License Plate Matching Using Neural Networks

Kelvyn SOSOO (GMU)  David OUYANG (CSUST)  Mengjun WANG (CSUST)
Mentors: Lee HAN (UTK) & Kwai WONG (UTK)
Overview

- License Plate Recognition (LPR) technology is used to gather vehicle location data
- Location Data includes instances of Amber Alerts, Toll Roads Speed/Travel Time, etc.
- The License Plate Matching (LPM) method incorporated includes a 97% match rate of vehicles, and a 60% read accuracy
- Programs Used: Python, Matlab

GOAL: Raise the 60% by using Image Processing. Find a new measure to matching plate by using supervised learning.
How It Works

Capture images → Image recognition → Plate matching
Procedure

Screen the License Plate images

Image Processing to segment every Character

Develop New method of matching

Neural network training
Image Processing

Step 1: Image binarization

Method: cv2.THRESH_OTSU
Step 2: Read the number of black pixels in the y direction

\[
\begin{align*}
\text{np.argmin(row_nz[0 : \text{floor(len(row_nz)/2)])} &= 59 & \text{row_nz(59)} &= 43 \\
\text{np.argmin(row_nz[\text{floor(len(row_nz)/2)} : ])} &= 95 & \text{row_nz(95)} &= 45 \\
(59, 43) & \quad (95, 45)
\end{align*}
\]
Step 3: Read the number of white pixels in the x direction

KEY POINT & CUT POINT: [33, 40, 54, 72, 86, 104, 120, 150]
Final Outcome
Supervised Learning: Neural Network

- Previous slide presented the outcome of Character Segmentation
  - It is very time consuming to transfer the characters to the proper label/category
- Instead of spending countless hours manually moving files, Data Augmentation was implemented
- Categories included A-Z and 0-9
Convolutional Neural Networks

Performance

- After four epochs, the model was able to reach a validation accuracy of 95.18%
Next Step

- Think of a way to store License Plates with this method
- Find way to string together characters for Matching step
Plate Matching

Ex: the tag of a license plate is 4455HZ
LPR 1 reads the plate as 44S5H2,
LPR2 is 4455HZ

Constraints

1. The travel time of LPR1 to LPR2
2. The ED of plate 1 change to plate 2
3. The conditional probabilities of character transition

Less ED, higher probability, in the time period

Goal: To judge whether Different plate characters are from the same car
Edit Distance Operation

\[ d(x \rightarrow y) = \min\{d(i - 1, j - 1) + \gamma(x_i \rightarrow y_j), d(i - 1, j) + \gamma(x_i \rightarrow \epsilon), d(i, j - 1) + \gamma(\epsilon \rightarrow y_j)\} \]

Association Probability

\[ p(y|x) = \frac{p(x,y)}{p(x)} = \frac{\sum_t p(x,y|t)p(t)}{\sum_{y,t} p(x,y|t)p(t)} \]

\[ p(y|x) = \sum_t p(y|t)p(t|x) \]

Self-learning Matching

C(M) represents calculate association matrix from a set of matches M

1) \( k = k + 1 \);
2) \( M_k = M(C_{k-1}) \);
3) \( C_k = C(M_k) \);
4) Stop if \( C_k - C_{k-1} < \epsilon \).
Research Plan

1. Work on ED algorithm, self-learning algorithm of matching

2. Work on the matching part and weight function by python

3. Calibrate the parameters and validate them

Done

1. Read papers

2. Determine the project frame

Next step

Learn matlab and transform the matlab code into python code
Challenges

Challenges Already Faced:
- Sufficient Training Data
- Character Cutting Details
- Clearer Images
- Handling of Similar Characters

Foreseeable Challenges:
- Adapting Matlab Code
- Meshing Network and Matching
THANKS FOR LISTENING, ANY QUESTIONS?