**License Plate Recognition and OCR project – Chris, Yun, Phuong**

**Motivation**

We want to create a program that can detect the license plate from an image and crop it for identification.

**Data**

The dataset contains 1821 images, captured by a license plate reader (LPR).

* “Context” image showing the vehicle, each image is 800 x 600 pixels
* “IR\_patch” image showing the cropped license plate
* “XML” file contains metadata

1. **License Plate Recognition Challenges and Approaches**
2. **Lack of Homogeneity**

Each image has different background, brightness depends on what time the picture was captured. This is difficult because one single approach of preprocessing cannot recreate the same threshold that we want.

1. Approach 1: create presets for different categories. For examples, (Light Car, Light Background), (Light Car, Dark Background), (Dark Car, Light Background), (Dark Car, Dark Background)
2. Approach 2: Region of Interest Detection
   1. We applied the basic image processes like gray-scaling, thresholding, contouring, etc. However, the key element for this approach was motion blur. If you applied both horizontal and vertical motion blur on the image, you will see the most darkest area which is the most condensed region.
3. **Identify Contour Associated with License Plate**

After isolating the car and getting the contours, the next challenge is identifying the contour of the license plate. There are many rectangle shape objects such as rear windows, backlights, signs; and sometimes the license plate does not have a clear outline but blends in with white car.

* 1. Approach 1: License plate has unique and consistent dimensions, so we add the dimension thresholds to contours before seeking a match.
  2. Approach 2: We used histogram equalization to get greater contrast. A good image will have pixels from all regions of the image, but brighter image will have all pixels confined to high values. While general histogram equalization loses a lot of details, **adaptive histogram equalization** is helpful because this divides an image into small blocks or "tiles". Then each of these blocks are histogram equalized as usual.

1. **Optical Character Recognition Challenges and Approaches**

Challenges:

1. **Noise**

We used motion blur to remove noises. If you apply motion blur horizontally on the image you will see the darkest region where the actual letters are aligned. Then, you can easily capture the plate number region with thresholding and masking the image.

1. **Tilt**

We applied Radon Transform to correct the angle rotation. It basically get the right angle from the origin of an image then calculated the rotated angle to fix it.

1. **To increase the accuracy**

We included all the images through out the image process, and OCR on each one of them and see which string is the most license plate number-like.

**Results**

In regards to the object detection section of the project we took a random sample of images, and after running our code, verified that we were able to detect 45% of the license plates. Unfortunately, we were unable to set up a cost function using the xml files to derive an MSE for our bounding boxes relative to those provided.

In regards to the OCR section of project we believed that we obtained decent results for the majority of the license plates. The major issues included preprocessing leading to some of the letters not coming out fully. That being said we believe that we received decent results.