Generating Audio for Muted Piano-Playing Video Using Deep Learning

Project Milestone

XIA Junzhe
20493411
jxiaaf@connect.ust.hk

YANG Baichen
20493198
byangak@connect.ust.hk

HUANG Zeyu
20493631
zhuangbi@connect.ust.hk

1. Introduction

It is noticed that human playing the piano effectively generates a series of well-patterned actions, i.e. the position of hands and the keys and the depth of keys being pressed down. The notes that the piano generates strongly obey to this visual pattern. Hence, it becomes reasonable to recognize the visual patterns from piano-playing videos and reproduce the instrumental sounds using machine learning.

In this project, we aims to achieve hand gesture detection in piano playing and generate the correponding audio afterwards. It is also expected that we can thereby generate piano sound by our model without a real sound-producing piano, but only with a ”fake” one.

2. Problem Statement

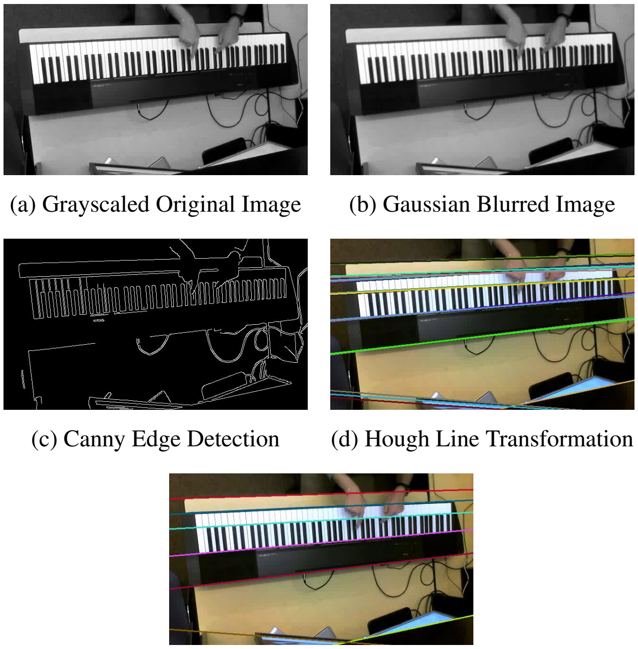
In this section we will mainly discuss our problem formulation, dataset sources, the evaluation pattern and expected results.

2.1. Keyboard Detection Approach

Among all of the previous work that we have studied, no deep learning based method is used to localize the keyboard in video frame. Instead, all of them used traditional CV method such as Hough Line Transformation, brightness comparison to determine the coordinates of four corners of the piano keyboard (Fig.1). We have first partially reproduced this algorithm.

After pruning repeated lines, some hard-coded method will be used to finally determine the positions of four corners. However, after applying this method to multiple images with different light condition and hand positioning, we found following drawbacks:

- To achieve a satisfying performance, several hyperparameters in Hough Line Transformation and Canny Edge Detection needs to be adjusted dramatically in different images.



(a) Grayscaled Original Image

(b) Gaussian Blurred Image

(c) Canny Edge Detection

(d) Hough Line Transformation

(e) Prune Repeated Lines

Figure 1: Pipeline of Finding Boundaries of Keyboard

- To accurately determine the corners' coordinates, it is required there is no hands posing above the keyboard. It is hard to guarantee since most of piano videos do not contain such a ”empty” frame. In some cases, the camera angle will even change slightly, where the traditional method will fail.

2.2. Evaluation and Expected Result

3. Technical Approach

3.1. Improving Keyboard Detection using Deep Learning

Due to the weak robustness of the traditional CV method, we decided to use a convolutional neural network to localize coordinates of four courners of the keyboard. However, we do not have any existing dataset for this task. So we labelled a number of images captured from different piano playing videos, which come from the dataset of a previous work, as well as some videos on the Internet.

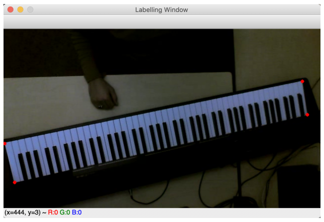


Figure 2: Labelling Tool

To augment this dataset, we will also consider artificially putting keyboard images onto some random backgrounds and perform some basic transformation such as flipping and stretching on images present.

In parallel with augmentation of the dataset, we have also started to evaluate the performance of several different CNNs on this task.

3.2. Finalizing Keyboard Localization Network

After experiments and evaluations on different networks, we will be able to decide which CNN we will use for keyboard localization.

3.3. Keyboard Separation Algorithm

After the keyboard can be correctly recognized, we will use perspective transformation to normalize the keyboard in video frame into a preset rectangle structure. Then we will work on an algorithm to extract boundaries of each key, namely, further separate the keyboard into individual keys on this.

3.4. Training Key Recognition Network

When individual keys can be localized and normalized, it is feasible for us to perform deep learning on each key to determine their conditions (pressed or not pressed). The quality of the eventual generated music heavily depends on the performace of this network.

3.5. Training Velocity Detection Network

A music piece without emotional variation is rigid and undesired. To detect the velocity information of a key pressing event, we propose a recurrent network to analysis relative frames in order to derive an approximate velocity value of this pressing.

3.6. Audio Generation

Upon we can derive all notes information, we need to reconstruct them as a MIDI file. During this process, we may need to smooth out some noisy notes such as those with exceptionally short duration. Besides, it may be necessary to normalize the velocity information we retrieved to avoid unexpected dynamic change.

4. Preliminary Results

Currently we've conducted some pre-processing on the dataset and got some result.

4.1. Dataset

The current dataset we have is a external dataset from previous people's research [1]. It consists of several muted videos and their corresponding midi files. The major problem is that the audio and the video do not start and end at the same time, and the offset is not provided. Hence, we have to manually label the time offset from the beginning of the video to the beginning of the audio. This task is accomplished.

While we are labeling the data, we have found several traits of the dataset:

- There are several pianists, each with their own playing style.
- All keys on the keyboard are covered.
- They have included some fake gestures.
- The playing times vary.

But there are several shortcomings as well:

- The fake gestures are almost identical, which is moving the palms above the keyboard horizontally back and forth after finishing playing. We doubt its ability of eliminating all false positives including those more subtle ones occurring while playing.
- A large portion of the dataset is pressing the keys from left to right one by one using only one finger. This playing pattern sheds less shadow and cause less interference than playing the piano in the real world. The low speed also reduces the difficulty to learn this specific pattern. In short, we worry about a potential overfit on simple playing patterns.

- Several audio files are corrupted. Either they are empty, or they have midi messages with messed up time informations. What's more, the leftmost key and the rightmost two keys are broken and are not recorded. One particular key somewhere in the middle is presumably broken as well.

- Only one or two from the 14 players plays meaningful pieces instead of random fiddling. We think more meaningful pieces should be included, since playing some common chords may generate common and critical graphic information that is uncommon among piano newbie's keyboard-scrubbing.

We believe that it is helpful to add our own dataset, where, beside solving these problems, we want to provide the following enhancements

- Provide the keyboard localization data for training the keyboard localization network.

- Provide the "strength" information while pressing the key for training the velocity detection network.

As for the keyboard localization network, we have already collected and labelled some videos from YouTube. As for the key recognition network, we plan to record our own videos after the milestone, in parallel with the training process.

5. Future Work

In the future, there are plenty of work to be done. As a conclusion of the ideas above, we will

- enlarge and augment our dataset,

- refine our network structure,

- conduct evaluation on the result

References

[1] M. Akbari, J. Liang, and H. Cheng. A real-time system for online learning-based visual transcription of piano music. *Multimedia Tools Appl.*, 77(19):25513–25535, Oct. 2018.