



License plate recognition

New scene transitions

Below, we define a scene as the total number of frames that capture a single car. To detect when a scene has changed, we first apply the localization and recognition algorithms to each frame, saving the resulting plate string(s) along with their corresponding frame numbers and timestamps. Next, we group plates that may belong to the same scene based on two criteria: the difference score between their recognized strings (measured as the number of differing characters at the same positions) and the difference between their frame numbers. These two thresholds are provided as parameters to the method.

Recognition output merging

The output merging process occurs immediately after grouping the plates based on the scenes they belong to. To determine which plate number our algorithm associates with each group or scene, we employed a majority voting procedure. First, any strings with a length different from six were discarded. Then, for each of the six positions in the final output, we selected the character that appeared most frequently at that exact position among all the strings associated with the current scene. Finally, the resulting plates were validated against a Dutch license plate format to ensure compliance and to correctly place the dashes

Weaknesses

One possible weakness of our system is its inability to handle out-of-plane rotated license plates. As a result, the **localization process** becomes slightly less accurate, which affects the final output. Additionally, we use color segmentation that relies on the fact that license plates are yellow. So foreign license plates are likely not detected by our localization algorithm. Regarding the **recognition part**, a potential issue might be the inaccuracy of the image denoising techniques. By this, we mean that for some plates, the morphological operations do not produce the desired results, instead yielding an output that is either too noisy or overly cropped.

Future improvements

Certainly, future improvements can be made to our system to enhance its overall performance. One potential addition is a new character recognition method, based either on a machine learning algorithm or a more advanced version of a SIFT descriptor. This improvement would likely also address cases where the plate is not fully visible or has very low resolution. For the localization part, extending the system to support non-Dutch plates would undoubtedly be an upgrade. This could be achieved through the implementation of a robust edge detection algorithm and other similar techniques.

Our scores

Category 1	83.3%
Category 2	90%
Category 1+2	85%
Category 3	42%
Category 4	0%

Edge case handling

Edge cases involving similar characters are handled by tracking the XOR score for each character and considering not only the best match but also the second best, provided they are within a threshold of each other defined by the `accuracy_rate` parameter. Additionally, the Dutch plate validity check eliminates plates that do not conform to a specific format. As a result, in scenarios where an incorrectly recognized character leads to an invalid letter-number alternation, the plate is excluded by our algorithm.