Project Proposal - ECE 285

Laiwei Wei
Electrical and Computer Engineering
A59010188

Hao Yu
Electrical and Computer Engineering
A14050325

Abstract

This project aims to propose a novel method of typing by replacing the traditional keyboard by a combination of a monocular camera and a paper with drawings. Functionalities of this project are mainly achieved by object detection, object tracking, finger identification and tracking algorithms. In this proposal, the problem definition part will cover problems the traditional typing keyboards are facing, significance of this project, and major components involved in this project. Next, the tentative method section will introduce the potential approaches to achieve this project. Lastly, detailed explanations of scheduled experiments and test plans, which will ensure the functionalities of the product generated by this project, will be included.

1 Problem Definition

The traditional QWERTY-layout keyboard was first introduced in 1868, and it is still used on almost all English dominant countries today [1]. Even though the traditional keyboard has been popular in the last century, it came with many issues. First of all, traditional keyboards are constructed by mechanical and/or electrical components, which means they are vulnerable towards liquid and abrasions. Traditional keyboards are also complicated in structures, which means once damaged, it is hard to repair them. Secondly, traditional keyboards are usually large in sizes. Bring a traditional keyboard around is not an easy task, and there is a high chance that the keyboards will be damaged in the transition process. Additionally, it is hard to customize the layout and size of keys.

This project, however, aims to solve all the potential problems of traditional keyboard. Keyboard proposed in this project allows users to use a piece of paper as the input device, a monocular camera as the sensing device. Users are able to customize the layout and size of keys by drawing on the paper. The desired final product of this project will be low cost, portable, and remain almost the same functionalities as traditional keyboards. This project could bring better user experience to customers, and even totally renovate the industry of typing devices.

2 Tentative Method

The tentative method sections will mainly focus on introducing potential approaches to achieve purposes stated in the problem definition section.

2.1 Required User Input

Even though this project allows users to draw anything they like on a paper, there are still some constrains for improved accuracy and robustness. This project expects users to draw distinguishing features, for example, triangles or hexagrams, on four corners of the paper. For characters or digits users would like to input, ideally they should be positioned as sparse as possible. Size of each character or digits can be diverse, while its size is expected to be large enough to be seen by the camera. Additionally, the color of paper is preferably to be white, and the colors of handwritten keys are preferably to be black. Then, the user is required take a picture of the paper under fair light source. After the user uploads the picture taken, this project will inform the user if the input is acceptable. Lastly, while using this project, the field of view of the monocular camera should cover the whole paper, and this project will promote to inform the user if the paper keyboard is ready to be used. Fingers of users are required to be visible by the camera while typing.

2.2 Handwritten Character and Digit Recognition

To recognise characters and/or digits drew by users, this project utilizes convolution neural network to train on the NIST dataset [3] for character recognition and MNIST dataset [2] for digit recognition. For pre-processing, this project will first generate a binary mask of the user input image by treating the color of paper as 0 and color of other patterns as 1. After getting the binary mask, it will specify regions of interests by labeling the same binary values in up/down and right/left directions. Then, with the specified region of interests and pre-trained weights, this project will be able to predict and store the meaning of the handwritten characters/digits. The pixel coordinates of the region of interests of all recognized characters/digits will be stored in a hashmap as keys, and the corresponding character/digit as value.

2.3 Object Detection and Tracking

For locating the customized keys on the paper on image stream received by the camera, this project applies the Scale-Invariant Feature Transform (SIFT) [4] algorithm to find outstanding features on both the user input image and the image stream received by the camera. After that, the generated features will be matched by a K-nearest Neighbours (KNN) algorithm by their descriptors generated by SIFT. By selecting 10 features with the highest matching scores, a fundamental matrix which transforms the input image stream to the user input image can be calculated. After the fundamental matrix is found, the key layout on the image stream can be transformed to the user input image.

2.4 Finger Detection and Tracking

The finger detection and tracking step is conducted through a ML pipeline introduced by MediaPipe [5]. With the palm detection model combined with the hand landmark, this system enables this project to have very accurate and robust detection and tracking of each finger.

2.5 Key Selection

With the object and finger detection introduced above, the final challenge to be solved is key selection. Since the movement of pressing a key is quick and tiny, and the project limits the sensing equipment to a monocular camera, it is tough to rely on the camera to determine if a finger has pressed one key or not. This proposal proposes two possible approaches to achieve trustworthy prediction of which key the user pressed. First, an image dataset of finger pressing keys will be recorded. Then, finger detection and tracking will be done on the dataset to find the pixel coordinates of each joint of the finger while pressing the key. The project can further train on that dataset for predicting which finger is pressing the key. Second, an algorithm can be created with keys that can be seen by the camera and the finger positions as input, and the potential key that is been pressing as output. Based on which keys are covered by fingers and the positions of tip of fingers, this algorithm might be able to predict which key is selected.

3 Experiments

3.1 Test materials

Following experiments are conducted after main part of virtual key board algorithm is well implemented, the materials required in the test contain:

Hand written nine-keys keyboard: written on a white paper. 9 keys in the keyboard represent the number from 1 to 9. Space between keys is sat to be 1 cm for clear identification.

Camera: Normal Camera with resolution of 1920×1080.

3.2 Procedure

The detailed procedures of experiments are divided into two parts.

First, the robustness of the keyboard will be tested. All tests conducted in this section will share the same paper keyboard and user input picture. This section will be composed by two parts. The first part will test if covering some features in corners will affect the final accuracy. The second part will test if moving paper slightly or impetuously will impact the output. The detailed explanations can be found in **Table. 3.2**

Robustness Test	
Number of Corners Hidden	Moving Frequency
0	0
1	0
2	0
3	0
4	0
0	0.1
0	0.2
0	0.5

Second, the universality of this project will be tested. This section of test is composed of two parts. In the first part, different customized keyboard layout will be tested. Features to be tested are characters, digits, mixture of characters and digits, size of characters, size of digits, and layout of keys. The purpose of this part is to verify that this project has the potential to work with customized keyboards. In the second part, three testers with different typing speed and fingers' size will be testing the paper keyboard. This is to ensure customers with different typing speed and size of fingers will have similar user experience.

3.3 Metrics

Following Metrics are adopted when conducting the experiment and treated to to identify the success of the experiment:

$$\textbf{Key-pressing Accuracy} := \frac{\text{number of correct identification}}{\text{number of valid key press}}$$

The desired key-pressing accuracy should be larger than 95%.

$$\textbf{Static Accuracy} := 100\% - \frac{\text{number of input received}}{\text{static time in seconds}}$$

The desired static accuracy should be larger than 95%.

References

[1] Bellis, M. (2020, January 13). Why your computer keyboard has a QWERTY layout. ThoughtCo. https://www.thoughtco.com/history-of-the-computer-keyboard-1991402, (retrieved May 2, 2022).

[2] Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine, 29(6), 141–142, (retrieved May 2, 2022).

[3] William E. Acree, Jr., James S. Chickos, "Phase Transition Enthalpy Measurements of Organic and Organometallic Compounds" in NIST Chemistry WebBook, NIST Standard Reference Database Number 69, Eds. P.J. Linstrom and W.G. Mallard, National Institute of Standards and Technology, Gaithersburg MD, 20899, https://doi.org/10.18434/T4D303, (retrieved May 2, 2022).

[4] Ng, P. C., Henikoff, S. (2003). SIFT: Predicting amino acid changes that affect protein function. Nucleic acids research, 31(13), 3812–3814. https://doi.org/10.1093/nar/gkg509, (retrieved May 2, 2022).

[5] Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., ... Grundmann, M. (2019). Mediapipe: A framework for building perception pipelines. arXiv preprint arXiv:1906.08172, (retrieved May 2, 2022).