

OPTICAL CHARACTER RECOGNITION

Enrollment:

The enrollment function takes input a list of characters. For feature extraction the function utilizes Scale Invariant Feature Transform (SIFT) present in OpenCV package. Before, extraction, each image is passed through the threshold function. If the pixel value is less than 190, it makes the pixel value 0 otherwise makes the pixel value 255. Then with the help of SIFT feature extraction, keypoints and descriptors of each character are obtained and stored in a list. The SIFT feature extraction do not return any keypoints if there are no distinct features in an image. So, in case, there are no keypoints or descriptors present for a character we bypass that character. The function returns an array of keypoints and descriptors of the characters passed as input.

Detection:

The detection function takes the test_img as input and returns the coordinates of candidate characters and matrix of the candidate characters with their pixel values. The detection function is based on two pass connected component labelling (CCL) algorithm. First, the test_img is passed through a threshold function which returns an image containing only two values (0,255); 0 if pixel value is less than 190 and 255 otherwise. The implementation of two Pass CCL algorithm is as follows:

First Pass: In the first pass, motive is to label all pixel values that are in foreground i.e., having pixel value as 0. We maintain a copy of labelled image and use threshold image obtained for comparison. Initialize starting label to be 1. If pixel value is 255 it is background and is bypassed.

If the pixel value is 0 in threshold image and,

- i) If the corresponding pixel above is not 0 and to the left is also not 0 then the labelled image pixel is assigned new label.
- ii) If the corresponding pixel above is also 0, then the labelled image pixel is assigned label of above pixel.
- iii) If the corresponding pixel to the left is also 0, then the labelled image pixel is assigned label of the left pixel.

Second Pass: In the second pass, the detection algorithm merges the labels that are connected. It checks for every pixel if the pixel above or to the left has a label and assigns the min of the label. It also checks if down or to the right of the pixel has a label and assigns the min of that label.

After this, we assign sequential labels from left to right and move towards finding the coordinates of connected objects. As the no. of unique label values will be the no. of candidate characters detected, for each label we save the x-coordinate which is the row and y-coordinate which is the column of the pixel value that has that label. The min of x-coordinate and y-

coordinate of each label represents the top-left coordinates of the detected character. The height and width of each label is calculated as the difference of max & min of y-coordinate and x-coordinate respectively. The coordinates of each label as (x, y, h and w) along with the pixel values of the candidate character is then returned as an array after padding.

Before Detection:

**BuFfaLo Is the 2nd Largest clty In
the U.S. state of New York and the
Largest clty In Upstate New
York. As of 2019s census
estlmates, the clty proper
popuLatlon was 255,284.**

After Detection:

**BuFfaLo Is the 2nd Largest clty In
the U.S. state of New York and the
Largest clty In Upstate New
York. As of 2019s census
estlmates, the clty proper
popuLatlon was 255,284.**

Recognition:

The recognition function takes input the list of character images, the candidate images 2-D arrays and their coordinates obtained from detection and the keypoints and descriptors of enrolled characters. It returns an array results which has values as {"bbox": [x (integer), y (integer), w (integer), h (integer)], "name": (string)}. The recognition function loops over every candidate image obtained from detection and gets its descriptors and keypoints. Then for every enrolled character, it calculates the sum of square distances between each descriptor of an enrolled image and the descriptors of candidate character. It then calculates the ratio distance between first best match and second-best match of that descriptor. This ratio distance is calculated for each descriptor of enrolled character w.r.t candidate character descriptors. It then checks the ratio distance, if it is less than 0.62 it is considered a good match. If the matching descriptor count ≥ 4

we consider that enrolled character is a match to the candidate otherwise we consider it “UNKNOWN”.

CONCLUSION:

With this approach, using sift for enrollment, two pass component algorithm for detection and SSD between descriptors for recognition, the OCR achieves **0.677 F1 Score** for the given test image.