

Project 1

Name: Vivek Singh

UBID : Vsingh28

I have implemented this code with the help of 3 main modules –

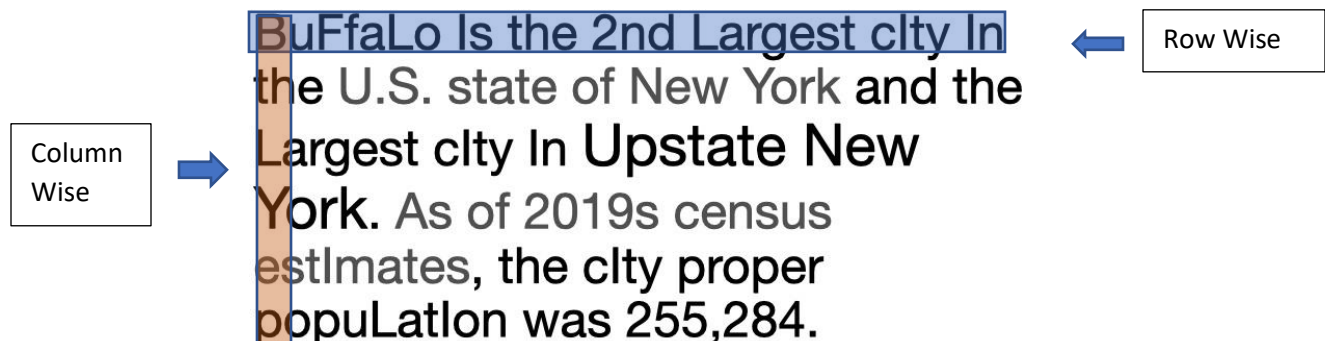
- **Enrollment** – In this function, we are using the template files to read and extract features from a given image.
- I have used the ORB method to detect the descriptors in the given image or template and then saved these descriptors for future comparisons in the return value of the function called “key name” (name) and desc[array].
- **Detection** – This function is mainly used to detect the feature from the test image.
- Here, I have used the histogram approach to find the features

Row wise:

- First, I have tried to scan the image row-wise and then checked if the row is having any pixel which is background (by taking a threshold greater than 245).
- Then these row indices are stored in an array. For the row, whenever I find that there is a gap in the index of rows stored in the array, that means there are foreground rows in between them and then I can create a sub-image from these foreground rows.

Column wise:

- I use a similar process for the columns. I have sliced the white space columns and then we get the character bounding box.
- I have stored these bounding boxes in the format of a dictionary with the values inside it being the bounding box/extracted character itself, starting row, starting column. Height and the width.
- I can get the starting and the ending address of these bounding boxes when after we filter out the whitespace columns from each sub-row image.
- Now I have used the orb operator for each of the bounding boxes and then we need to compare the descriptors of the box with the template ones (descriptors saved in a file



- **Recognition:**
- Once the feature got extracted from detection we can pass them in the recognition function with the enrolment feature.
- After calling both the arrays, I have compared them with each other and set the condition:
If corresponding array matches then put "FOUND" otherwise "UNKNOWN"