

2020 Intel Cup Undergraduate Electronic Design Contest

- Embedded System Design Invitational Contest

# Final Report



Intel Cup Embedded System Design Contest

## Guitar tab transcription system based on Computer Vision

Students:

Wyoming-Hsu, Yue Ming-Han, Pu Shen-Zuo

Tutor:

Ming-Ling

School:

Southeast University

# **GUITAR TABS TRANSCRIPTION SYSTEM BASED ON COMPUTER VISION**

## **ABSTRACT**

The project designs a guitar tabs transcription system based on Computer Vision, no matter the real time performance or the video, the music will be changed into guitar tabs. The project introduces a novel way in guitar music transcription domain. In this project, firstly, an instance segmentation model trained by Mask-Rcnn is used to extract guitar fingerboards from videos in real time. With a series of image processing operations, we obtain the boundaries' information of frets. Then an api is called to get fingertips' coordinates from the video. We match fingertips' information with previous boundaries' information to build arrays which have three elements (string number, fret number and time information). Finally, a program changes these arrays to readable guitar tabs or midi files.

## **Introduction**

Music score is the base of music, recording humans' music on the paper from ancient times till now.

We are in the information era, we can learn an instrument from videos on the net easily. However, for an instrument lover, especially for those starters, it might be a difficulty to obtain the music score they are interested in.

At present, there are two major methods to transcribe music to music scores. The first is to transcribe by ears. The second is to apply Fourier audio analysis. Both of them have flaws, the problems of efficiency and accuracy.

We now try to design a system which is based on computer vision to transcribe guitar music.

## **Methodology**

### **1. Mask-Rcnn**

As the guitar video was input, the video firstly will be divided to frames. The instance segmentation model will be applied to every frame. The model will target the guitar fingerboard and draw a mask on the fingerboard frame by frame.



**Fig.1 Mask-rcnn's application**

About the model, we use the code from [github/matterport/Mask-rcnn](https://github.com/matterport/Mask-rcnn). A dataset was built which has about 400 different guitar pictures and 200 pics for a data augmentation, labeled one by one. We also applied a pre-trained model COCO used for transfer learning to make up for the lack of samples. After 957 iterations, the model train loss and validation loss both decrease to 0.06.

## 2. Open-Cv

After the first step, we have the mask. Bitwise function will extract the picture under the mask from original image. By extracting fingerboard, it greatly helps us to perform following operations.



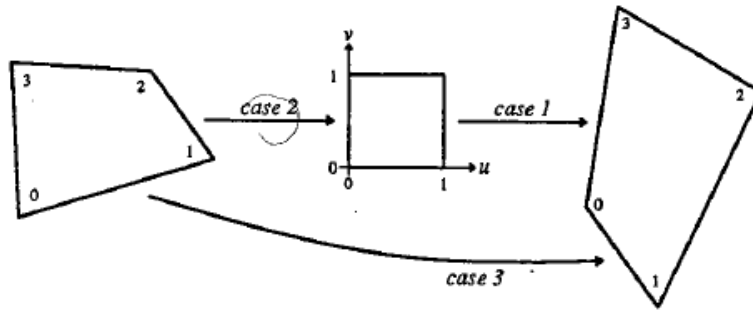
**Fig.2 Bitwise function's application**

FindContours function is applied to fingerboard. It aims to return the coordinates of the fingerboard in the input image.



**Fig.3 findContours function's application**

Also, the perspective transform is used to normalize the fingerboard to ease the frets and strings detection.



**Fig.4 perspective transform**

After operations above, we are going to detect fret boundaries and strings. First, we enhance the picture to make frets' lines or strings more identifiable by combined morphological image processing and histogram parameter calculation. Then we detect frets and strings from separate enhanced picture. Two functions are used.



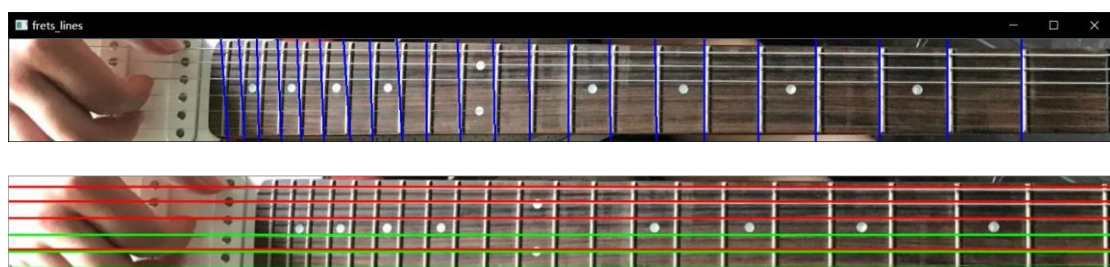
**Fig.5 Enhanced frets**

#### 1.frets\_detect function

It aims to detect all of frets coordinates, which will return in variate "line\_sort", in the input image. Hough line transform and bubble sort are applied in the function. Besides, it includes solving first order linear recursive equation to speculate the frets coordinates which aren't detected. With the result of the function, it is easy for the computer to detect which frets finger touched.

#### 2.string\_detect function

It aims to detect all of strings coordinates, which will return in variate "line\_string", in the input image. Hough line transform and bubble sort are applied in the function. Besides, linear interpolation enables the function to speculate the undetected strings. With the result of the function, it is easy for the computer to detect which strings finger touched.



**Fig.6&7 Detection of Frets and Strings**

### 3. Api

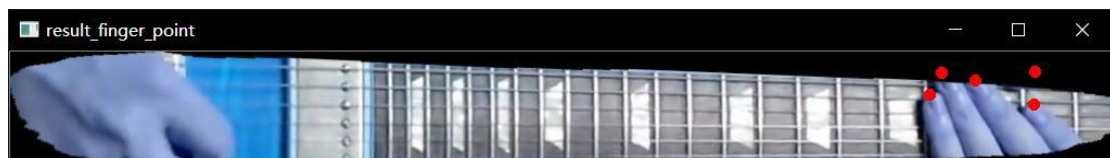
Firstly, we have two plans for the fingertips detection. The first one is to use google's mediapipe engine, but we don't find a pre-trained model. The second one is to call the api. Api is easy to use. We choose the hand key points detection api from baidu which can send back the

coordinate of key points on hands. The accuracy is acceptable.



**Fig.8 Api's application**

Then the coordinate will be combined with previous frets' and strings' information.



## 4. Tabs Transcription program

With previous operation, we have got enough information. A list of arrays will store it as below.

`[5, 3, 1], [], [5, 3, 2], [], [3, 0, 3]...`

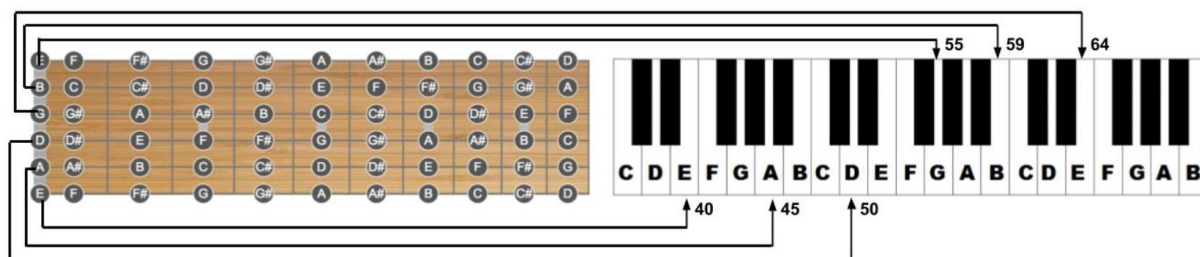
The program is compiled to change the arrays we have into readable guitar tabs.

Six lines consist of '-' and numbers.

```
e|-----
B|-----
G|----0-0-2-2-0-----
D|-----3-3-2-2-0-0-----
A|-3-3-----3-----
E|-----
```

**Fig.9 the little star score transcribed by program**

The midi generator program is based on an open source lib midiutil. The notes' generations follow picture below.



**Fig.10 guitar notes' pitch**