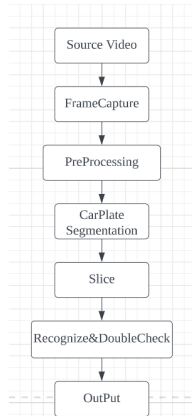


## Final Flowchart

For all the input videos, they should be first captured in the fixed speed.

After the capturing process, we resize the image to [500, 500] and find the HSV mask of it. Then, we use canny to find the edges on the grayscale masked image and use contour to find the possible plates. After that, we filter all the contours with physical rules like the width/height ratio of plates and segment the area with plate. Since the plate now could be rotated, we use Hough to find the lines in it and generate the rotationMatrix with the angle. The rotated segmented plate will be cut again with contour based cutting method and denoised by a morphology based method to get a rotation-cancelled, denoised final plate result.

For any segmented plate, we slice them with the slicing pipeline. This is powered by a floodfill algorithm, a contour based connecting component finding method and a denoising method which is based on morphology. After this process, we will get 6 black and white images each contains one character.



Finally, the characters comes to the recognition pipeline. Our recognition is mainly based on a SVM model and a SIFT algorithm double checker. Then we get the final output.

## Evaluation

As this is the final poster of our project, we want to find out how actually our programs perform under the standard estimation of the courses.

Thus, we choose the whole 3-minute training video as our test case. As our training set doesn't encounter with the video, the overfit problem may be avoided. We choose to get the frame

## Evaluation

once per half second, and we output a csv file.

Finally, we use the evaluation program to calculate our final accuracy.

The accuracy is 74% for categorie I & II and 0 for the rest.

## Edge cases handling

Our system handle edge cases mainly in four aspects. Firstly, we increase the accuracy of the car plate segmentation. (Compared with the second plate, the first must be easier to recognize.



Secondly, we improve the training set of the SVM model. By adding more data with variation and different resolution to edge cases like 8 and B, K and X, we get a higher accuracy on the final result.

Moreover, we use the plate rule to check the result. For example, in 98-THD-4, if the 8 is recognized as B, it should disobey the plate rules then it should be corrected to 8.

Finally, we use a SIFT algorithm based prediction model to double check the results containing edge cases. If the result of SVM doesn't match any legal car plate pattern like D-XXX-DD, we will use SIFT to recognize it again to try to get a more suitable result.

## Scene change handling

We deal with scene change by handling different input.

For C1 images, our system should detect 1 possible plate in it. Then it should be fine.

For C2 images, if the image only contains part of the car plate like this:



Since the number of the recognized characters is lower than 6, we neglected this output.

For C3 images, the contour will tell us there are 2 possible plates in the original image. Then we will change to the double plate pipeline which dealing with the plates one by one.

For C4 images, because we didn't implement the Contour-only localization, this part we will get no possible plate area after the Color segmentation part, then no output.

## Merging Output

Since there might be multiple outputs for the same plate, it is important to merge the outputs. Our strategy is dealing with output step by step.

For C1 images, we choose to output all results of the recognition.

For C2 and C3 images, we choose to output all results only if there is more or less than 6 characters.

For C4 images, there might be output(Because our HSV range is quite wide), we choose to output all the things.

## Future Improvements

### & Weakness

This project is finished in such a short time with lots of accidents, there must be some weakness points. The biggest problem is that we still cannot handle edge cases well. 8 and B are too close to determine. Also for the slicing part, this algorithm cannot deal with rotated pictures at all.

For the improvements, it could be done in three parts. Firstly, we can implement the Contour-only segmentation. This will make foreign plates recognition possible. Secondly, we can dealing with multiple results with scoring system. Only the result with highest score will be the actual output. Finally, we can use Neural Network to do the character recognition. Though it is not allowed in the project this time, we strongly believe that there is a high potential in the field of machine learning.