Automatic Piano Music Transcription

Jianyu Fan — Jianyu.Fan.GR@dartmouth.edu

Qiuhan Wang — Qiuhan.Wang.GR@dartmouth.edu

Xin Li — Xin.Li.GR@dartmouth.edu

Recap

Our aim is to create automatic transcription, converting piano music recording to MIDI [1] files. The wav music recording files contain all the information about audio signals that can be used to extract audio features directly. We transcript the wave files into MIDI files which have simple representation of music information such as pitch, duration, rhythm, and dynamics.

Process

- Feature Extraction

In this part, we have two sub-questions to solve: the onset detection and audio frequency feature extraction. We have used the self-labeled data gathered before.

1. Onset detection

We have run the onset detection algorithm [2][3] to get the starting point of each note. The algorithm uses the edge detecting filter, doing fast convolution and finding where exactly the peak occurs. Here is the result:

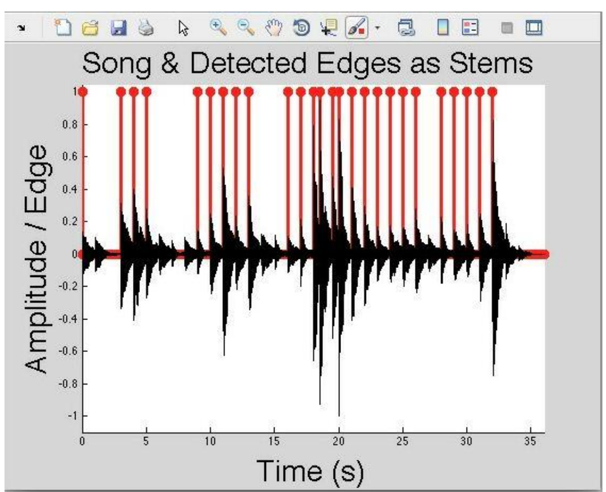


Figure.1 Onset Detection

2. Audio Frequency Feature Extraction

We have separated each wave file into frames. The starting position of each frame is computed by onset detection. We have used total energy and harmonic feature in each frame as features [4][5].

We have computed feature one that measures the total energy in the signal. The energy is expected to be high in most cases. However, it might be lower at the beginning of the note, at the end of it, or between notes. In each frame, we have computed each point through frequency axis and got the result of the position of harmonics. The feature vector will then be $(y^1 \dots, y^K)$. We assumed that the components of the vector are conditionally independent. y^1 is the total energy in the signal, and $y^2 \dots y^k$ are harmonics features.

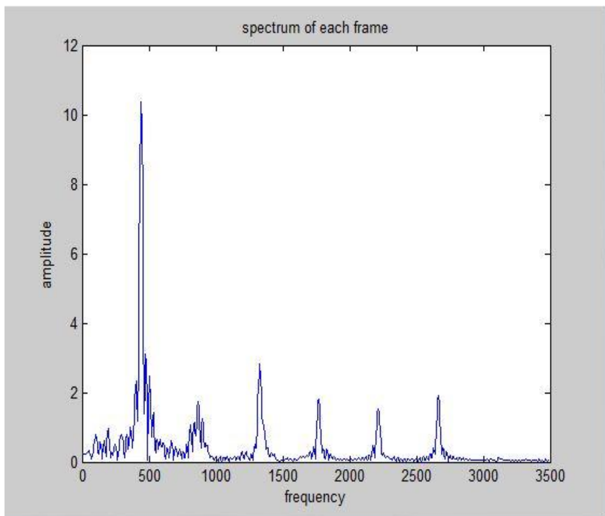


Figure.2 Spectrum of One Frame

In Figure2, the frequency that reaches the highest amplitude is the fundamental frequency, and while the other frequencies that have lower amplitudes are the harmonics (they also might be fundamental frequencies for other different notes).

Figure 3 and Figure 4 are the results of Feature Extration.

feature <6872x89 double>

	1	2	3	4	5	6	7
939	0	0	0	0	0	0	0
940	0	0	0	0	0	0	0
941	0	0	0	0	0	0	0
942	0	0	0	0	0	0	0
943	0	0	0	0	0	0	0
944	0	0	0	0	0	0	0
945	1	0	0	1	0	0	0
946	1	1	0	0	1	0	0
947	1	1	0	0	1	0	0
948	1	1	0	1	0	0	0
949	1	1	0	0	1	0	0
950	1	0	0	0	1	0	0
951	1	0	0	0	1	0	0
952	1	1	0	0	1	0	0

Figure.3 Feature Data

state <6872x1 double>

	1	2	3	4	5	6	7
939	0						
940	0						
941	0						
942	0						
943	0						
944	0						
945	42						
946	42						
947	42						
948	42						
949	42						
950	42						
951	42						
952	42						

Figure.4 Labeled Data

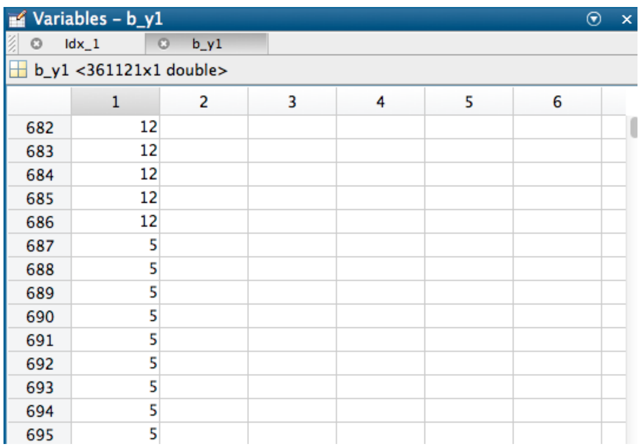
When y^1 equals 0, it means the absence of any notes, then all the other features will be 0. In the other case, the following columns will contain the harmonic feature from $y^2 \ldots y^k$. If it is 1, it means it is a harmonic point.

Figure.4 represents the labeled data. 0 here means rest and the number refers to the position of note on a piano key board.

- Test Data using K-means

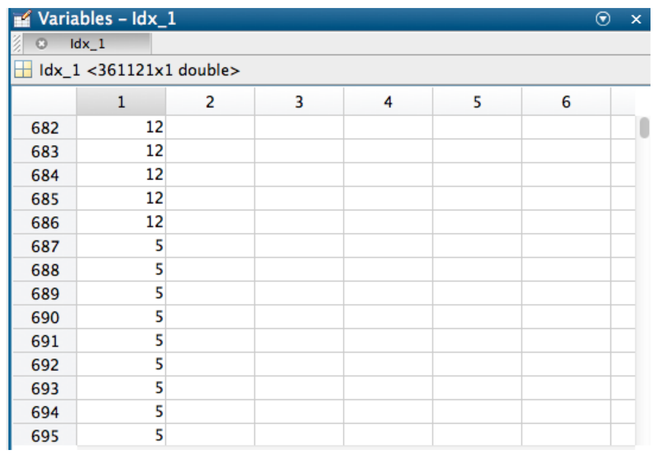
We moded all note pitch into one octave which contains 12 pitch classes. We used K-means to test whether the different note could be well clustered. Then we compared the labeled data with data cluster by K-means and got the error rate. We set the "rest" from 0 to 12 to different with note pitch and have used the most voted class method for the label. The test set size is 361121 frames. Here are the results:

The error rate is: 0.2177


Variables – b_y1
Idx_1 b_y1
b_y1 <361121x1 double>

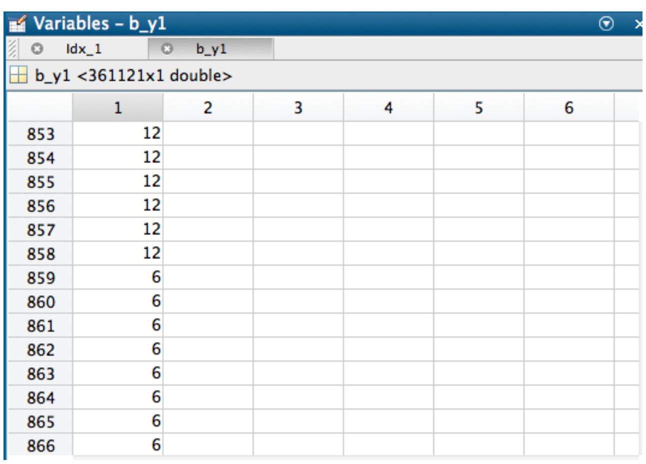
	1	2	3	4	5	6
682	12					
683	12					
684	12					
685	12					
686	12					
687	5					
688	5					
689	5					
690	5					
691	5					
692	5					
693	5					
694	5					
695	5					

Figure.5 Labeled Data


Variables – Idx_1
Idx_1
Idx_1 <361121x1 double>

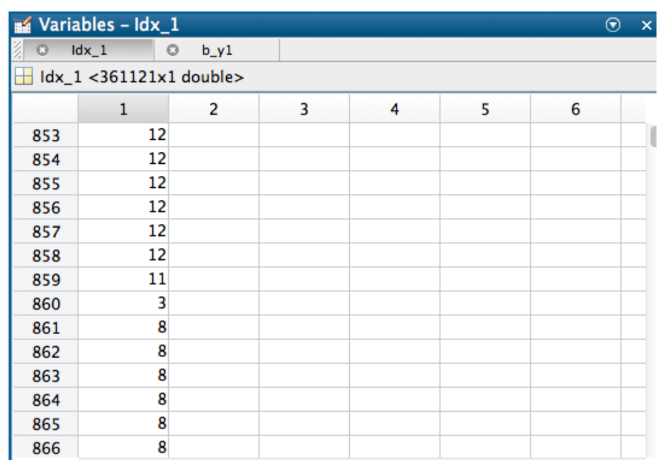
	1	2	3	4	5	6
682	12					
683	12					
684	12					
685	12					
686	12					
687	5					
688	5					
689	5					
690	5					
691	5					
692	5					
693	5					
694	5					
695	5					

Figure.6 Good Results of K –means


Variables – b_y1
Idx_1 b_y1
b_y1 <361121x1 double>

	1	2	3	4	5	6
853	12					
854	12					
855	12					
856	12					
857	12					
858	12					
859	6					
860	6					
861	6					
862	6					
863	6					
864	6					
865	6					
866	6					

Figure.7 Labeled Data


Variables – Idx_1
Idx_1 b_y1
Idx_1 <361121x1 double>

	1	2	3	4	5	6
853	12					
854	12					
855	12					
856	12					
857	12					
858	12					
859	11					
860	3					
861	8					
862	8					
863	8					
864	8					
865	8					
866	8					

Figure.8 Bad Results of K –means

From the figures above, we could see that K-means may somehow recognize the note with relatively high error rate, and there will be some notes that may be incorrectly labeled by this method. The error is partly because the error we get during the process of feature extraction and the partly because of the K-means itself.

- HMM

While applying the HMM Model [6], we encountered two problems. First, the frame was too small to be a representative state. Second, the small window size produced 89 features in each frame. When we tried to label each frame, we treated each feature's information as a binary number and then converted all the frames into a decimal number. Since we got 89 features for each frame, the result could reach 2^{89}. The Matlab cannot present that large data in positive number form, which will act as index role in matrix.

Future Work

- Modify the frame size and window size to re-extract training data
- Continue working on HMM model prediction
- Evaluate the system using testing data

References

[1]. MIDI. In Wikipedia. Retrived. Jan 24,2013, from http://en.wikipedia.org/wiki/MIDI

[2], Bello, J.P.; Daudet, L.; Abdallah, S.; Duxbury, C.; Davies, M.; Sandler, M.B.; , "A Tutorial on Onset Detection in Music Signals," Speech and Audio Processing, IEEE Transactions on , vol.13, no.5, pp. 1035- 1047, Sept. 2005`

[3], Piano note onset detection, from http://cnx.org/content/m14196/latest/

[4]. C. Raphael, "Automatic transcription of piano music," in Proc. Int. Conf. Music Information Retrieval (ISMIR), Paris, France, 2002.

[5]. Automatic transcription of piano music based on HMM Tracking of Jointly-Estimated Pitches. 16th European Signal Processing Conference (EUSIPCO 2008), Lausanne, Switzerland, August 25-29, 2008, copyright by EURASIP

[6]. Kevin Murphy. Hidden Markov Model (HMM) Toolbox for Matlab, 1998, from http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html