Data Preprocessing for Traffic Videos

Zhiying Zhu*, Jie Gong ¶

The City College of New York*, Rutgers University¶

Introduction:

American streets were originally developed with road designs tailored to drivers and automobiles. Yet with the growing popularity of new micro-mobility options including bicycles, e-bikes, e-scooters, skateboard and a wide range of others, the landscape of urban transportation has evolved drastically. This rapid change challenged the current schema of traffic system and placed increasing risks on pedestrians and other road users. Over 6,000 pedestrians died on U.S. streets in 2018, an 14% increase over 2015 and a 27% increase over 2014 [1, 2, 3].

Each city has different population demographics, road design, and traffic complexity, and as of now, no research works have been done to understand the casual factors for collisions in the New Jersey community. It would be helpful to disentangle the casual factors at moments when traffic accidents really happen. However, those events are usually very rare, and the sparsity of data makes it difficult for any computer modeling or deep neural network training. To mitigate this, we introduce the concept of "near misses". "Near misses" collision refers to those situations in which an accident almost going to happen or there is huge risk related to the situations.

The rarity of pedestrian fatalities and those injuries associated with e-scooters makes disentangling the causal factors difficult. Hence, this project focuses on "near misses" between vehicles and pedestrians, e-scooters, e-bikes, and bicycles. We want to disentangle these underlying factors for near-miss collisions between pedestrians and any micro-mobility users in New Jersey. And the main objective of this project to use advanced computer vision technique to create a data preprocessing module, which takes in a raw video and outputs a processed video with the accurate detected pedestrians, bikes, e-scooters, and vehicles labels, that can later be used to detect near-miss collision.

Related Work:

Architectural or behavioral factors may play an important role in unraveling factors contributing to road collision. Many researchers devoted their efforts to understand the impact of road design, road density, bicycle speed, or phone use on pedestrian-bicyclist or pedestrian-electric-scooter collisions [4, 5, 6]. Other researchers hypothesize potential collision or near-miss collision will lead to actual collision and focus on identifying the contributing factors to near-miss collisions between pedestrians and micro-mobility users in California [7].

Many have approached the object detection problem using the You Only Look Once version 3 (YOLO-v3) package in the Open-Source Computer Vision (OpenCV) library [8]. OpenCV is a popular open-source computer vision libraries with many real-time image and video processing

and analytics capabilities [9]. The YOLO package is a real-time object detection algorithm that only takes one glance at the image to predict the class of the object and the location of the object. It treats object detection as a regression problem and use a single neural network from images and compute spatially separated bounding boxes and predicted class probability. The score encodes for both how likely the detected class is and how well the predicted bounding box encloses the object. Some other researchers have used the MobileNet-SSD algorithm in OpenCV for object recognition [10]. MobileNet-SSD is designed with an end-to-end Convolutional Neural Network (CNN) architecture. It takes an image as an input, then pass it through many convolutional layers with various filters, and maps features from different positions of network to predict the bounding boxes and class label.

Design:

I. Data Collection

To understand the real traffic situation, a camera is replaced to record the 24-hour traffic interactions at Asbury Ave, New Jersey. Each video has a size range from 38GB to 58GB and is about 24 hours long. The position of this camera is very informative since it captures complex road conditions and it records videos both day and night. We specifically pick a location in which the traffic situation is difficult. This will give us more observation of "near misses" collisions and traffic violations. There are risk factors not only during busy hours, but also during night when the light conditions are bad or even during free hours when participants are more likely to violate traffic laws. 24-hour monitoring enables us to catch diverse risk factors and avoid data bias. The position also has relatively busy traffic and a diverse community. This also enhances the quality of our dataset. Python 3.8.3 will be used as the main programming language to perform object recognition and feature extraction.

II. Object Detection

A cascade model is used to perform the object detection on the raw videos. The cascade model is composed of an existing YOLO v3 model [11] and a self-trained customized model. It is designed in such a way to retain both the pre-train YOLO labels and the customized labels (i.e., scooter rider and biker). As seen in Figure 1, the cascade model takes a raw video as an input, then it first passes through the YOLO v3 algorithm to detect any non-person labels. If a person label is detected, then the cascade model will expand the image region where the bounding box for the person label is detected and feed into the customized model. If the customized model detects any biker or scooter rider, it will then return the biker and scooter rider label accordingly with the detected bounding box and confidence score. If the customized model did not detect any scooter rider or biker, then the cascade model will then use the person label, bounding box, and confidence score obtained from the previous step from the YOLO v3 model.

The YOLO v3 model contained the pre-train weights and neural network configuration files that was trained on the COCO dataset [12]. The YOLO v3 model is capable of detecting 80 different class labels which include these traffic relevant objects such as traffic lights, car, bicycle, person, skateboard, motorbike, bus, train, and truck. However, the YOLO v3 model lacks the scooter rider,

biker labels and cannot differentiate between biker, scooter rider, and pedestrian within the person label, which are the essential part of this project. Hence, a customized model was trained to specifically detects scooter rider and biker.

The customized model was trained using the Darknet framework. To obtain the training dataset, a script was developed to crop images containing bikers or scooter riders from one raw 24-hour video. To supplement the cropped video images for sufficient training sample, online images searched through the Google engine was also used as part of the training sample. A total of 1,150 training images were labelled using the open source *LabelImg* application [13]. To make sure the customized model can still perform object detection despite various video resolutions, a separate script was written to generate 10 different resolutions of the same image with 10% incremental image resolution, as shown in Figure 2. Altogether, 10,350 images were feed into training the customized model.

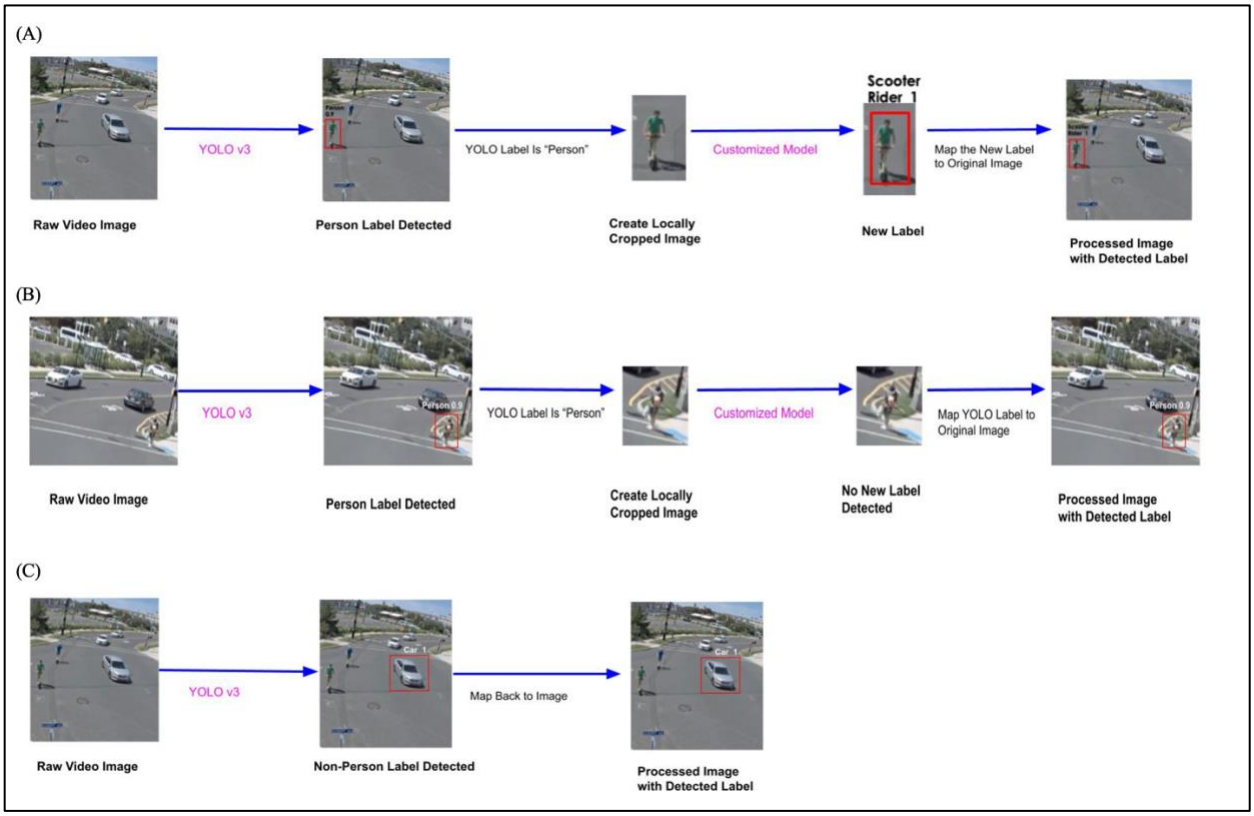

Figure 1. **Cascade Model Architecture**. The cascade model takes a raw video as input, and it first uses YOLO v3 for detection and use the Customized Model. (A) When a YOLO detects "Person" as the label, it crops the local region, feeds into the customized model, and displays the new label (i.e., scooter rider or biker) obtained from the customized model back to the image; (B) When a YOLO detects "Person" as the label and feeds the cropped image into the customized model. If the customized model does not detect the presence of scooter rider or biker, it displays the previous YOLO label back to the image; (C) When a YOLO detects a non-person label, it skips the customized model and directly display the label onto the image.

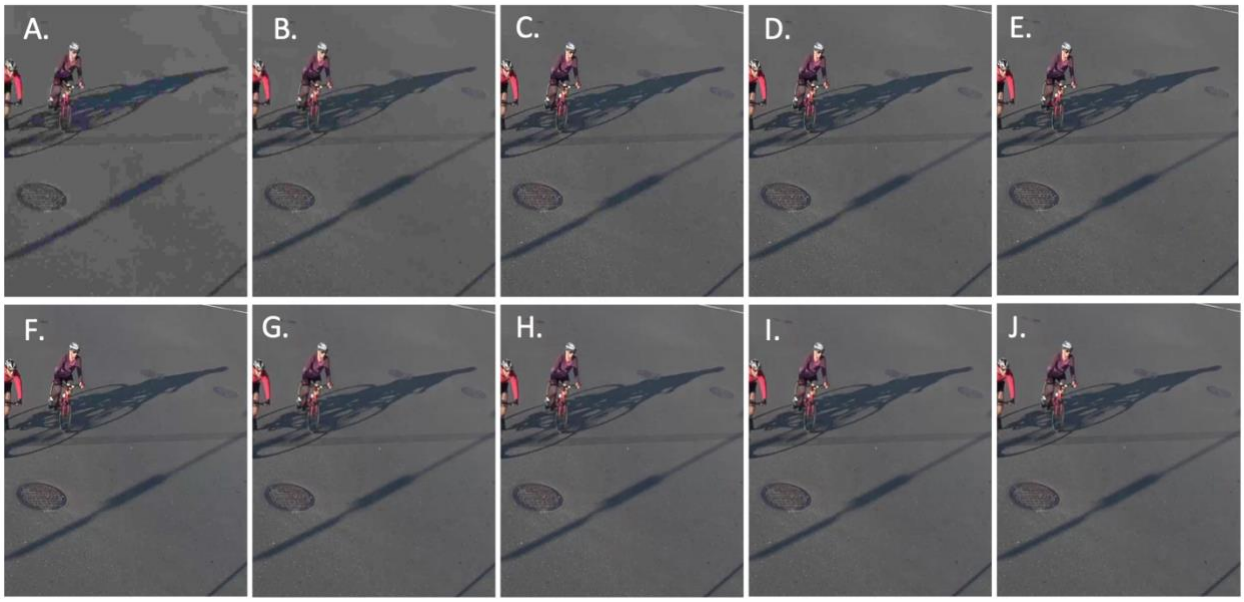

Figure 2. The same image with 10 different resolutions. (A) 10% of the image resolution; (B) 20% of the image resolution;(C) 30% of the image resolution; (D) 40% of the image resolution; (E) 50% of the image resolution; (F) 60% of the image resolution; (G) 70% of the image resolution; (H) 80% of the image resolution; (I) 90% of the image resolution; (J) Original image resolution

III. Evaluation Metrics

The cascade model will be evaluated on 374 biker and scooter rider images, that are cropped from a different raw video that the model had never seen before. These images were first manually labelled with the LabelImg tool. The manual class label and bounding box are considered the ground truth. The cascade model predicted class label and bounding boxes are evaluated against the ground truth. Model performance will be assessed through these following metrics: the number of predicted true label, the number of predicted false label, the number of missed, accuracy, and average intersection over union (IoU). The formulas for accuracy and IoU are shown Equation 1 and Equation 2, respectively.

Equation 1.

$$Accuracy = \frac{total\ of\ predicted\ true\ labels}{total\ of\ predicted\ true\ label + total\ of\ predicted\ false\ label + total\ of\ missed\ label}$$

Equation 2.

$$IoU = \frac{Intersection\ Area}{Union\ Area}$$

Results:

As shown in Figure 3, you can see that the cascade model can successfully detect the desirable labels of biker and scooter riders from our videos. It can compute the correct class label and draw appropriate bounding boxes on detected object on top of the image. From Figure 4, you can see that the cascade model is a decent method for retaining the pretrain YOLO v3 models (i.e., traffic light, car, motorbike) and obtaining the new labels (i.e., biker and scooter rider) from the customized model as well as overlaying all label and bounding boxes on the same image.

The model performance and evaluation of the customized model are showed in Table 1. The algorithm is evaluated against a total of 374 biker and scooter rider images combined. Out of the 374 images, there are with 218 biker images and 256 scooter rider images. The ratio of images to the number of bounding boxes is not 1:1 because there are images with multiple objects detected. 218 biker images have 272 bounding boxes; 156 scooter rider images have 211 bounding boxes; and 374 biker and scooter rider images combined have 483 bounding boxes. As you see in Table 1, the customized model has 94.85% accuracy in detecting biker label, 83.89% accuracy in detecting scooter rider label, and 90.06% accuracy in detecting either biker or scooter rider label. The model has a 15% chance of missing a scooter rider label, 5% chance of missing a biker label, and 9% chance of missing either a biker or a scooter label. The average intersection over union (IoU) is 95% for the biker label, 93.7% for scooter rider labels, and 94.5% for either biker or scooter label.

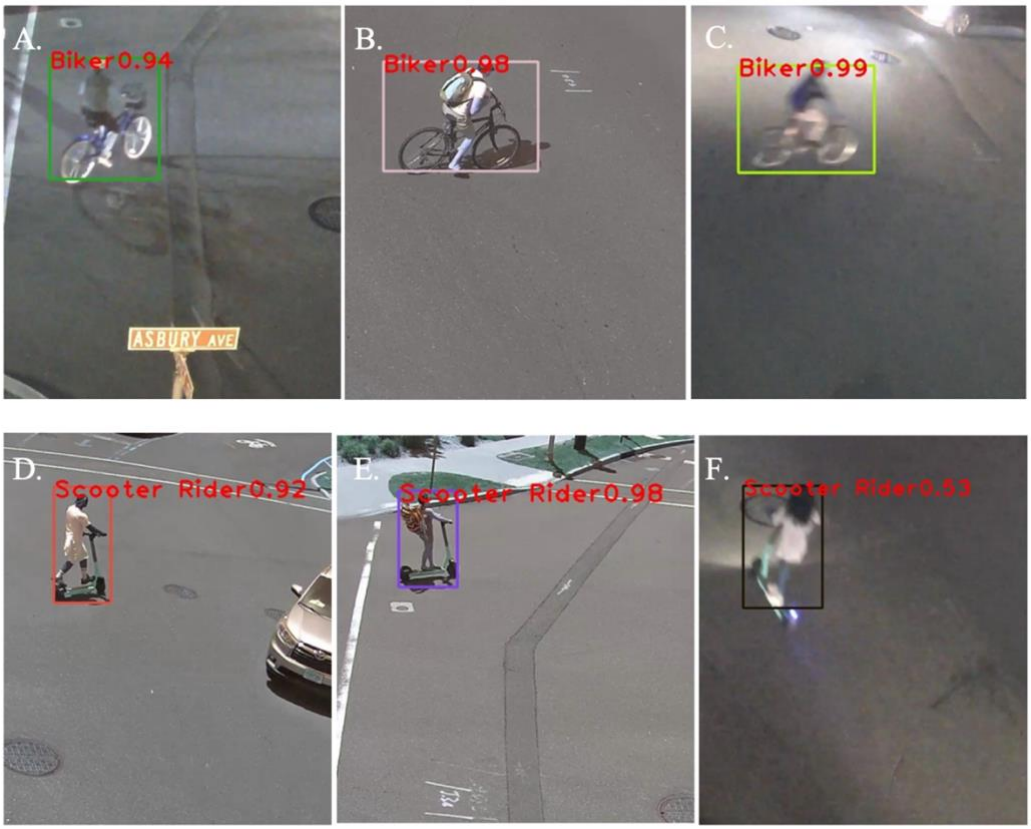

Figure 3. Biker and Scooter Rider Detection. (A – C) are images with biker label detected. (D – F) are images with scooter rider label detected.

Figure 4. Image of the processed video. The cascade model is able to retain the pre-train YOLO v3 model label as well as the customized model label.

	Biker	Scooter	Biker + Scooter
Num of Image	218.00	156.00	374.00
Num of Box	272.00	211.00	483.00
Num of True	258.00	177.00	435.00
Num of False	4.00	10.00	14.00
Num of Miss	10.00	24.00	34.00
Accuracy (%)	94.85	83.89	90.06
Avg IOU (%)	95.00	93.65	94.47

Table 1. Model performance. The testing dataset contains 374 biker and scooter rider images combined with 483 bounding boxes. Out of the combined images, 218 images are biker images with 272 bounding boxes whereas 156 images are scooter rider images with 211 bounding boxes. The number of predicted true label, the number of predicted false label, the number of missed, accuracy, and average intersection over union are calculated for biker testing images alone, or scooter rider images alone, or biker and scooter rider images combined.

Conclusion and Future Work:

The cascade model successfully retains both the pre-train YOLOv3 label and the customized label in the video. As you see in Table 1, the model has better accuracy in detecting biker labels than detecting scooter rider labels. The overall accuracy of 90% in detecting either biker or scooter rider label shows that the performance is decent. The model is also more likely to miss a scooter rider label than a biker label on an image. There are many contributing factors for the less accurate performance in detecting scooter rider label. One potential factor is that the training sample contains slightly more scooter rider images than biker images. Another potential factor is that there are differential unique features between scooter riders and bikers that need to be further investigated.

Future works for this project involve improving the model's accuracy on detecting scooter riders either through further retraining the models on images that it fails to recognize or simply label more scooter rider images as part of the training sample to further enhance the customized model. Given that the moving velocity of various micro mobilities should varies. In another word, the traveling speed for a pedestrian, a biker, a scooter rider should theoretically be different. Therefore, calculating the optic flow for each of the detected object to differentiate biker, scooter rider, and person labels would be another method to validate the current model.

Bibliography

[1] N. H. T. S. Administration, "Traffic safety facts: 2018 data: Pedestrians," Annals of Emergency Medicine, 2018.

[2] N. H. T. S. Administration, "Traffic safety facts: 2015 data: Pedestrians," Annals of Emergency Medicine, 2015.

[3] N. H. T. S. Administration, "Traffic safety facts: 2014 data: Pedestrians," Annals of Emergency Medicine, 2014.

[4] K. D., "Pedestrian and bicycle volume data collection using drone technology," *J Urban Technol,* vol. 27(2), pp. 45-60, 2020.

[5] P. P. Hatfield J, "An investigation of behaviour and attitudes relevant to the user safety of pedestrian/cyclist shared paths," *Transp Res Part F,* pp. 35-47, 2016.

[6] B. A. G. G. Gkekas F, "Perceived safety and experienced incidents between pedestrians and cyclists in a high-volume non-motorized shared space," *Transp Res Interdiscip Perspect,* 2020.

[7] D. a. K. P. Kim, "Analysis of potential collisions between pedestrians and personal transportation devices in a university campus: an application of unmanned aerial vehicles," *Journal of American college health,* 2021.

[8] J. &. F. A. Redmon, "Yolov3: An incremental improvement," *arXiv preprint arXiv,* p. 804.02767.

[9] "OpenCV Library," [Online]. Available: https://opencv.org/. [Accessed 23 09 2021].

[10] L. W. e. al, "SSD: Single Shot MultiBox Detector," *European conference on computer vision,* vol. 1512.02325, pp. pp. 21-37, 2016, October.

[11] P. a. J. M. Viola, "Robust real-time face detection," *International journal of computer vision,* vol. 57(2), pp. 37-154, 2004.

[12] L. Dinalankara, "Face detection & face recognition using open computer vision classifies," *ResearchGate,* 2017.

[13] G. &. H. T. Levi, "Age and gender classification using convolutional neural networks," *Proceedings of the IEEE conference on computer vision and pattern recognition workshops,* pp. 34-42, 2015.