INTERPETING A MUSICAL SCORE SHEET

Objectives:

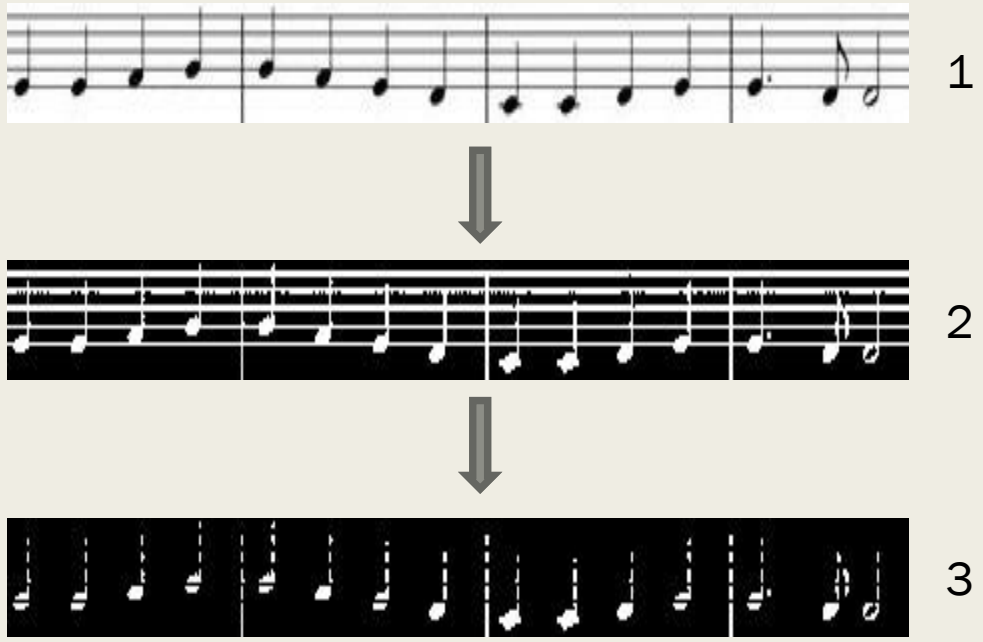
- Identify the difference between notes and the rest.
- Identify the pitch of the note using pixel locations.
- ~~Differentiate different kinds of notes.~~
- Output a sound clip of a musical score sheet.

~~Abc~~: not completed

Identifying where the staff lines are.

- The musical sheet is transformed into grayscale and were inverted with a certain threshold (1 to 2). The matrix now have values either 1 or 0.

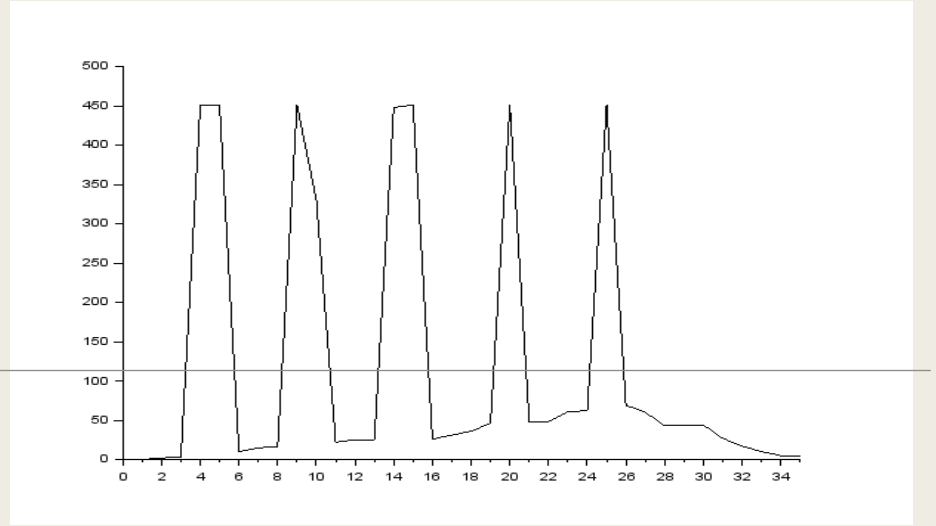
- A histogram (4) was used to check the sum between rows. Rows whose sums are over a certain threshold are regarded part of the staff lines and are removed. (2 to 3)



Identifying where the staff lines are.

- The staff locations are identified manually as one staff are 3-4 rows wide.

threshold

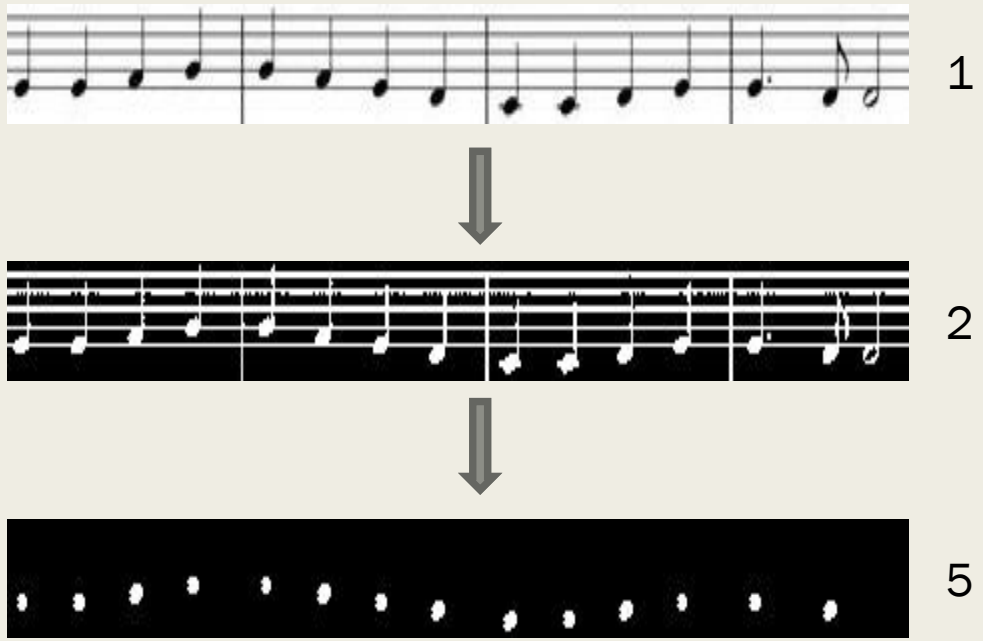


4

Histogram of the sum per row

Identifying the notes.

- The musical sheet is transformed into grayscale and were inverted with a certain threshold (1 to 2)
- Notes are generally circular. The image is first eroded with a 5x5 circular structuring element and then dilated back with the same structuring element. (2 to 5)



1

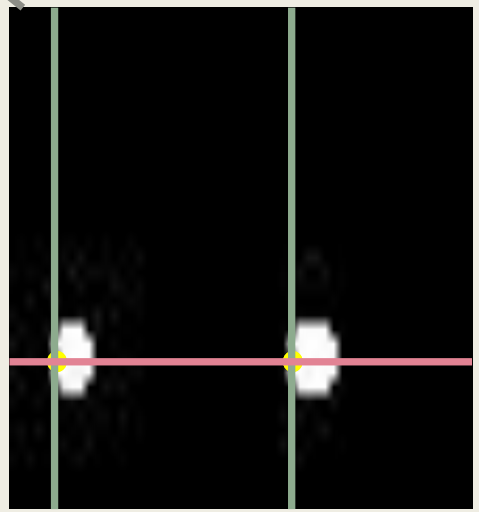
2

5

Identify the pitch of the note using pixel locations.

- To locate the notes, a histogram (8) was made to get the sum per column. The first column with a non-zero before having consecutive non-zero sums are retained while (lines in green) while the rest becomes zero.
- The same procedure is used in row. (red line). Because of this only pixel points (in yellow) are left.
- Using the data in identifying staff lines the pitch frequencies are determined by region. Each pixel point belongs to a region of a certain pitch.

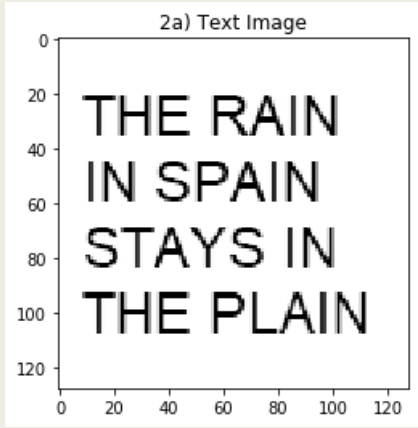
6



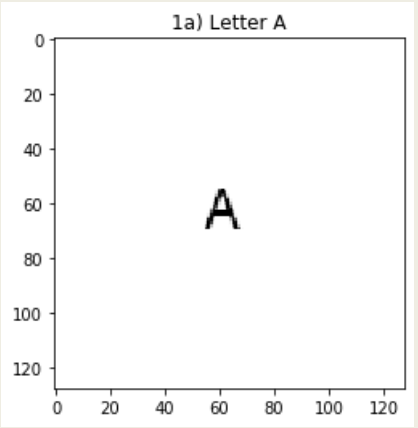
7

Differentiate different kinds (lengths) of notes.

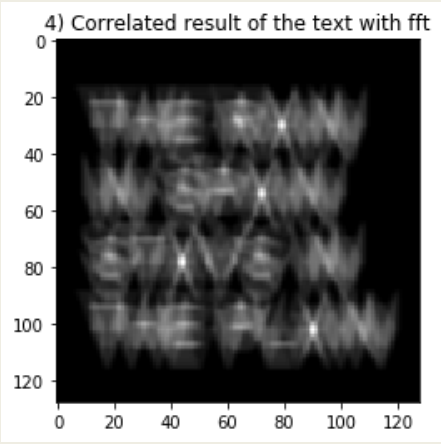
- I was stuck think how to diferentiate notes with filling vs w/o filling (whole vs quarter) and notes with stem/flag vs w/o stem/flag (quarter vs eight).
- I heard many people and Mam Jing recommended a way called template matching. Peaks are obtained on regions that looks the same.
- A threshold can be used to only detect the brightest spots.



Base Image



Pattern to Match



Result

Output a sound clip of a musical score sheet.

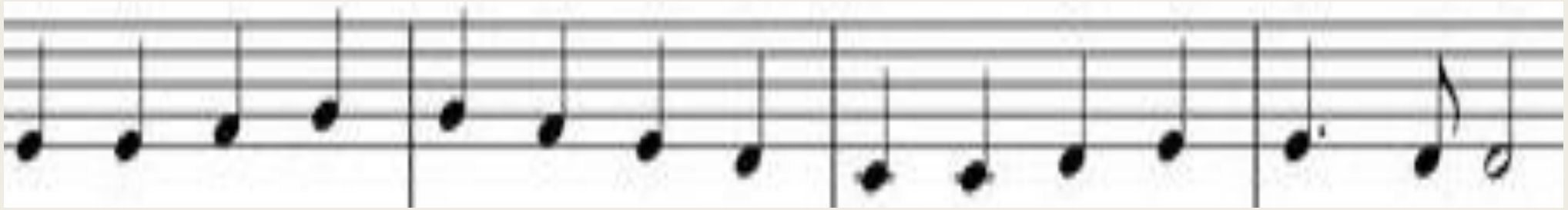
- Mam Jing gave us a function on how a note sounds. “f” is the pitch in frequency (C,D,E,F…) while “t” is the length (whole, half quarter…)
- The soundsec() command is used to defined t. Since t is a matrix, the loop part of the code is used to append a ‘note vector’ with each other.
- The sound clip is saved via savewrite()

```
93
1   function n = note(f, t)
2       n = sin (2*%pi*f*t);
3   endfunction;
97
98
99
100 for i = 1:size(Notes,1);
101     if i == 1 then
102         SoundClip = note(Notes(i), t_14);
103     else
104         count2 = size(t_14,2);
105         SoundClip($+1:$+count2) = note(Notes(i), t_14),
106     end,
107 end;
```

Code for sound

How’s the sound

- The music sheet used is the first four beats of Ode to Joy.
- All of the shaded notes (quarter, eighth) got sounded as quarter notes. (dots and flags were not distinguished)
- Unshaded notes (half note) was not sounded.



Future Works and Special Thanks

- Learn how watershed function works (from main reference) or use template matching.
- Identify stems, flags and dots.

- Main Reference: https://pdfs.semanticscholar.org/145e/d587d85877befde88502ab880590ee5f455b.pdf
- Pitch to Frequency converter: https://pages.mtu.edu/~suits/notefreqs.html