CATCHING SPEEDERS VIA MOBILE PHONES AND MACHINE LEARNING: AN OPPORTUNITY TO IMPROVE SPEED ENFORCEMENT

Jahnavi Malagavalli
Graduate Research Assistant
Department of Computer Science
The University of Texas at Austin
jahnavimalagvalli@utexas.edu

Tong Wang
Graduate Research Assistant
Chandra Department of Electrical and Computer Engineering
The University of Texas at Austin
tong.wang@utexas.edu

Kara M. Kockelman, Ph.D., P.E. (corresponding author)
Dewitt Greer Centennial Professor in Engineering
Department of Civil, Architectural and Environmental Engineering
The University of Texas at Austin
kkockelm@mail.utexas.edu Tel: 512-471-0210

Eric Sean McDonnell
Undergraduate Research Assistant
Department of Civil, Architectural and Environmental Engineering
The University of Texas at Austin
ericmcd@utexas.edu

ABSTRACT
Smartphone cameras and computer vision (CV) can be used to assist public agencies in enforcing traffic laws and improving safety. This paper documents automated enforcement applications around the globe for excessive speed, noise, illegal parking, driving through a red light and other forms of reckless driving, while illuminating how non-stationary, hand-held cameras can be used (as well as the millions of traditional, stationary cameras that already exist alongside roadways). Speeding is a significant cause of car crashes, and despite existing laws and automated speed reduction methods, accidents related to speeding still occur worldwide. While fixed camera surveillance is one automated method for reducing speed, its effectiveness is limited to specific locations. Violators tend to accelerate once they pass these fixed camera sites. To address this issue, a CV technology has been developed using existing open-source methods, which can estimate vehicle speed and identify license plates from mobile phone recordings. This innovation
enables the identification of speed violators from any location, especially in areas without stationary cameras. By utilizing cellphone videos provided by the public, it becomes possible to report violations to law enforcement and improve overall road safety. The use of computer vision techniques can assist in providing necessary information to verify traffic violations. This technology was able to identify 47% of license plate text and correctly identify vehicle speeds in 5 out of 6 recordings. The license plate detection can be further improved by fine-tuning the license plate detection model and super-resolution model on license plates. It is also observed that text detection is comparatively good when the video of the car is recorded nearby. The paper found that existing open-source models for make and model are not accurate enough, and fine-tuning the model on more recent car make and model data is likely required to improve performance.

**Keywords:** Speed estimation, Mobile camera video recording applications, Automated enforcement, License plate recognition

**BACKGROUND**

Speeding is a major contributor to crash counts, severities, and costs in the US and elsewhere (Richard et al. 2014; National Safety Council 2022; Macias et al. 2023). Speed limit laws, active enforcement, road design strategies, speed limiter/governors, speed enforcement cameras and intelligent speed adaptation in vehicles help avoid excessive speeds (Hirst et al. 2005, Sadeghi et al. 2016). Average speeds on roadways increased during COVID, as drivers appeared to drive more recklessly because of lighter traffic and higher speed limit allowances in many places (Tucker and Marsh, 2021; Aashto 2022, Speed Camera Program 2023). It is presumed that police pursuits increase the amount of crashes that occur, as speed-limit violators try to out-race police and may end up crashing and/or killing someone (Rivara and Mack 2004; Today 2015). As such, replacing human interaction with automated enforcement is safer for everyone. Automated speed enforcement (ASE) in the UK, Australia, New Zealand, Canada, New York City, Washington D. C., Chicago, and other cities, states, and countries currently relies on radar devices to identify and still camera shots (for automated license plate reading) to record traffic violations, read license plates, and trigger notices of warnings and fines to associated vehicle owners (Governors Highway Safety Association, n.d.).

Overall crashes fell from 9.2% to 51% after ASE implementation, across 5+ nations (Miller et al. 2016). Some systems with singular cameras face issues where drivers who are aware of the position of the camera will deliberately slow down and speed up again after passing the camera, a so-called “kangaroo effect” (Elvik 1997). To get around this issue, other systems being used in places such as the UK or Australia are designed with multiple cameras to operate over a larger stretch of road, calculating the average speed of the driver (Delaney et al. 2005; Owen et al. 2016; Bates et al. 2016). However, such systems run into issues where the exact or momentary speed of a vehicle is unknown.

Many methods have been proposed to reframe how cameras are advertised to the public in order to make such surveillance more acceptable, although some people remain skeptical (Raty 2010; Feldstein 2019; Ralph et al. 2022a & 2022b). And police ticketing rates have fallen, thanks to
decreasing budgets/insufficient funds, automated enforcement (for red-light running and speeding) in some cities and states, and rising speed limits (Tuttle 2015). Around 20 US states and many other countries use automated enforcement. Another technology to limit speed is speed limiters or governors, they are widely used to restrict the maximum speed of trucks in various countries. For example, Canada's speed governors limit trucks to 65 mph, while in Australia, trucks are restricted to 65 mph or less depending on the region (Canada 2013; Buchs 2022). The USA is also planning to require speed governors for trucks, with a proposed limit of 70 or 75 mph (FreightWaves, n.d.).

**Automated Enforcement Around the World**

Automated vehicle identification for traffic law enforcement is useful in many settings. These include automated collection of tolls and fees, identifying illegally parked vehicles, stolen vehicles, loud vehicles, and high emitters (dirty vehicles), and enforcing other transportation-related laws (like speed limits and red-light compliance). Automated parking enforcement using license plate recognition systems, in contrast to crash reduction, offers the potential to improve efficiency and save time in identify parking violations compared to time-consuming manual check of parked vehicles for valid parking permits. For instance, in a Texas city, a single police officer scanned 48,101 plates using such a system, resulting in 255 traffic citations, 26 suspended licenses, and other violations in the course of 96 hours of use over 27 days (Wood, 2021). Automated cameras and CV techniques using license plate recognition systems are being used for parking enforcement in various locations worldwide, including Amsterdam, Melbourne, Texas, and California. These systems can be installed on enforcement officer cars or at parking lot entrances and exits to scan and identify vehicles violating parking regulations. Once a vehicle is found in violation, the system can alert parking enforcement officers or automatically issue citations. The use of these technologies has the potential to increase compliance with parking regulations while reducing the need for manual patrols (Van Den Berg, 2014; Dinh and Kim, 2016; Wood, 2021; Kadah, 2022). Overall, automated parking enforcement using license plate recognition systems is a promising solution for efficient and effective parking enforcement, with potential benefits in terms of reducing manual patrols and increasing compliance with parking regulations (Parking Network, 2021; Mulloy, 2023).

Furthermore, excessive noise and emissions are significant contributors to health problems such as sleep disturbance, cardiovascular, and psychophysiological issues, and can even shorten life expectancy, with vehicles, particularly motorcycles, being a major source of these problems globally (Perna et al. 2021; Timperley 2022). To address this issue, some cities like Paris and New York have implemented automated noise enforcement using noise radars to detect excess noise and cameras to capture the license plate of the offending vehicle, allowing for fines to be issued to the vehicle owner (Esteban 2019; Official Website of the City of New York 2022). In addition, various tools can be used to identify high-emission vehicles, including remote sensing devices that detect pollutants in the exhaust using infrared and ultraviolet technology, on-board diagnostics that monitor engine performance and emissions control systems, and video analytics that analyze factors like vehicle speed (Huang et al. 2018; Oluwaseyi and Sunday 2020; Valido et al. 2022).
In the safety realm, many nations now subscribe to a Vision Zero, whose goal is zero road deaths (and near-zero debilitating injuries) (Tingvall et al. 1999; Marusin et al. 2018). Avoiding excessive speed and other illegal driving maneuvers (like left turns on red, wrong-way travel and rapid lane changes in congested traffic) are one of the best ways to lower severe-crash counts, and could save trillions of dollars a year around the globe (with US crashes alone currently costing nearly $1 trillion annually, or roughly $3,000 per capita per year (Liu and Subramanian 2009). In Asia, at least 8 countries rely heavily on automated enforcement predominantly to enforce speed limits, with another 10 Asian nations using a mix of automated and manual enforcement (UN.ESCAP, 2020). Nations like Israel (Factor et al. 2022), Malaysia (Hamzah et al. 2013), and China (He et al. 2013, Mozur et al. 2022) use automated enforcement. In Hong Kong, speed cameras are used in conjunction with manual policing. While the cameras themselves are more effective in catching and reprimanding reckless driving, the threat of verbal reprimand adds an “embarrassment factor” which can be a deterrent to some reckless drivers. However, with both manual and automatic enforcement a “kangaroo effect” – where drivers slow down when being watched only to speed up again after leaving the camera or enforcement officer’s view - is seen when drivers knew the location of cameras or police beforehand (Chen et al. 2020).

As of February 2023, 22 U.S. states, plus Washington D.C. and at least 35 Texas communities with preexisting contracts (The Texas Tribune 2019, KVUE 2022) (Figure 1a) use red-light cameras, and 18 states plus Washington D.C. are using speed cameras (Figure 1b).

Figure 1a: Red Light Camera Use in US States
Figure 1b: Speed Camera Use in US States

Figure 1. Automated Enforcement Laws across US States (IIHS 2023)

In contrast, 7 U.S. states have outlawed red-light cameras and 7 have outlawed (agency-owned and operated) speed cameras (Figure 1b). Fortunately, privately owned cameras can record images (and video) in all public settings in the US, including the public rights of way (typically roadways), and those images can be used to support traffic law enforcement and crime-reduction. The following section describes the CV techniques for vehicle detection, identification, and speed inference.
VEHICLE IDENTIFICATION AND SPEED INFERENCE

Object detection is a well-studied CV topic (Padilla et al. 2020; Srivastava et al. 2021; Ren and Wang 2022), and includes vehicle detection. Detection algorithms can be classified as one-stage (You Only Look Once (YOLO) and Single Shot Detector (SSD)) or two-stage detectors (Region with Convolutional Neural Network (R-CNN) and faster R-CNN). Two-stage detectors use two neural networks: one to find regions of interest (in each image or frame) and another to classify those regions, so they take more time to deliver higher accuracy (Kim et al. 2020). YOLO is a popular method for object detection due to its efficient performance (Kim et al. 2019; Mao et al. 2020; Akhtar et al. 2022). It is also used to detect various traffic violations such as riding a bike without a helmet, not wearing a seat belt, and red light signal jump (Ravish et al. 2021). The latest version of the YOLO algorithm, v7, appears to outperform two-stage detectors in terms of both time and accuracy (Wang et al. 2022). This paper uses both YOLOv7 and Faster RCNN frameworks.

Vehicle or object tracking algorithms use deep learning (a series of neural networks) to predict object positions across video frames using both spatial and temporal features. Tracking tools generate bounding boxes to improve object detection and identification. DeepSort (Wojke et al. 2017) is a popular tracking algorithm that extends the Simple Online and Realtime Tracking (SORT) (Bewley et al. 2016) technique by using two association matrices (for object velocity and appearance) to create downstream-frame boxes, via Kalman filters. This paper uses both DeepSort and its further improvement, called StrongSort (Du et al. 2023) for faster and more accurate vehicle tracking.

License Plate Detection and Recognition

The most common way to identify unique vehicles is via Automatic License Plate Recognition (ALPR) algorithms. It is a three-step process: first, the license plate is localized, and then character segmentation and recognition techniques are applied to extract the text. License plate localization is done in two ways: traditional, handcrafted feature-based (Du et al. 2013) and deep learning-based methods (Laroca et al. 2019), with object detection techniques like YOLO (Zhu et al. 2022; Akhtar et al. 2022). Current techniques use different YOLO models to extract vehicles and license plates, respectively. Text recognition on these license plates is accomplished through segmentation (a two-step process involving a segmentation model and a recognition model) or segmentation-free methods (a one-step process). There are several optical character recognition (OCR) techniques available (EasyOCR, 2021; Kuang et al. 2021; pytesseract. 2022), which also preprocess images to boost the chances of recognition - like de-skew, smoothing edges, and converting images to black and white (Karandish, 2019). ANPR improvements are mostly hindered by poor image quality and low resolution cameras (Hamdi et al. 2021). There has been a lot of work on improving image quality (Dong et al. 2016, Hamdi et al. 2021), and general adversarial networks (GAN) have proven successful in super resolution reconstruction, and can be leveraged to perform image denoising pre-processing (Hamdi et al. 2021). There are publications that present the entire pipeline used for ALPR on fixed camera videos (Silva and Jung, 2020;
Zhang et al. 2021), including via drone recorded videos (Kaimkhani et al. 2022). However, no work appears to have been published on the inference of ALPR from mobile phone video recordings.

**Recognizing Vehicle Make, Model, and Color**

Many agencies and businesses rely on humans reviewing video to identify vehicle make and model as well as obscured or hard-to-read plate information. Make, model and color help agents identify vehicles whose plates are obscured or damaged (Lee et al. 2019; Hu et al. 2022).

Identifying make and model is challenging due to the similarity between different vehicles, as shown in Figure 2 (Hsieh et al. 2014b). To address this, algorithms reflect at vehicle dimensions, shapes and vehicle logos (Yang et al. 2013) - with special detail typically visible on a vehicle’s front view and sometimes also its rear view (Saravi and Edirisinghe 2013; Hsieh et al. 2014b; Baran et al. 2015). with special detail typically visible on a vehicle’s front view and sometimes also its rear view (Saravi and Edirisinghe 2013; Hsieh et al. 2014b; Baran et al. 2015). These methods tend to use support vector machines (SVMs) (Hearst et al. 1998) or measure the closest distances between feature vectors to estimate vehicle models. Researchers are also using compact CNNs to identify make and model more efficiently (Dehghan et al. 2017; Lee et al. 2019).

It is challenging to recognize vehicle color accurately from videos due to weather conditions, image quality, and vehicle striping patterns (Rachmadi and Purnama, 2015). Some studies have utilized SVM (Chen et al., 2014) and histogram-based approaches (Dulee et al. 2010) approaches to identify vehicle color. Others use CNNs to classify color (Rachmadi and Purnama 2015, Fu et al. 2020, and Hu et al. 2022).

![Figure 2. Similar Vehicle Models: Toyota Highlander 2023 and GMC Terrain 2022 (Edmunds. (n.d.).)](image)

**Speed Estimation**

Various CV techniques are proposed to estimate vehicle speed (Llorca et al. 2021). In general, these techniques have 3 high level components as shown in Figure 3. They start by taking video recordings and external environmental parameters like scale factor. Then detection and tracking algorithms are used to calculate the distance traveled by the vehicles in the 2D domain. Finally, the speed of the automobile is calculated with an estimated distance in the real world (calculated with a scale factor) and the time difference between the frames.
There are a few prominent methods implemented in the existing literature. Real-world distance estimation is a difficult task as it is usually computed with a set of assumptions, such as all roads being flat. Methods for distance calculation include – homography-based (Kim et al. 2018), augmented intrusion line-based, pattern- or region- (Dahl and Javadi, 2019) based, or based on prior knowledge about the actual dimensions of some of the objects, such as the size of the vehicles (Moazzam et al. 2019). These restrictions are reduced when using stereo vision (Jiang et al. 2019).

Figure 3. High-level components of vision-based speed estimation (Source: Llorca et al. 2021)

In almost all the methods, calibration is an essential part in estimating the speed of the vehicles, as calibration helps to calculate the intrinsic camera parameters (sensor size and resolution, focal length) and the extrinsic parameters (location with respect to the road surface). A widely used method for camera calibration is with the help of vanishing points (Orghidan et al., 2012). Vanishing points (VPs) can be estimated by various algorithms, which can be separated into two groups. The first group, known as geometry-based methods, leverages the fact that the VPs occur at the intersections of straight lines. These methods estimate the VPs by, for example, associating lines to VPs (Tai et al., 1993; Gamba et al., 1997; Wu et al., 2021; Feng et al., 2010; Tardif, 2009), line clustering (Schaffalitzky and Zisserman, 2020; Bazin et al., 2012; Barinova et al., 2010; McLean and Kotturi, 1995), or searching within a Gaussian sphere (Collins and Weiss, 1990; Magee and Aggarwal, 1984; Straforini et al., 1993). The second group of methods focuses on learning to infer VPs from large-scale datasets containing VP annotations. Borji (2016) used a convolutional network to infer VPs, while Zhai et al. (2016) extracted global image context with a deep convolutional network to constrain the location of possible VPs. Chang et al. (2018) trained models on one million Google street-view images. [Discuss user specify VPs]

Based on the estimated VPs, the camera’s parameters can be estimated. When assuming the camera is free of skew and the principal point is at the center of the frame, deriving the intrinsic camera parameters becomes straight-forward with the position of vanishing points. By considering the camera positions, the extrinsic parameters could also be calculated. These estimated camera
parameters enable the creation of a transformation between the camera's coordinate system and the world coordinate system.

The common problem with all of these methods is that they were developed for fixed traffic cameras. This study will determine if this vanishing point technique can be equally powerful for mobile cameras.

**METHODOLOGY**

This section describes computer methods used in this study to infer speed and identify vehicles, for use in community support of policing dangerous driving and related behaviors.

The paper proposes a deep learning framework for speed estimation and vehicle identification, with object detection, object tracking, character recognition, and vanishing points techniques. The research assumes that the mobile phone is held steady while recording the video, as inclination angles and phone movements were not considered, and most of the existing research focused on fixed cameras. Although this assumption may not always be realistic, this paper aims to demonstrate the feasibility of using CV to infer a vehicle's speed and license plate from a mobile recorded video. Architecture of the vehicle speed inference and recognition system is depicted as shown in Figure 4. It is described as follows -

After performing object detection and tracking using either MaskRCN or YOLOv7 and DeepSort or Strong Sort algorithms, each cropped image of the tracked vehicle is sent to the license plate detection model (Anpr-Org (2023)), this is the YOLOv7 model finetuned on a license plate dataset taken from college-dbbbrk/anpr-x1a2o project in Roboflow site (Dwyer 2022). Once the bounding box of the license plate is detected, the cropped vehicle image is passed to a Super Resolution Model, which enhances the image; this image is then passed to the EasyOCR model for text recognition. In addition, the cropped image of the vehicle is also sent to the vehicle make and model recognition model, which is trained on Stanford cars and The Vehicle Make and Model Recognition dataset (VMMRdb) (contains cars in US metro areas) datasets. The generated output for each frame, including the bounding boxes, name and classification confidence of both vehicle, license plate and its OCR output and car make along with vehicle ID are stored in a text file for further analysis.

To estimate the vehicle speeds, this paper adopts the speed estimation pipeline proposed by Dubská et al. (2014) to pinpoint the 3D bounding boxes of vehicles. Specifically, the algorithm first obtains the contours of each vehicle by utilizing the MaskRCNN framework. A pretrained ResNet-50 network is used as the feature extractor for the framework to extract the masks of vehicles from video frames. Although the assumption was made that the mobile phone is static while recording the video, the perspective of the camera remains unknown. Therefore, VPs need to be estimated from the video. This paper adopts two different modes to obtain the VPs. The first relies on the users to provide the estimated locations of VPs. The users are instructed to infer the approximate locations of VPs by themselves. Generally, the first VP can be located by finding the intersection of the left-side and the right-side of the road in the video, and the second VP can be approximately
located by extending the edges of the vehicles. The second mode employs an automatic estimation of VPs. It utilizes the VP detection algorithm proposed by Lu et al. (2017). They used two lines to get the first vanishing point $V_1$, then uniformly takes a sample of the second vanishing point $V_2$ on the great circle of $V_1$ on the equivalent sphere. Although this is not a deep-learning based method, it has fast inference speed that allows for potential mobile application. In our experiment, we estimate the VP only once using the first frame.

The estimated VPs enable the estimation of 3D bounding boxes of vehicles. Specifically, the algorithm finds tangent lines of the vehicle blobs that are coinciding with the vanishing points (Dubská et al., 2014). Starting from each VP, the algorithm finds two lines that are tangent to the vehicle blob; the 3D bounding boxes of vehicles are decided by all these lines and the intersections between these lines following Dubská et al. (2014). Meanwhile, these VPs also enable the construction of the perspective transformation. From each of the estimated VPs ($V_1$ and $V_2$), two lines are extended to intersect with the points inside the frame. Four intersection points of these extended lines can construct a rectangle in world coordinates (these lines were selected to avoid including one of the VPs in the rectangle). Assuming the vehicles are moving towards one of the VPs, the perspective transformation $f$ could be constructed to rectify this rectangle so only the vertical (or horizontal) movement of vehicles is preserved.

Having obtained the coordinates of 3D bounding boxes of the vehicles, the vehicle speed was then estimated from the movement of bounding boxes. Denote the two points at both ends of the 3D bounding box as $A = [a_x, a_y]^T$ and $B = [b_x, b_y]^T$, respectively. These two points move to $A' = [a'_x, a'_y]^T$ and $B' = [b'_x, b'_y]^T$ at the next frame. The relative movement of vehicles between frames is then derived by $\lambda = \frac{\|f(A) - f(A')\|}{\|f(A) - f(B)\|}$. To determine the distance that vehicles have moved during a single frame, the actual distance is calculated by multiplying $\lambda$ by the median size of real-world vehicles, in this paper the speed is calculated for the cars so the median size of real-world cars (4.5 meters) is used. The speed is then the distance multiplied by frame per seconds (FPS).
RESULTS

The paper presents an integrated model for both speed inference and vehicle identification. However, due to time constraints, experiments were conducted using different object detection models (Mask R-CNN and YOLOv7) and tracking models (DeepSort and StrongSort). As these models yielded similar levels of accuracy, the paper reports the results of license plate detection and vehicle make identification obtained using YOLOv7 and StrongSort as the baseline. Additionally, Mask-RCNN and DeepSort were used for speed inference. The color identification techniques that were available as open-source did not perform well when tested on sample examples of vehicles. Therefore, only the results for vehicle license plate and car make recognition are presented. In these experiments, only the car make is considered because the model identification test on sample data has too much variation.

The results presented in this study were obtained using the test images of UFPR-ALPR dataset, which constitute 40% of the entire dataset. UFPR-ALPR dataset consists of 4,500 fully annotated
images (including over 30,000 license plate characters) from 150 vehicles captured in real-world scenarios in Brazil where both the camera and the vehicle were moving. The images were captured using three different cameras, namely, GoPro Hero4 Silver, Huawei P9 Lite, and iPhone 7 Plus (Laroca et al., 2019). When using pretrained weights, the license plate detection model was able to detect 78.5% of license plates with a predicted and true box intersection greater than 70%. The OCR results presented in the Table 1 were obtained as part of one of the authors Adv Computer Vision course project, it was observed that applying super resolution with pretrained OCR gave better results than using just OCR. Super-resolution techniques has potentially improved the results, as demonstrated in Figure 5. In the figure, the 'Input image' output from EasyOCR is 'EE', while the prediction for the Real-ESRGAN output image is 'IU B6t5o62', with both predictions having a confidence score of less than 0.5. Nevertheless, as can be seen in the second image in Figure 4 processed by EasyOCR, some characters are identified correctly.

<table>
<thead>
<tr>
<th>Model</th>
<th># Correct OCR</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>OCR</td>
<td>252/1800</td>
<td>14.2%</td>
</tr>
<tr>
<td>Super Resolution + OCR</td>
<td>407/1800</td>
<td>22.6%</td>
</tr>
<tr>
<td>Super Resolution + Fine-tuned OCR</td>
<td>847/1800</td>
<td>47.0%</td>
</tr>
</tbody>
</table>

Table 1: OCR model results

However, even after applying super resolution the EasyOCR pretrained model was only able to identify 22.6% of the license plates correctly. The main reason for the low accuracy of character recognition in license plate recognition is the lack of clarity in the extracted license plate image, coupled with the use of an OCR model that is not specifically trained to recognize license plate information but rather designed to recognize regular text. So the OCR model was fine-tuned on a small subset of UFPR license plates and synthetic data, which improved the accuracy. The model was then able to identify 47.0% of the license plates.

![Image of license plate improvement using Super-Resolution](image.png)

Figure 5. License plate image improvement using Super-Resolution

<table>
<thead>
<tr>
<th>Phone Type</th>
<th>Camera Resolution</th>
<th>Video Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iphone 14 Pro</td>
<td>48 megapixel</td>
<td>1080p@30fps Full HD</td>
</tr>
</tbody>
</table>
Table 2: Cell Phone Resolution (GSMArena.com)  

<table>
<thead>
<tr>
<th></th>
<th>Resolution</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Samsung M33</td>
<td>50 megapixel</td>
<td>2160p@30fps Ultra HD</td>
</tr>
<tr>
<td>Oneplus Nord2</td>
<td>50 megapixel</td>
<td>1080p@30fps Full HD</td>
</tr>
</tbody>
</table>

Note: fps*: frames per second

The license plate and Make and model systems are also tested on the videos of vehicles at different speeds that were collected by the authors with the following 3 mobiles and resolution properties as shown in Table 2. For speeds under 40 mph: Text recognition is correct only when the vehicle license plate is near camera as shown in the Figure 6 and for speeds between 40 to 60 mph: ANPR model did not give any text output because the license plate images are of low quality as shown in the Figure 7. Make & model system could only identify vehicle make correctly, but not model type, and even make is inconsistent as shown in the Figures 6 and 7.

![Image of vehicle with text recognition output](image1)

Figure 6: Processed output for the vehicle moving at a speed of 30-40 mph: The license plate of the vehicle detected with the confidence score of 0.96 and text is predicted to be 'TEXAS' with a confidence score of 0.61, while the number is predicted to be 'RL* 23**' with a confidence score of 0.28. The make and model year of the vehicle is predicted to be a Mini Cooper 2009.

![Image of vehicle with text recognition output](image2)
Figure 7: Processed output for the vehicle moving at a speed of 50-60 mph on IH-35 in Austin, Texas: Although the license plate of the vehicle was detected by the model, no text could be predicted as it was not visible. The make and model predicted for the vehicle is a GMC Terrain SUV, but the actual make and model is a Toyota Sienna.

**Testing on public datasets**

The speed estimation algorithm in this paper was first tested on a public dataset called VS13 (Djukanović et al. 2022). This dataset contains video recordings of one car at different speeds, captured at different cameras angles. The ground-truth speeds of vehicles are provided in this dataset. In this paper, the Mazda is selected as examples to assess the performance of speed estimation algorithm. The speed estimation of vehicles was estimated between the 100th to 160th frames in videos for user-input VP, and 100th to 130th frames for automatically estimated VP. The median of estimated speeds is reported as the estimation of vehicle speed.

<table>
<thead>
<tr>
<th>Estimated Speed (automatically estimated VPs)</th>
<th>Estimated Speed (user-input VPs)</th>
<th>Ground-truth Speed</th>
<th>Absolute Error (km/h)</th>
<th>Relative Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>29.30 km/h</td>
<td>28.04 km/h</td>
<td>30 km/h</td>
<td>0.70 / 1.96</td>
<td>2.33 / 6.53</td>
</tr>
<tr>
<td>39.28 km/h</td>
<td>42.30 km/h</td>
<td>40 km/h</td>
<td>0.72 / 2.30</td>
<td>1.80 / 5.75</td>
</tr>
<tr>
<td>50.11 km/h</td>
<td>56.16 km/h</td>
<td>50 km/h</td>
<td>0.11 / 6.16</td>
<td>0.22 / 12.32</td>
</tr>
<tr>
<td>73.87 km/h</td>
<td>81.47 km/h</td>
<td>60 km/h</td>
<td>13.87 / 21.47</td>
<td>23.12 / 35.78</td>
</tr>
<tr>
<td>22.07 km/h</td>
<td>77.97 km/h</td>
<td>70 km/h</td>
<td>47.93 / 7.97</td>
<td>68.47 / 11.39</td>
</tr>
<tr>
<td>81.47 km/h</td>
<td>85.32 km/h</td>
<td>81 km/h</td>
<td>0.47 / 4.32</td>
<td>0.58 / 5.33</td>
</tr>
<tr>
<td>86.33 km/h</td>
<td>100.30 km/h</td>
<td>90 km/h</td>
<td>3.67 / 10.30</td>
<td>4.08 / 11.44</td>
</tr>
</tbody>
</table>

Table 3: Estimated speed using automatically estimated and user-input VPs with different ground-truth speeds.

Table 3 showcases the estimated speed of vehicles using automatically estimated VPs and user-input VPs. It can be observed that automatically estimated VPs give better prediction on speed except for one case. The absolute errors are smaller than 5 kilometers per hour on five out of seven cases that have been tested. The user-input VPs can also provide accurate estimation of speeds, despite that the errors are slightly larger compared to using the automatically estimated VPs on several cases. However the VPs specified by users would still be valuable as they can serve as the backup solutions for speed inference once the VP estimation algorithm cannot provide reliable results.

**CONCLUSIONS**

Speed estimation is good, with errors of 5% and 5 km/h on 5 of 7 test cases. When super resolution was utilized before the pretrained OCR, license plate text recognition produced better results, but the accuracy remained poor. Fine-tuning the OCR model improved the performance significantly. Fine-tuning the super resolution model on license plates can help improve the performance even more. For the make and model system to perform better, it has to be trained on
more and more current make and model data and tested on real-time annotated images. Errors are lowest when filming lower-speed vehicles in landscape view (horizontal) with a 3-camera phone.

**Limitations and Future Work**

Directions for future research include extending the analysis to nighttime videos since all tests conducted for this system were based on daytime videos. Moreover, assuming that the mobile phone used for recording is stationary and not moving is not always realistic. It would be useful to investigate more complex scenarios in the future. In addition, integrating models capable of identifying a vehicle’s color could prove useful in cases where license plates are obstructed or missing, reducing the likelihood of missed traffic violations. Mobile camera properties such as aperture size and shutter speed can be experimented with to capture video recordings without motion blur. One potential extension could be the development of a mobile application that can send recorded videos to the computer vision system for analysis. However, recording videos while driving poses a safety risk to both the driver and others on the road. To address this concern, an alert system could be incorporated into the application to prevent drivers from recording videos while driving, thereby enhancing safety and reducing the risk of accidents. These suggested enhancements could be explored as part of future work on the system.

**ACKNOWLEDGEMENTS**

We would like to express our sincere appreciation to the following individuals for their valuable contributions to this project. We thank Jahnavi's Computer Vision class project teammates Eva Natinsky and Samridhi Roshan, whose efforts were instrumental in generating the improved results for license plate recognition as part of the Adv Computer Vision class project. Their dedication, hard work, and expertise were invaluable throughout the project.

We are also grateful to Aditi Bhaskar for her diligent and insightful paper edits, which significantly improved the quality of this manuscript. Additionally, we extend our thanks to outside experts who generously provided their time and expertise to review our work and offer valuable suggestions and comments.

**REFERENCES**


Anpr-Org (2023) GitHub - ANPR-ORG/ANPR-using-YOLOV7-EasyOCR. Available at: https://github.com/ANPR-ORG/ANPR-using-YOLOV7-EasyOCR.


KVUE 2022. License to scan: Plate readers found across Texas are mostly unregulated. https://www.kvue.com/article/news/investigations/defenders/texas-license-plate-readers-widely-unregulated/269-d32a0ea0-edb6-4ce4-a25b-61c135c581a6


pytesseract. (2022) PyPI. https://pypi.org/project/pytesseract/


Ralph, K. et al. (2022a) “Can a racial justice frame help overcome opposition to automated traffic enforcement?,” Transportation Research Interdisciplinary Perspectives, 14, p. 100594. https://doi.org/10.1016/j.trip.2022.100594.


UN.ESCAP (2020) Status of law enforcement on four major risk factors in the Asia-Pacific region. Available at: https://hdl.handle.net/20.500.12870/4727.


