Implement Real-Time Polyphonic Pitch Detection and Feedback System for the Melodic Instrument Player

Geon-min Kim[1], Chang-hyun Kim[2,*], and Soo-young Lee[1]

[1] Department of Electrical Engineering,
CNSL, Korea Advanced Institute of Science and Technology, Daejeon, Korea
[2] Department of Bio and Brain Engineering, CNSL,
Korea Advanced Institute of Science and Technology , Daejeon, Korea
{gmkim90,sy-lee}@kaist.ac.kr, flipflop98@gmail.com

Abstract. This research proposes an automatic transcription-feedback system of music which help people to learn musical instruments by themselves. The focus of this research is piano. We develop real-time polyphonic pitch detection-feedback system. For 'polyphonic pitch detection', we use inner product based similarity measure with discriminant note detection threshold and top down attention. Also, we develop two parallel processes on simulink and matlab separately for real-time system. On simulink workspace, real-time recording and signal flow management is implemented. This system takes 2mins. 12secs. for analyzing 1min. piece and have accuracy of pitch detection as 79.33% for test case (Chopin Nocturne Op.9 N.2).

Keywords: Real-time Polyphonic Pitch Detection, Feedback System, Note-scale filterbank, Multi-threshold, Top-down attention.

1 Introduction

More and more people want to learn new musical instruments, but there are not many possible ways for someone to study musical instruments by themselves. It is not easy for beginners to get self-feedback from playing the instrument alone. Thus, this research proposes an automatic transcription-feedback system which will help people to learn musical instruments by themselves.

Fundamental algorithms of pitch detection in time-frequency domain have been researched so far [1], [6]. Also Autotune[2] and Melodyne[7] are well known commercialized programs for monophony and polyphony pitch detection. However, those programs do not transcribe well on commercial CDs and real-time performances.

Therefore through this research, a system will be constructed which will provide real-time pitch detection for polyphonic music, express the music as sheet music, and give feedback to instrument player by comparing with the correct reference of the music. Also we want to compare the polyphonic pitch detection performance with

[*] Corresponding author.

T. Huang et al. (Eds.): ICONIP 2012, Part IV, LNCS 7666, pp. 140–147, 2012.

competitive algorithms [6], [8]. Emmanouil B. et al. [6] announced that multi pitch analyzer[8] algorithms have the best polyphonic pitch detection rate as 70.9% on Chopin Nocturne Op.9 N.2.

The focus of this research is the piano music. First, piano is a polyphonic musical instrument unlike other monophony instruments such as woodwinds or brasses. Also, piano is composed of 88 different sounds from A0 to C8, so it contains a long range of pitches which includes all the pitch ranges for many different musical instruments.

1.1 Note Scale Filterbank Output

In this system, we utilize note-scale filter bank output as feature of the music signal. It has 103 coefficients, which extract information from spectrogram of music. 103 coefficients are calculated by filtering spectrogram with the 103 filters. Each filters are a form of triangle, where their center frequencies are located at $27.5 \times 2^{[0:102]/12}$. Note that first 88 coefficients are located at fundamental frequencies of 88 notes of piano according to the previous research.[3] Remaining 15 coefficients are for extracting higher harmonics of note frequencies. By using note-scale filter bank spectrogram, we can selectively emphasize fundamental frequency information from the spectrogram, which makes pitch detection more easy task.

2 System

2.1 Real-Time System

Figure 1 shows the system of real-time polyphonic pitch detection & feedback system that we developed in this research. There are 3 main parts.

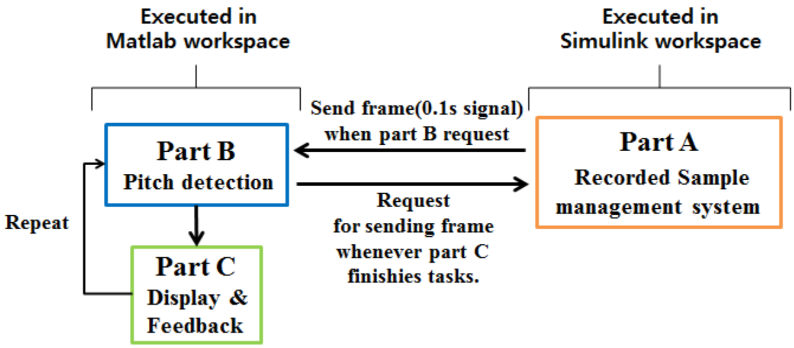

Fig. 1. Simplified real-time system diagram

- In part A, music signal is recorded from microphone and stored in queue with 0.1sec per frame length. Then it is sent to each frame one by one repeatedly to part B whenever part B requests for sending frame.

- In part B, the real-time pitch detection system gets a frame from queue in part A as first-in-first-out (FIFO) sense. Algorithms about poly-phonic pitch detection is shown in section 2.2
- In part C, we display the pitch detection result on time vs. pitch number axis, giving the player the feedback note correction information with ground truth music score.

The complete system diagram is shown in Figure 2. And the detail function of each subdiagrams are explained in section 3.2

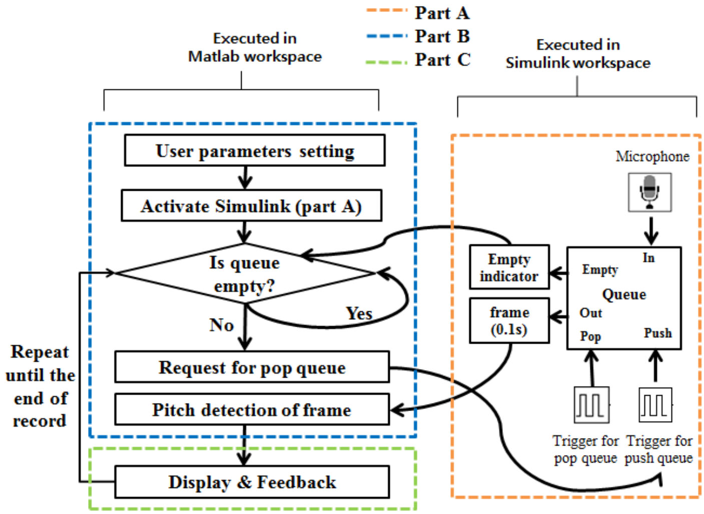

Fig. 2. Complete real-time system diagram

In part C, the program display a feedback, which consist of correct, incorrect, and missing note numbers. Figure 3 shows how feedback information is shown in display for 9s music.

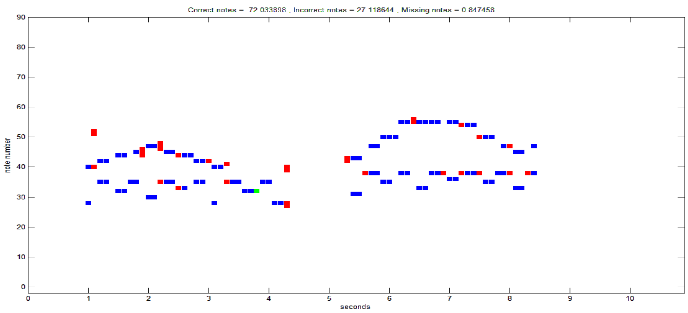

Fig. 3. Example of the feedback display (until 9s)

For given frame(0.1s), we filled each detected note with blue, red and green color on correct, incorrect and missing notes separately. Player can know whether they play correct pitch with correct beat or not. Program also shows the real-time accuracy of the performer's music by comparing with the score, which has the note sample index versus note pitch number axis.

2.2 Pitch Detection Algorithm

Figure 4 shows the algorithm of pitch detection. Firstly, standardized (Mean zero, Variance one) music signal is converted into note-scale filterbank output (=feature). Then this feature is normalized on every time frame. Next, inner product with the references(pitch templates), which are the average values of feature of 88 individual notes. Note that we can measure similarity of two different normalized vectors by inner product. The references are made by following sequences; 1) recording each individual notes of piano, 2) getting feature, 3) time-averaging and 4) normalize them. Since there are many overlapping harmonics between 88 notes, 88 features are not orthonormal with each other, which we can see the simulation result in section 2.3. We use two methods for supporting deficient parts of inner product as pattern recognition; 1) Different detection thresholds for each notes. 2) Top down attention. Details of these methods are explained in section 2.3 and 2.4 respectively.

After inner product process, if the value exceed detection threshold for each note, that note is regarded as played note candidates, otherwise regarded as silence. By using top down attention for these candidates, algorithm gives final detected notes.

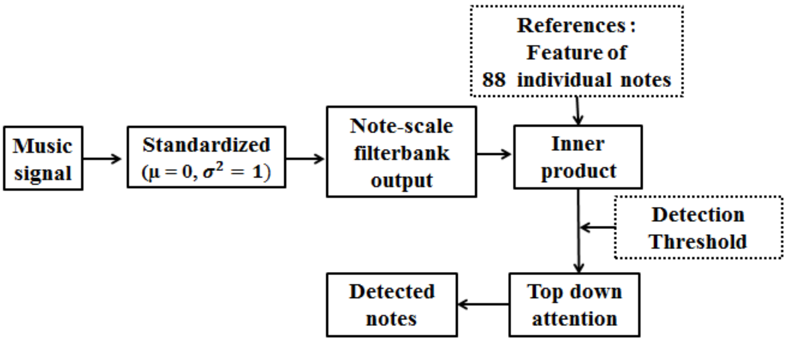

Fig. 4. System diagram for pitch detection algorithm

2.3 Method to Set Different Detection Thresholds for Each Notes

Before setting detection thresholds, we made an experiment by measuring similarity between 88 references. Fig 5 shows the inner product between 88 notes features. Red,yellow,and green parts except main diagonal shows references are not orthonormal with each other. Especially the notes lower than number 25(A2) and notes higher than number 80(E7) have high value of similarity with other notes pattern. Based on this observation, for each note, we set high detection threshold when the average value of similarity is high, and set low detection threshold when the average value of similarity is low.

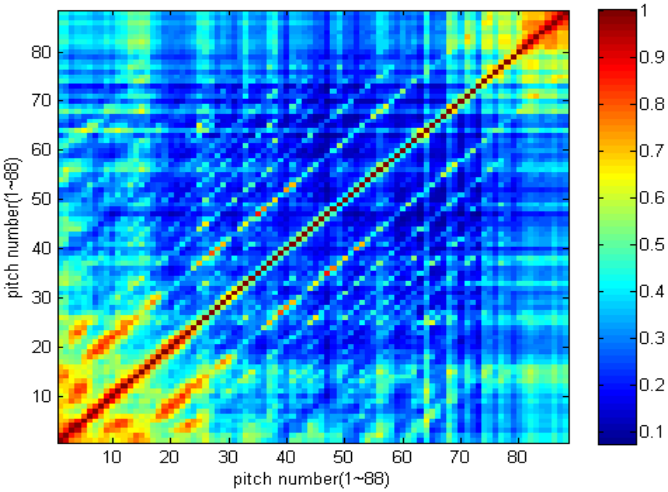

Fig. 5. Similarity between feature of 88 piano notes

2.4 Top Down Attention

Before top down attention step (see Fig 4), system already obtain the note candidates; some of them are 'really played notes', some of them are 'not played notes'. Top down attention is the method that can figure out whether individual pattern exists inside mixed pattern or not. For our system, it finds 'really exist' notes among candidate notes.

Followings are description of the algorithm of top down attention in our system;

1) Load feature of candidate notes .Let the number of candidate notes = N

2) If N=1, The algorithm ends

3) If N$\geq$2, pick two notes from candidates (let feature of two notes as $\vec{x},\vec{y}$). The number of method to pick different sets of two features is equal to $\frac{N(N-1)}{2}$.

4) Let feature of test music at given time as $\vec{M}$.

5) Find a,b which minimizes $\text{error } e = \|\vec{M} - (a\vec{x} + b\vec{y})\|$ by finding pseudo inverse of the linear system $(\vec{x} \quad \vec{y})\binom{a}{b} = \vec{M}$.

6) If one of $\binom{a}{b}$ makes 'e' smaller than given threshold, accept corresponds two notes.

7) Repeat from 3) until iterations run $\frac{N(N-1)}{2}$ times.

On each figure 6 and 7 are two test results with and without top-down attention.

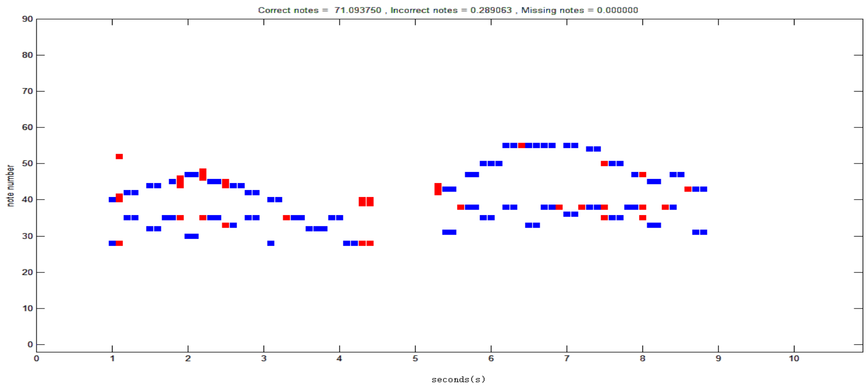

Fig. 6. Results without top-down attention (Accuracy : 71.09%)

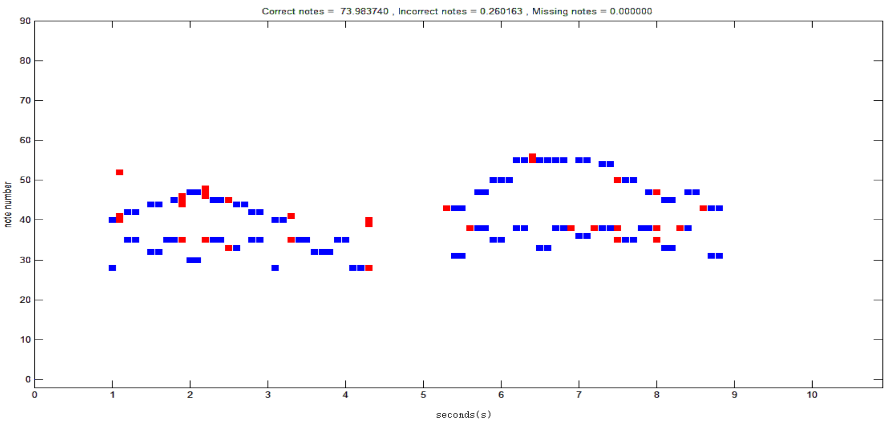

Fig. 7. Results without top-down attention (Accuracy : 73.98%)

We can see that top-down attention can reduce detecting false-positive note, which is note that is not played but detected as note alive by system.

3 Test and Performance

3.1 Performance Evaluation Criteria

We develop ‘real-time pitch detection system’ for this project. Thus performance can be evaluated by two criteria : Accuracy &Speed.

For accuracy,we usefollowing evaluation metrics : $\text{Accuracy} = \frac{Tp}{Tp+Fp+Fn} \times 100\ [\%]$, which is the simplest metric for evaluation of accuracy. Some of researcher [4], [5] uses ‘Precision’, ’Recall’, and ‘F-measure’ as their evaluation metrics. (Where, Tp = “true positive”: number of correct notes among played notes; Fp=“false positive”:

number of incorrect notes among played notes; Fn:"false negative": number of not played notes among reference notes).

For the speed of the algorithm, we measure the average computation times for analyzing whole music.

3.2 Test Data and Condition

For testing our system, in terms of accuracy and computation time, we use the piece ' Nocturne Op.9 N.2 'of Frederic Chopin. To measure accuracy of real-time pitch detection itself, we use MIDI reference, which contains correct answer of the piece, MIDI was created by the Prokeys 88(MIDI controller device) and Cubase 6 (MIDI sequencing program).

We are doing test with normal room (i.e. no silent condition), normal speaker and normal microphone, which can represents the normal user's recording environment.

3.3 Performance

Table 1. Evaluation measure and speed of our system(frame-based)

Measure	Speed
Accuracy = 79.33% (Tp = 3843 , Fp = 716 , Fn = 285) Precision = 0.843 , Recall 0.931, F-measure : 0.885)	532.4s process/4min 2s music ≅ 132s/1min music

Fig 8 shows the feedback display for test of our system.

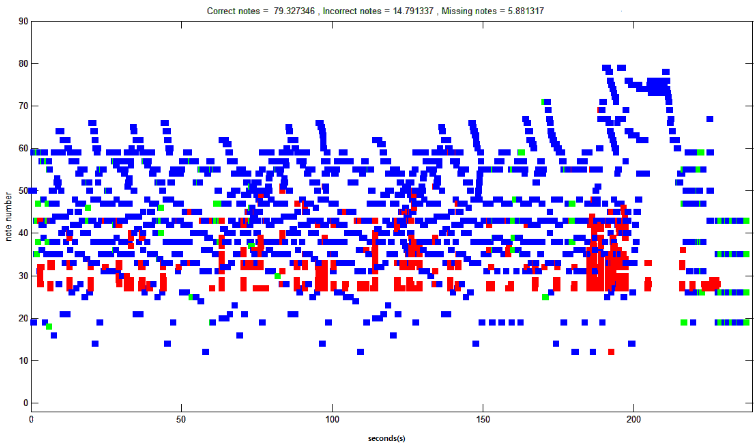

Fig. 8. Feedback display of the test

4 Conclusion and Future Work

In summary, we develop real-time polyphonic pitch detection-feedback system. For 'polyphonic pitch detection', we use inner product based similarity measure with discriminant detection threshold and top down attention (See Section 2.4). And for real-time system, we develop two simultaneously running process system in simulink and matlab. One is for real-time recording and signal flow management, and the other is for real-time pitch detection and displaying feedback to users.

As a final result, accuracy of pitch detection of our system is 79.33% for 4min music and takes 132s to analyze 1min piece, which is over 8% improvement to the state of art system[8].

For the future work, we can add the system for reduce the effects of room acoustics for considering different user's environment. And the accuracy should be improved by considering musiccal knowledge such as key, beat, harmonic science etc.

Acknowledgement. This research was supported by Qualcomm incorporated. Chang-hyun and Geon-min are co-first authors, who equally contributed to this paper. We thank professor Sang-Hoon Oh and Dr. Wonil Chang for their helpful advices.

References

1. David, G.: Pitch Extraction and Fundamental Frequency: History and Current Techniques. Technical report. University of Regina, pp. 2–12 (2003)
2. Worldwide Standard in Vocal Processing Tools: Auto-Tune, http://www.antarestech.com/products/auto-tune-evo.shtml
3. Edward, M.B.: Intervals, Scales, and Tuning. In: Deutsch, D. (ed.) The Psychology of Music. Academic Press, San Diego (1999)
4. Mert, B., Andreas, F.E., Stephen, D.: Evaluation of Multiple-F0 Estimation and Tracking Systems. International Society for Music Information Retrieval 2009 (2009)
5. Graham, E.P., Daniel, P.W.E.: A Discriminative Model for Polyphonic Piano Transcription. EURASIP J. Appl. Sig. P. 8, 1–9 (2007)
6. Emmanouil, B., Simon, D.: Joint Multi-Pitch Detection Using Harmonic Enve-lope Estimation for Polyphonic Music Transcription. IEEE J-STSP 5, 1111–1123 (2011)
7. Melodyne, http://www.celemony.com/cms/
8. Kameoka, H., Nishimoto, T., Sagayama, S.: A Multipitch Analyzer Based on Harmonic Temporal Structured Clustering. IEEE Trans. Audio, Speech, Lang. Process. 15, 982–994 (2007)