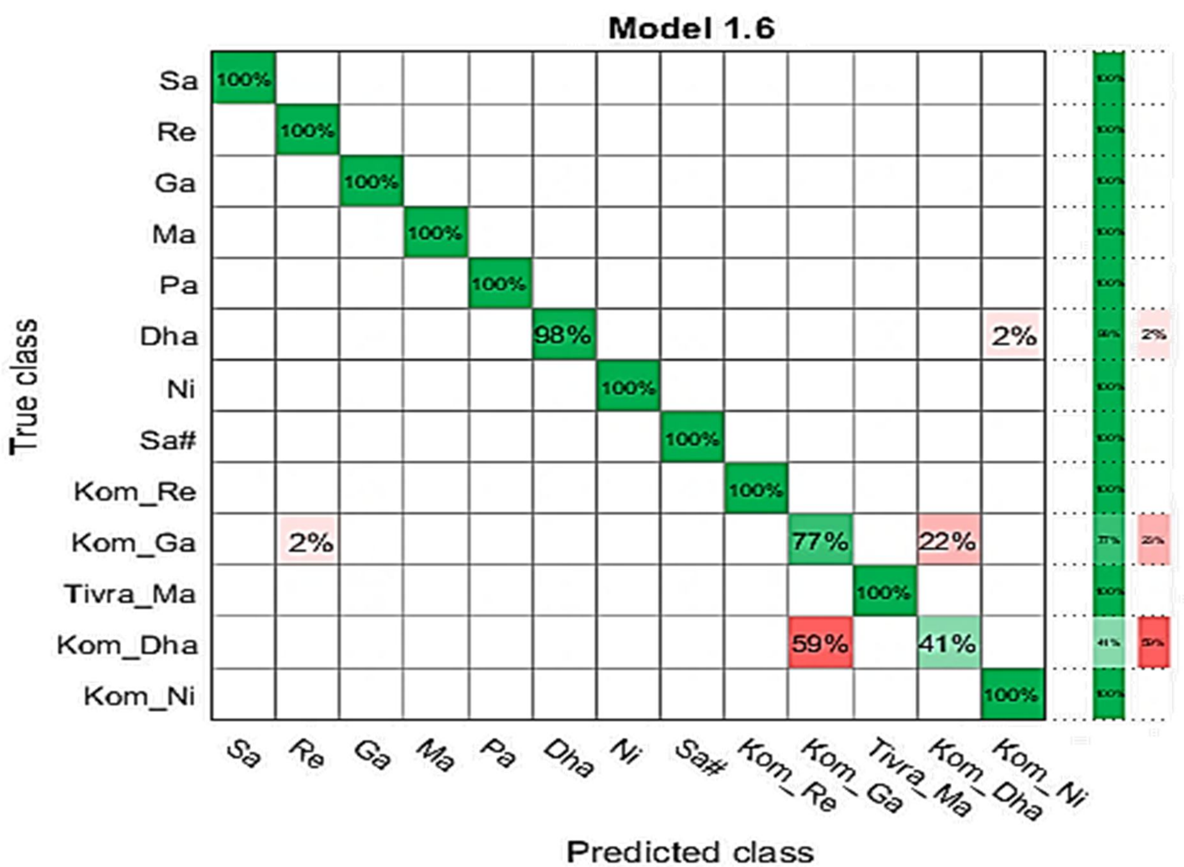

Fig. 6 Confusion Matrix for SVM Model

The red color box is shown in Fig. 7 at the left shows some of the transcript Note/Swara detected from Akar1.wav file. As can be seen in between “Sa” and “Re” note/swara there exist one more note called “Kom_Re” (Komal Re). This” Kom_Re” is a transition note from “Sa” to “Re”. These Transition notes are important but often neglected.

Again, the trained model was run on Bandish.wav file which is the second basic Indian classical music lesson taught to the students. Fig. 7 shows the transcript output Bandish.wav file. (550*1) and (2682*1) in the blue boxes present in Fig. 7 are the sizes of the transcript output, which means Akar1.wav has 550 notes and Bandish.wav file have 2682 notes. In a practical scenario, while listening we cannot detect all these notes.

To optimize the result, we carried out two more steps as follow

1) After extracting pitch values from the testing files, make groups of 20 samples from the starting till the end.
2) Calculate the mean value of each group and store the processed pitch values.
3) Then follow the same steps i.e. passing these values in the trained model for classification.

The steps mentioned to optimize the result are implemented on Bandish.wav file. To compare the results before and after optimization we discussed the Bandish.wav file with the professional music teacher, according to them, it is based on “Raag Maru Bihag” and from listening to this track they can detect 150 notes roughly. Not only the number of notes/swara is important but pitch tracks also need to be done correctly.

The output after modification in the system is shown in Fig. 8. In Fig. 8(a), the plot of the Bandish.wav track has its amplitude on the y-axis and the number of samples on the x-axis. Fig. 8(b), the plot of the transcribed stem plot of the audio file. The x-axis shows the number of notes and the y-axis shows the names of the notes/swara.

The Histogram of the audio file in Fig. 8(c), shows a number of occurrences of an individual note. On the x-axis, the names of the notes are specified and, on the y-axis, a number of times the note has occurred. All the results are discussed and analyzed with the classically trained Music teacher. As can be seen from Fig. 9 that the notes have reduced from 2682 to 134.

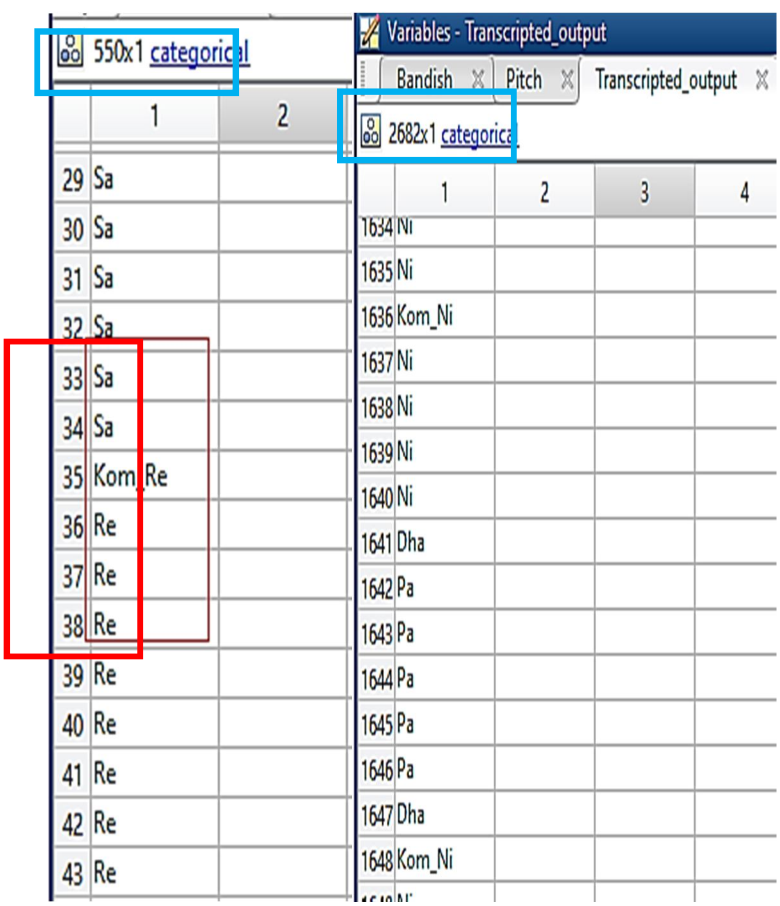

Fig. 7 The transcripted output of Aakar1.wav to the left and Bandish.wav file to the right. Fig. 9 The transcripted output of Bandish.wav after modification of the system

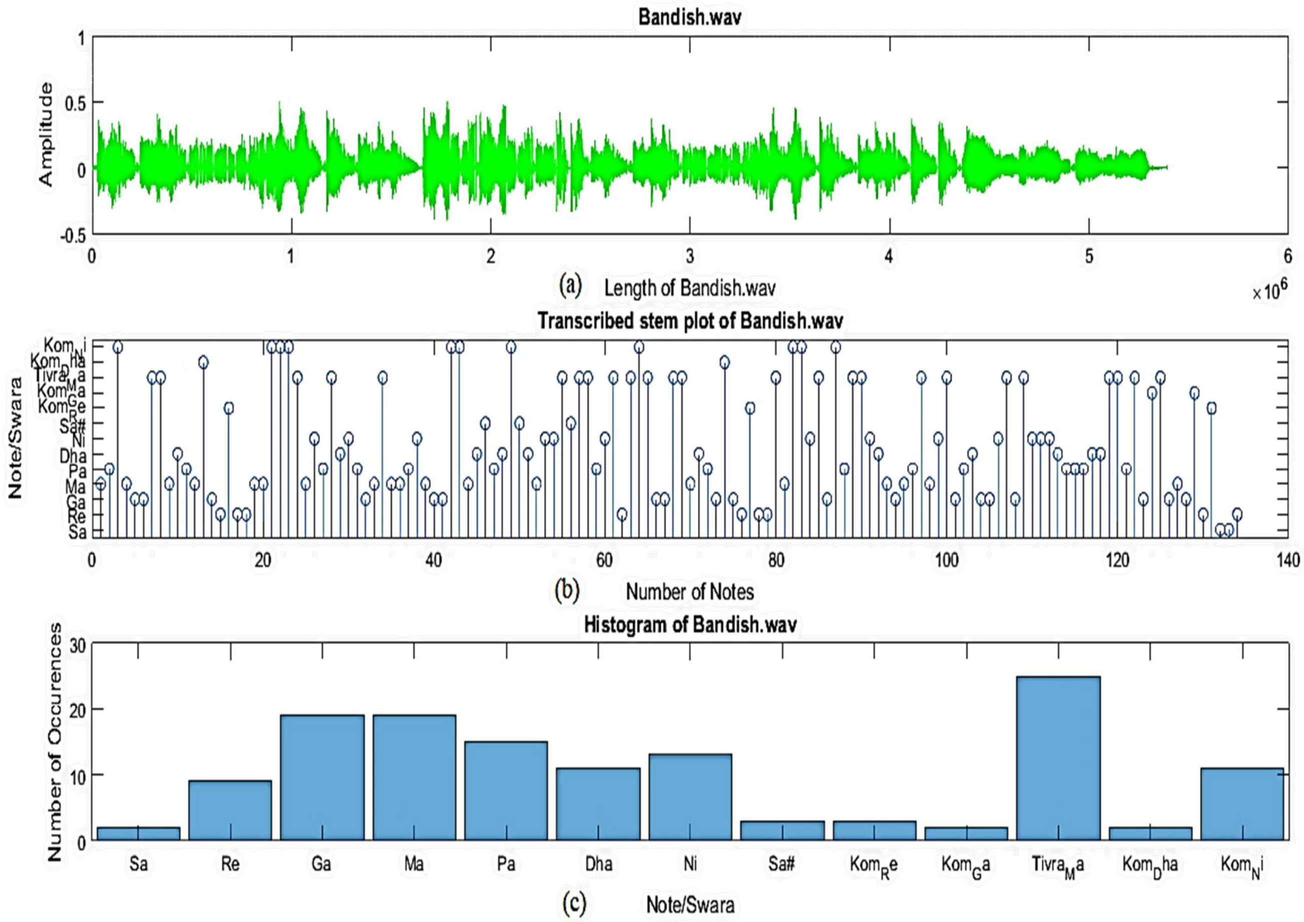

Fig. 8 (a) The plot of Bandish.wav audio file, (b)Transcribed stem plot, and (c) Histogram of Bandish.wav file.

V. CONCLUSIONS

In this paper, we have presented Autocorrelation function for pitch detection and SVM Classifier for classifying notes/swara from the monophonic vocal signal for their transcription. Database for both training and testing are made depending on our assumptions. A general comparison of the Autocorrelation function and YIN algorithm is also covered. SVM with 93.6% accuracy has been built and implemented on the test dataset. The flat version of 'Ga' and 'Dha' notes i.e. 'Komal Ga' and 'Komal Dha' respectively have slightly overlapping pitch ranges due to which accuracy of the model decreases. The optimized output can be found out by grouping extracted pitch samples and finding out their mean value. The output without optimization can be used for research as it gives detailed notes/swara transcription. Optimized output can be further processed and can be provided to the end user.

The accuracy of the system can be still increased by incorporating multiple trained Indian classical singers in the training stage. The same algorithm can be carried out using vocal and drone as a source of base music (drone music helps trained singer to synchronize pitch), then examining the accuracy of the system. Short duration notes and long duration notes can be distinguished on tracking the repetition rate of extracted pitch values. This system can also be modified to an Indian classical music learner application and for detecting the ragas present in the Hindustani music.

VI. ACKNOWLEDGMENT

The Authors would like to thank Dr. H.G.Virani, Head of Electronics and Telecommunication Department, for the motivation. We are also grateful to Dr. Samarth Borkar, Assistant Professor in the Electronics and Telecommunication Engineering Department for his help and guidance. We would also like to thank Mr. Gaurish Tadvalkar & Mr. Sagar Borkar for their support and guidance in understanding the basics of Indian classical music. Our special thanks to Sindhuraj Music Recording Studio.

REFERENCES

[1] P. Dhara, P. Rengaswamy and K. S. Rao, "Designing automatic Note transcription system for Hindustani Classical Music," in International Conference on Advances in Computing, Communications, and Informatics (ICACCI), Jaipur, 2016.

[2] R. B. Kodag, M. D. Patil and C. J. Gaikwad, "Harmonic product spectrum-based approach for Tonic extraction in Indian music," in International Conference on Computing, Analytics and Security Trends (CAST), Pune, 2016.

[3] S. B. Puri and S. P. Mahajan, "Review on automatic music transcription system," in Third International Conference on Computing, Communication, Control and Automation (ICCUBEA), 2017.

[4] Z. C. Fan, J. S. R. Jang, and C. L. Lu, "Singing voice separation and Pitch extraction from Monaural Polyphonic audio music via DNN and Adaptive Pitch tracking," in IEEE Second International Conference on Multimedia Big Data, Taiwan, 2016.

[5] R. Zhao, S. W. Lee, D. Y. Huang and M. Dong, "Soft Constrained leading voice separation with music score guidance," in 9th InternationalSymposium on Chinese Spoken LanguageProcessing (ISCSLP), Singapore, 2014.

[6] F. Pishdadian and J. K. Nelson, "On the transcription of Monophonic melodies in an Instance-based Pitch classification scenario," in IEEE, 2013.

[7] P. Rao, J. C. Ross, K. K. Ganguli, V. Pandit, V. Ishwar, A. Bellur and H. Murth, "Classification of Melodic Motifs in Raga music with Time-series matching," New Music Research, vol. 43, no. 1, 2014.

[8] A. d. Cheveigne and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," Acoustical Society of America, vol. 111, no. 4, pp. 1917-1930, 2002.

[9] S. Pandiaraj, L. Gloria, N. R. Keziah, and K. R. S. Kumar, "A proficient vocal training system with Pitch detection using SHR," in IEEE, Coimbatore, 2011.

[10] S. Gao and Y. Zhu, "Extracting vocal Melody from Karaoke music audio," in IEEE, 2005.

[11] V. Bharathi, A. A. A and R. Ramya, "Vocal Pitch detection for musical transcription," in International Conference on Signal Processing, Communication, Computing, and Networking Technologies, 2011.

[12] J. Salamon and E. Gómez, "Melody extraction from Polyphonic music signals using Pitch Contour characteristics," IEEE Transaction on Audio, Speech, and Language Processing, vol. 20, no. 6, pp. 1759-1770, 2012.

[13] G. E. Poliner, D. P. W. Ellis, A. F. Ehmann, E. Gómez, S. Streich, and B. Ong, "Melody transcription from music audio: approaches and evaluation," IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 4, pp. 1247-1256, 2007.

[14] S. Ghisingh, S. Sharma and V. K. Mittal, "Study of Indian Classical music by singing voice analysis and music source separation," in 2nd International Conference on Telecommunication and Networks, 2017.

[15] S. Balke, C. Dittmar, J. Abeßer and M. M¨uller, "Data-driven solo voice enhancement for Jazz music retrieval," in ICASSP, Ilmenau, 2017.

[16] N. Kroher and E. Gómez, "Automatic transcription of Flamenco singing from polyphonic music recording," IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, vol. 24, no. 5, pp. 901-913, 2016.

[17] K. Akant and S. Limaye, "Pitch Contour extraction of singing voice in Polyphonic recording of Indian Classical music," in International Conference on Electronic Systems, Signal Processing, and Computing Technologies, Nagpur, 2014.

[18] K. K. Ganguli, A. Lele, S. Pinjani, P. Rao, A. Srinivasamurthy and S. Gulati, "Melodic shape stylization for robust and efficient Motif detection in Hindustani vocal music," in Twenty-third National Conference on Communications (NCC), 2017.

[19] Y. S. Murthy and S. G. Koolagudi, "Classification of vocal and non-vocal regions from audio songs using spectral features and Pitch variations," in IEEE 28th Canadian Conference on Electrical and Computer Engineering, Halifax, 2015.

[20] V. Rao and P. Rao, "Vocal Melody extraction in the presence of Pitched accompaniment in Polyphonic music," IEEE Transaction on Audio, Speech, and Language Processing, vol. 18, no. 8, pp. 2145-2154, 2010.

[21] G. B. S and N. U. Bhajantri, "An exhaustive review of automatic music transcription techniques survey of music transcription techniques," in International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), 2016.