

# 1 data pre-processing

## 1.1 data introduction

The American sign language has 28 signs for each letter and one sign for space and another sign for delete. The "space" and "delete" here do not refer to the word space or word delete, it means the space between word and word and the action to delete last sign. When you use American sign language, you use the sign of space as a separator and the sign of delete for correcting your wrong signs.

Our data is downloaded from kaggle with 3000 images for each asl and another 3000 for the background of the images. The images contain a hand with background. In order to get rid of the influences of the background, we need to pre-process the data. In other words, change the color of the original images into black and white and remove the background. For the result, we expect to have the final images with only silhouette of the hands.



Figure 1: Signs for H and background

## 1.2 pre-processing methods

We wrote three functions to pre-process the data including background removing, denoising, and size unifying.

### 1.2.1 background removing

**First step: color changing** First turn the images into grey ones based on RGB formula. We calculate the level of grey from the red, green, blue tunnels of the original images.

$$Gray = \lceil (R * 30 + G * 59 + B * 11) / 100 \rceil$$

There are 256 levels of gray, we set a threshold  $L$  based on lighting conditions.

**Second step: generate binary matrix** In the 256 levels of grey, 255 stands for white and 0 stands for black which means whiter the color, larger the pixel. Then we generate the binary matrix. If pixel at  $(x, y)$  that is bigger than  $L$  which means the color that is lighter than the color we choose, the  $matrix(x, y)$  will be 1 and otherwise 0;

$$matrix(x, y) = \begin{cases} 1 & pixel(x, y) \geq L \\ 0 & otherwise \end{cases} \quad (1)$$

**Last step: remove background** Then we minus the matrix of images and background and convert the matrix into images to get the black and white image with background removed.



Figure 2: background removed image for H

### 1.2.2 denoising

**First step: area counting** For every pixels of the images, we count the black points around it and plus the point itself. For points on different position, it has different number of points around it.

$$points\ in\ area = \begin{cases} 4 & upper\ or\ bottom\ left\ or\ right\ vertex \\ 6 & first\ or\ last\ lineor\ left\ or\ right\ side(not\ vertex) \\ 9 & otherwise \end{cases} \quad (2)$$

If the black points in the area is smaller than the value we set, it means the point is a noise point. Basically we set the value to 5, which means to remove the vertices and the black points which has less than 4 points near it. We set the color of the point to white. After this step, it is more easy to erase the noise part.

**Second step: cut area** After removing the noise points, it is more easy to identify a certain area to remove to get the silhouette of the hand. We simply set the color of certain area to white And then cut the white border of the images.

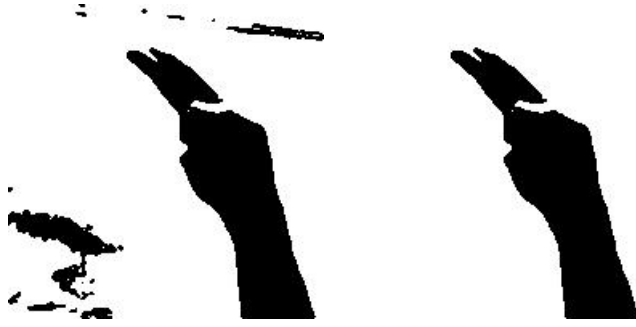


Figure 3: denoised images for H

### 1.2.3 size unifying

The last one is to convert the cut images into same size and the hand is in the center of the bottom. In this way, we can convert all the images into 200 multiply 200 matrix and then fit the model.