MIR Final Presentation

王士亞 | 洪若淮 | 蔡絜

Content

01 Chord Recognition

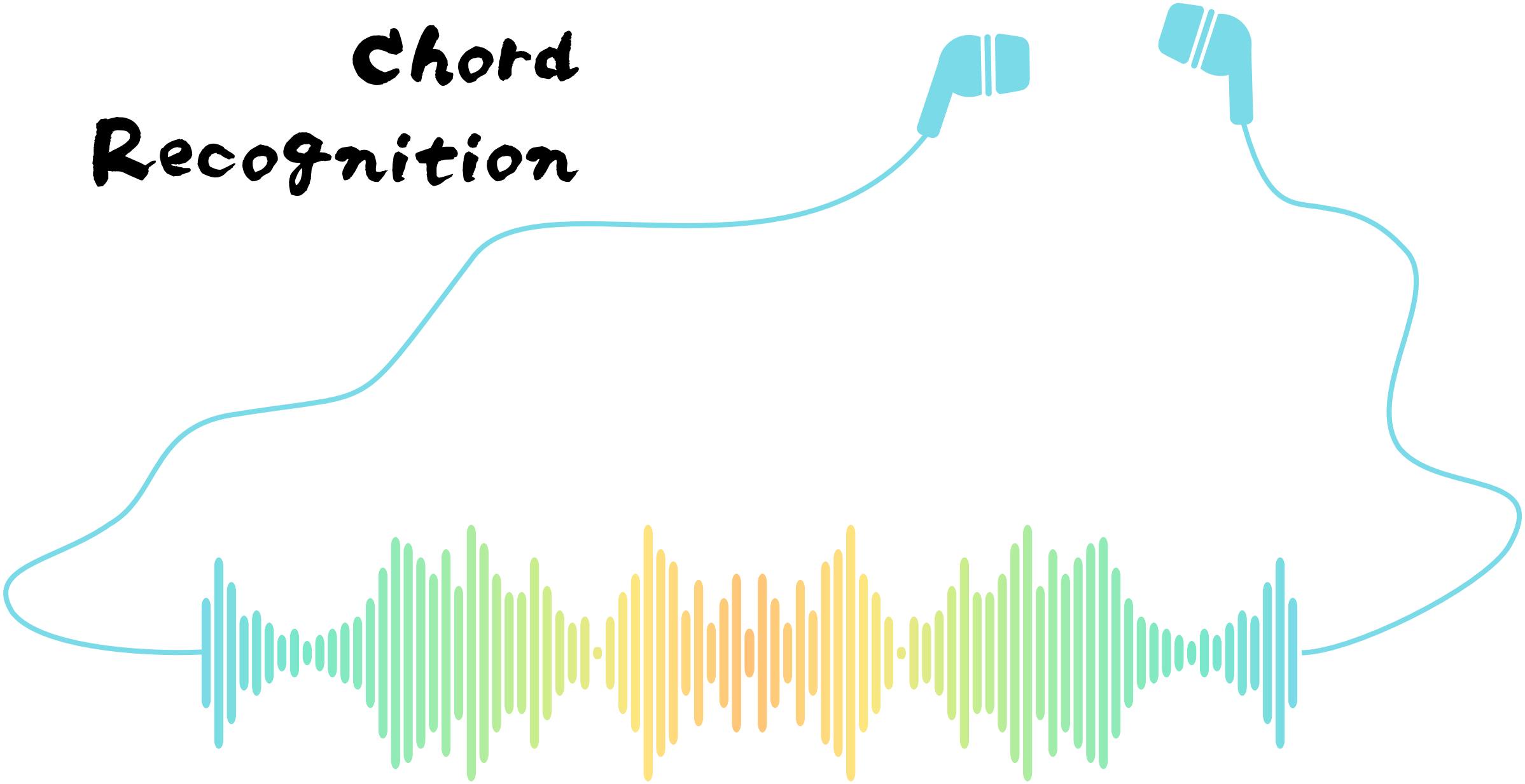
02 Motivation

03 Methodology

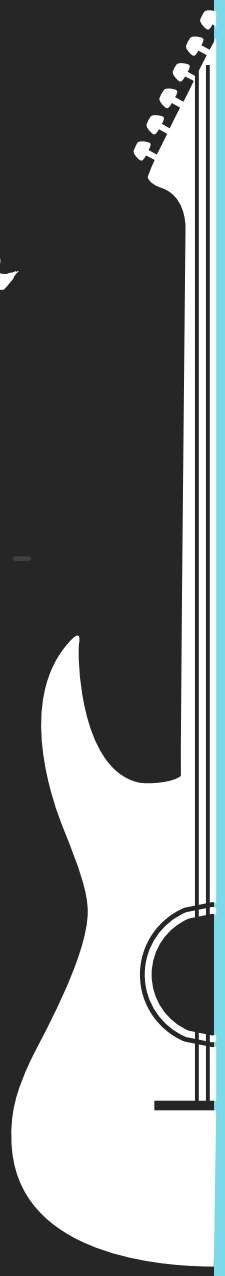
04 Demo

05 Discussion & Future Perspective

Chord Recognition



Motivation



When we listen to a song, we often want to know what the chords are in order to play the guitar or other instruments. But sometimes it's hard to find the music sheets for some songs.

So we came up with an idea of chord recognition to find the chords of an input waveform and use python to implement it.

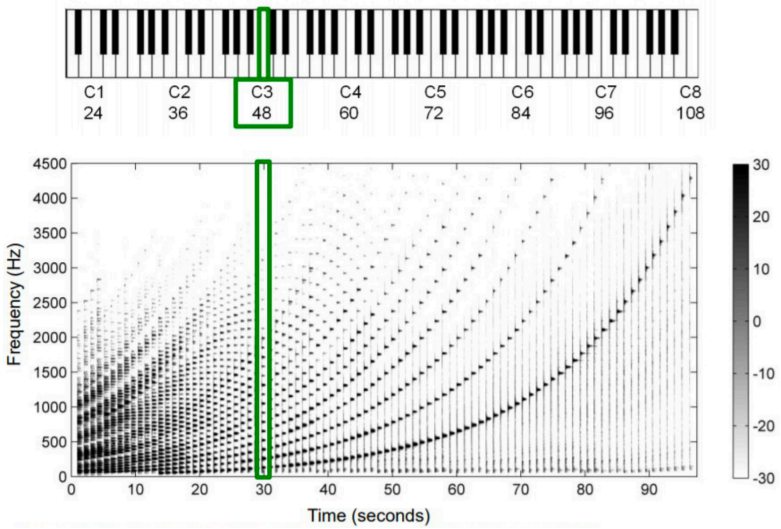
Methodology

Methodology

1

Spectrogram

matplotlib.pyplot.specgram

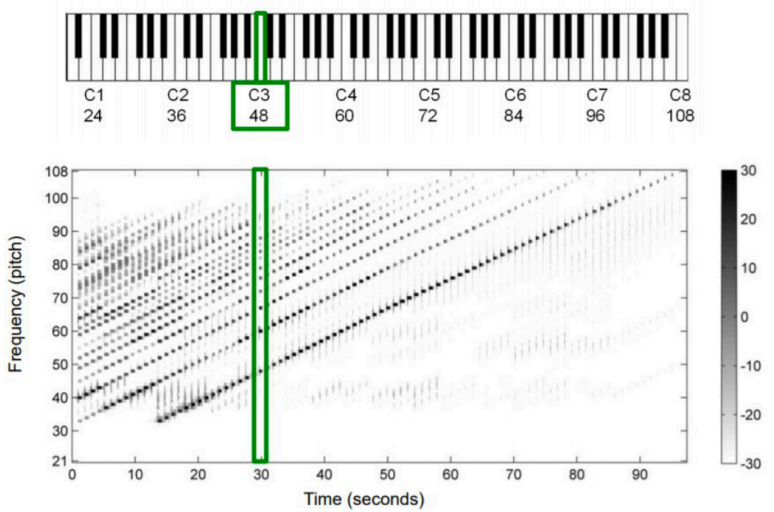


Dot product

2

Filterbank

log-frequency filterbank
(a.k.a CQ filterbank)



3

CQ Spectrogram

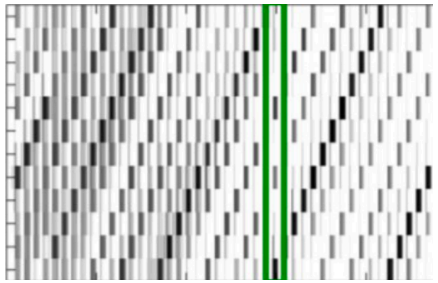
Spectrogram · CQ filterbank

Methodology

4

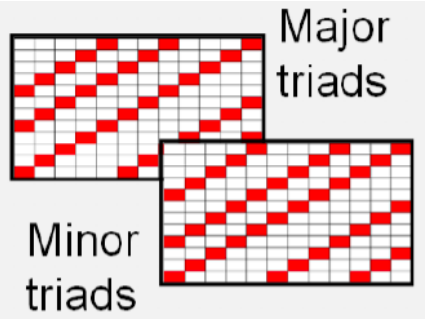
Chromagram

Merge all the bins with the same note name of all octaves together



5

Template

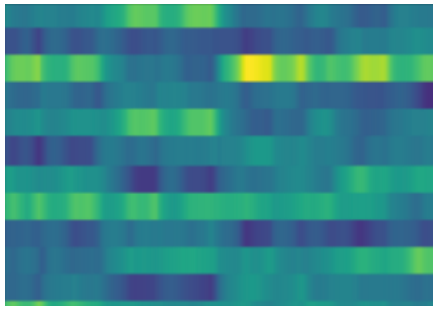


Dot product

6

Template Matching

Chromagram · Template



Methodology

7

Postfiltering

Smoothing for enhancement:
take average on each of the component of the sequence (mean filter)

8

Chord Detection

Max function

Methodology

Filterbank

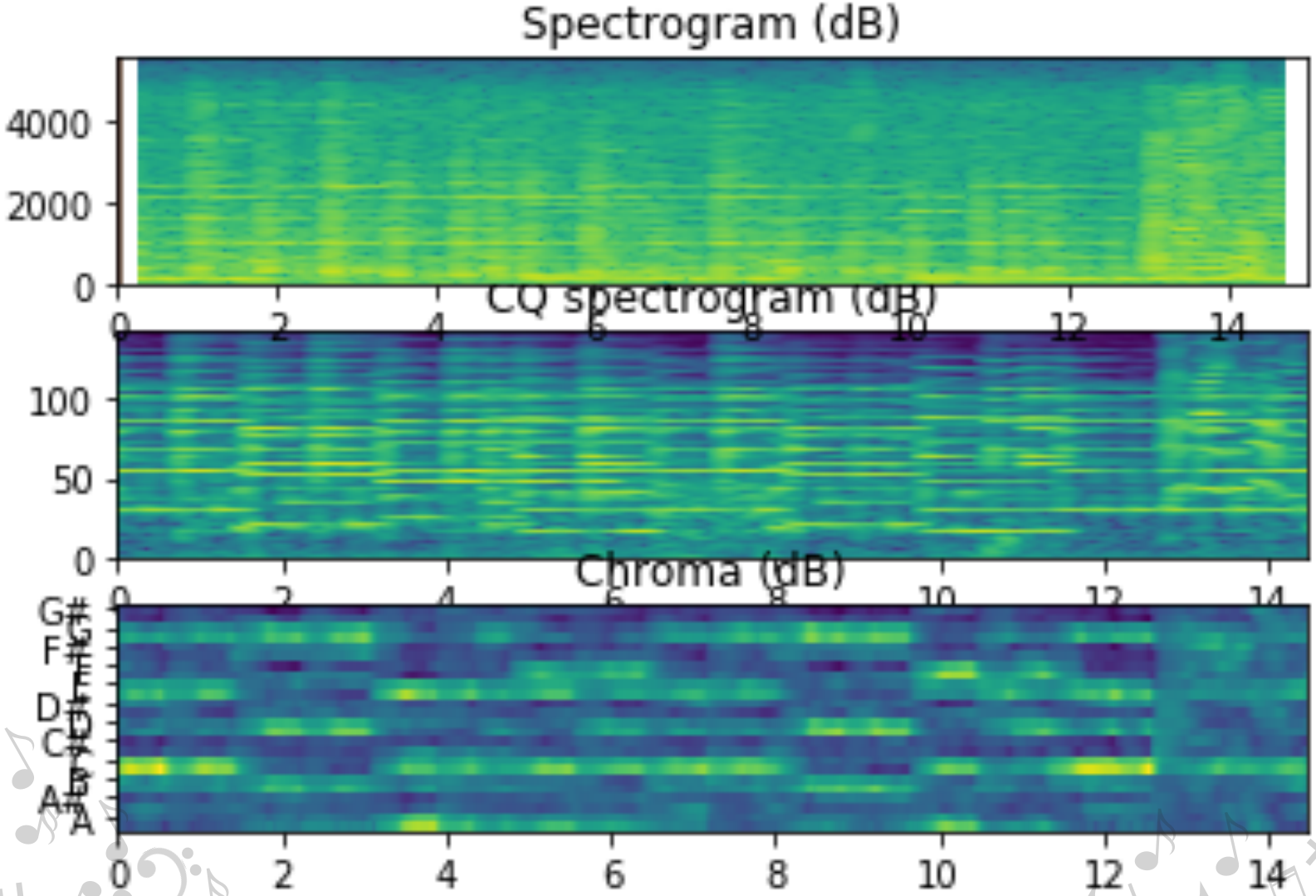
```
def generate_filterbank(NFFT,fs,b,z):
    #b is bins per octave
    #z is number of octaves
    #b = 24
    #z = 6
    #fs(downsampled) = 44100/4 = 11025

    octmax = b*z

    octpts = np.arange(-1,octmax+1)
    #print 'octpts',octpts
    #print len(octpts)

    ctrfrq = oct_to_hz(octpts)
    print ("ctrfrq",ctrfrq)
    #print len(ctrfrq)

    ctrrep = np.floor((NFFT+2)*ctrfrq/((fs/2)))
    print ("ctrrep",ctrrep)
    print (len(ctrrep))

    bank = np.zeros([len(octpts)-2,int(NFFT/2)+1],dtype=float)

    for j in range(0,len(octpts)-2):
        y = np.hamming(ctrrep[j+2]-ctrrep[j])
        area = trapz(y, dx=5)
        if area==0:
            area=1
        y2 = (y/area)
        bank[j,int(ctrrep[j]):int(ctrrep[j+2])] = y2
        #plot(bank[j,:])

    #show()
    return bank
```

Methodology

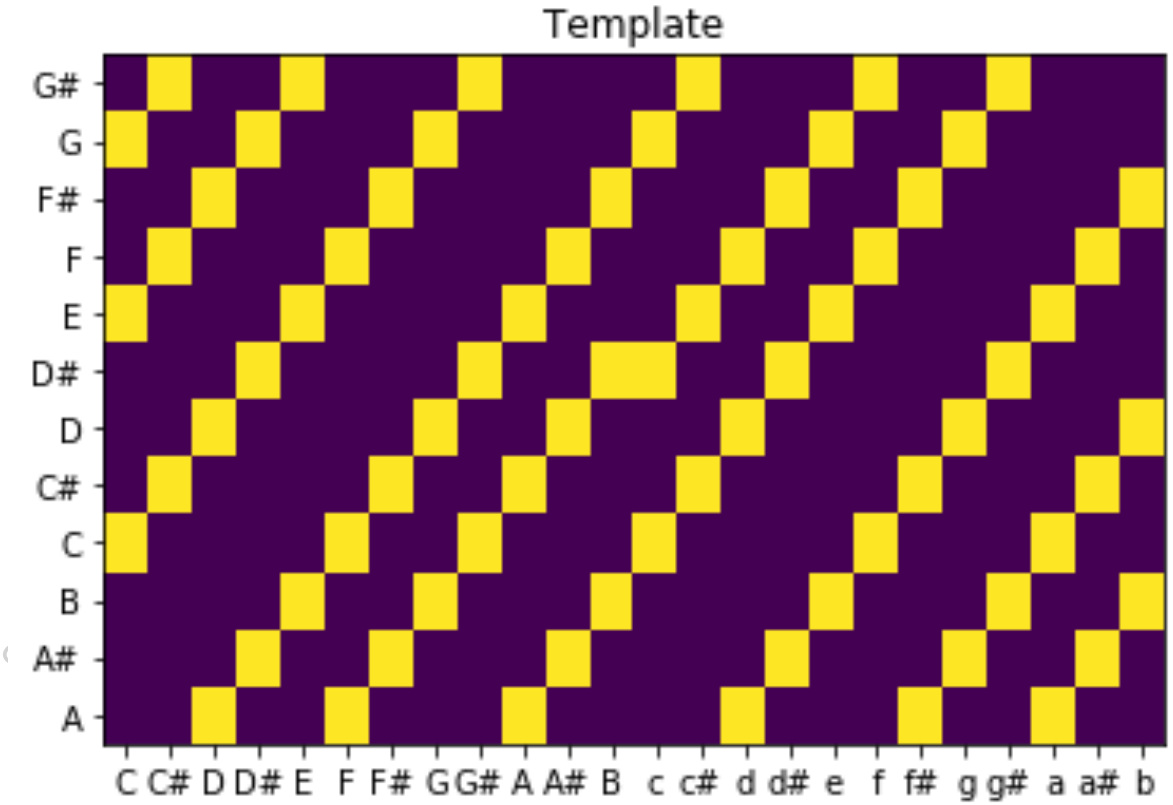
Chromagram

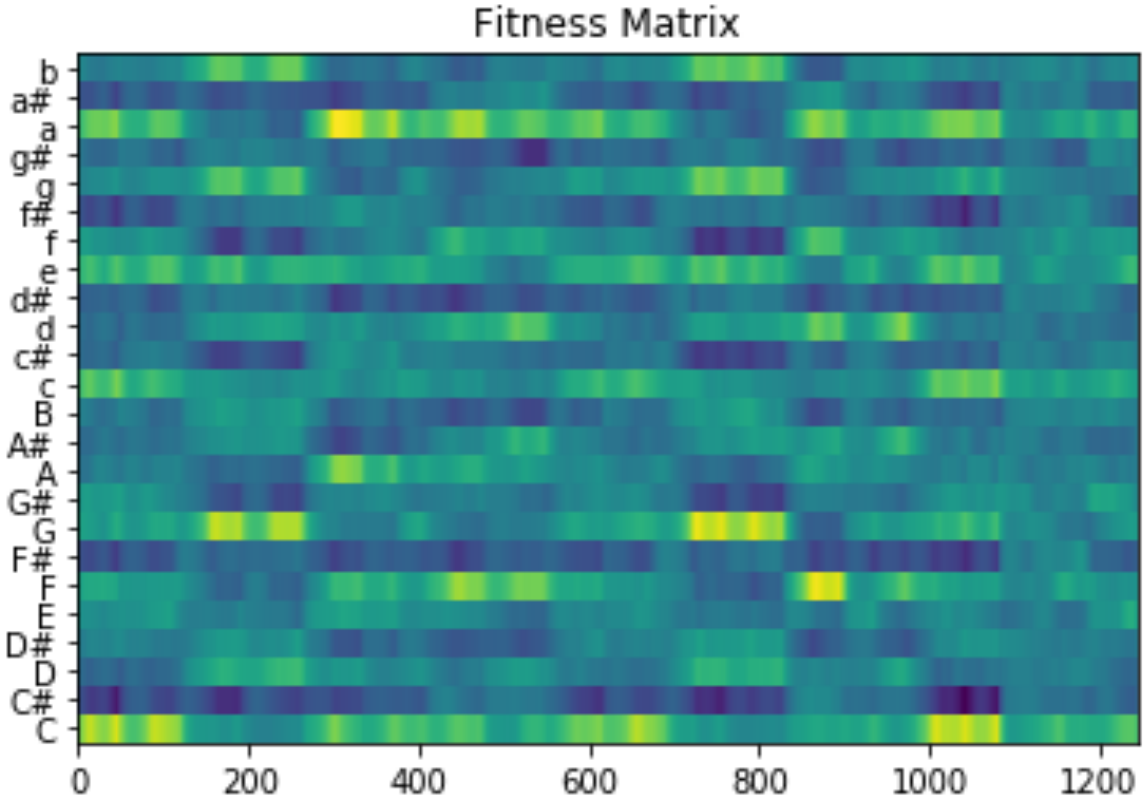
```
"""
Generate Chroma from CQ spectrogram
"""
row,col = shape(salm)
#print row
#print col
b = 24

chrm = np.zeros(b*col).reshape(b,col)

for i in range(col):
    c = salm[:,i]
    for j in range(b):
        chrm[j,i] = sum(c[j:row:b])

#print chrm
chrm_new = np.zeros(b*col).reshape(b,col)

chrm_new = chrm

chrm_new_norm = np.zeros(b*col).reshape(b,col)
#normalise
for i in range(col):
    chrm_new_norm[:,i] = chrm_new[:,i]/sum(chrm_new[:,i])
    #chrm_new_norm[i,j] = np.median(chrm_new[i,j-10:j+10])
#print chrm_new_norm
```

Methodology



Methodology





Methodology

Postfiltering

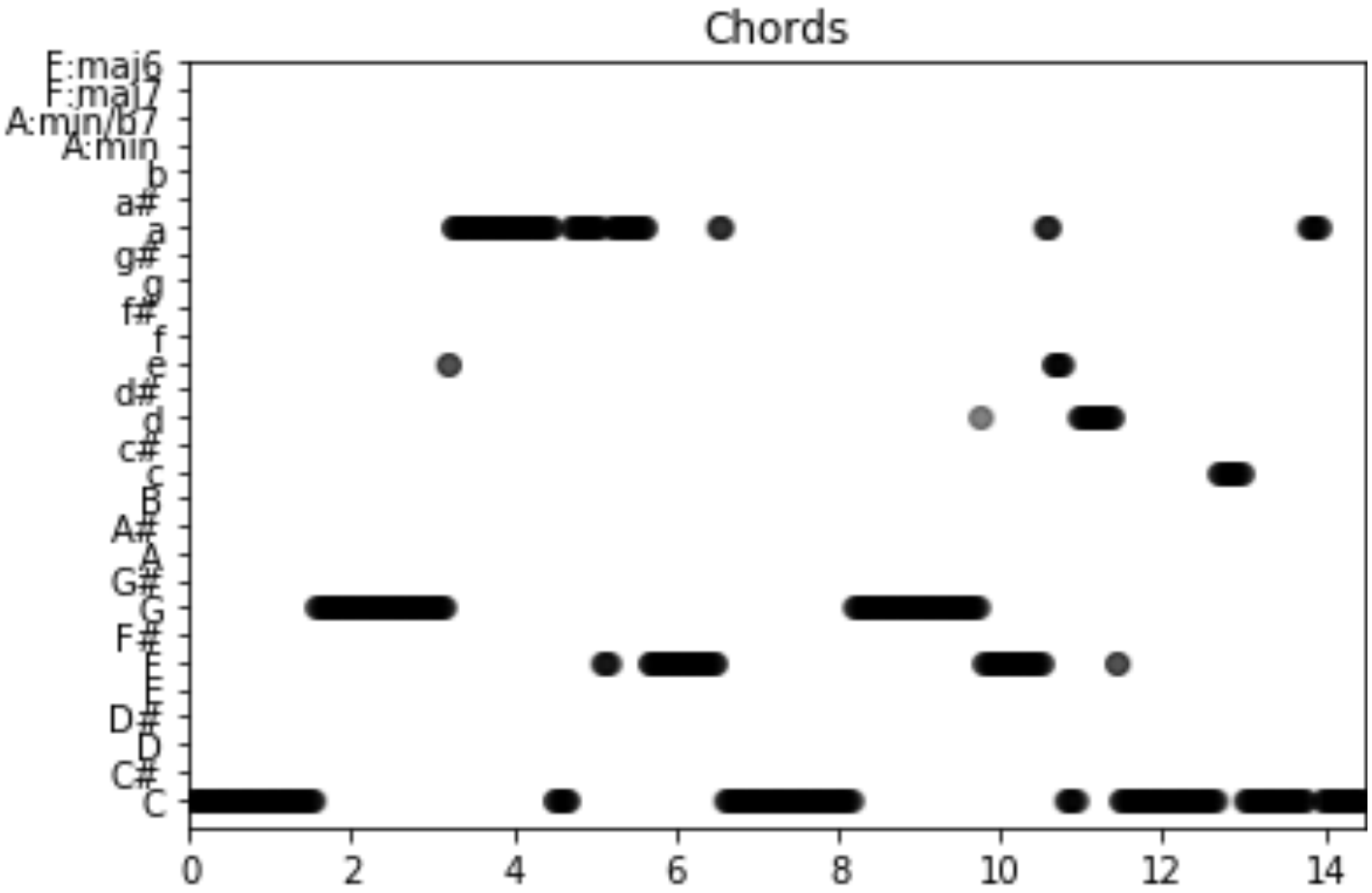
```
"""
Read the fitness matrix file and plot it
"""
chrm = np.load('tempogram1.npy')
print (shape(chrm))
b,col = shape(chrm)

whereAreNaNs = np.isnan(chrm);
chrm[whereAreNaNs] = 0;

chrm_new = np.zeros(shape(chrm))

for i in range(b):
    for j in range(col):
        chrm_new[i,j] = np.mean(chrm[i,max(0,j-10):min(col,j+10)])  #mean filtering

chrm_new = chrm
```

Chord Detection

```
"""
Chord detection using max function
"""
chrm_max = np.zeros(shape(chrm_new))
chrm_array = np.zeros(col)

for i in range(col):
    maxl = chrm_new.argmax(axis=0)
    chrm_max[maxl[i],i] = 1

chrm_array = chrm_new.argmax(axis=0)
'''
```

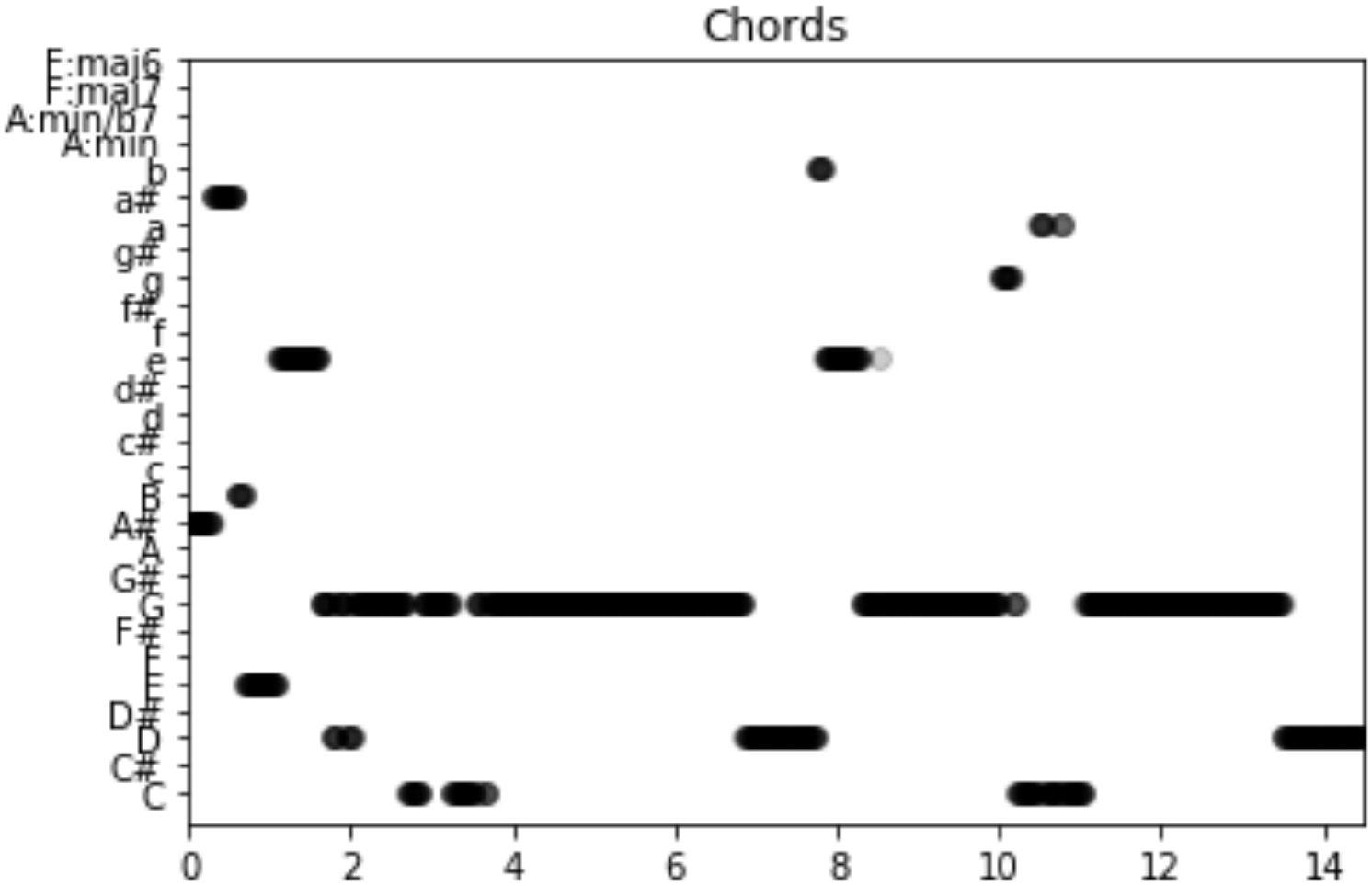
Methodology

Let It Be



Methodology

晴天



Demo

Discussion & Future Perspective

Discussion & Future Perspective

Expand the template

Add more chords to the template. E.g. A:min, Fmaj7

Enhance accuracy

Supervised or exemplar-based learning

Develop into website

Reference sample: Chord Tell

Thank You