Implementation of Traffic Congestion Classification Method from CCTV Video Based on Image Feature Analysis with YOLO Algorithm

Fernaldy
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
fernaldyl@gmail.com

Rinaldi Munir
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
rinaldi@staff.stei.itb.ac.id

***Abstract*—Traffic congestion is one of the major problems in the field of transportation. Traffic congestion classification can be done to detect congestion so that the occurring traffic congestion can be noticed and handled immediately. Traditional traffic congestion classification methods, such as methods that rely on ground sense coil and GPS, are expensive and require much effort to be implemented. With the help of artificial intelligence, classifying traffic congestion in the video obtained from the traffic surveillance camera is possible to be done. Information of the vehicles in the traffic CCTV video can be obtained by using YOLO algorithm. YOLO algorithm is an object detection algorithm based on convolutional neural network. Traffic features, such as traffic flow, occupancy, density, and speed, can be extracted from the object detection result by utilizing image processing methods. Artificial neural network can then be used to classify the status of the traffic based on these four features. Based on the experiment results, the accuracy, precision, recall, and F1-score of the classification model are 84.75%, 84.66%, 84.75%, and 84.69%, respectively.**

Keywords—traffic congestion classification, CCTV video, YOLO

I. Introduction

Traffic congestion is one of the major problems in the field of transportation. It can result in an increase in air pollution, fuel consumption, and stress level, along with a decrease in people's productivity [1], [2], [3], [4]. These problems can degrade the quality of life of the community. Therefore, traffic congestion is an important issue.

Implementing intelligent transportation system (ITS) is one of the solution to deal with traffic congestion. ITS is a system with the ability to monitor and manage tasks related to the transportation field, such as classifying traffic congestion [5]. When congestion is detected, the authority can then be notified, so that the occurring traffic congestion can be handled immediately. Therefore, the impacts of the traffic congestion can be minimalized.

However, not every ITS model is suitable to be implemented, especially in the developing countries where the traffic congestion frequently happens. The reason is that some ITS model requires high costs and efforts for the infrastructure development and maintenance [6]. Two of the examples are ground sense coil based ITS and GPS based ITS. Ground sense coil based ITS can damage the road surface during its construction and maintenance, while GPS based ITS requires the traffic users to have GPS tracker with them all the time [7], [8].

The number of traffic surveillance cameras installed is increasing [9]. These traffic surveillance cameras can be used as an alternative source of data for ITS in traffic congestion classification because it has quite affordable costs for infrastructure development and maintenance compared to other alternatives [7]. With the advancements in the field of image processing, artificial intelligent, and machine learning, various approaches can be used for classifying traffic congestion by utilizing video obtained from the traffic surveillance camera [10].

Several researches have been done to classify traffic congestion in traffic CCTV video. One of the researches classifies traffic congestion by analyzing the texture of the frame in the video to calculate the density of the traffic using gray level co-occurrence matrix (GLCM) [11]. Another research classifies traffic congestion by analyzing the speed of the traffic in the CCTV video using Lucas-Kanade optical flow [12]. Both of the methods proposed in [11], [12] only observe a single traffic feature. Reference [7] shows that traffic congestion classification method with the best performance is the one that analyzes all four traffic features, including traffic flow, occupancy, density, dan speed. Traffic congestion classification method proposed in [7] analyzes all those four traffic features. However, the vehicle detection process in the proposed method utilizes gaussian mixture model (GMM) which is less efficient and has poor performance when the vehicle in the CCTV video stops for a long time. When the vehicle in the CCTV video stops for a long time, the vehicle will be classified as part of the background and not as a foreground object.

In this paper, a traffic congestion classification method that analyzes all four traffic features, including traffic flow, occupancy, density, and speed, that utilizes YOLO algorithm for vehicle detection is proposed. The YOLO algorithm is an efficient object detection algorithm based on convolutional neural network. Reference [12] shows the advantage of YOLO algorithm over SSD algorithm and faster R-CNN algorithm in terms of efficiency, with the performance being roughly equal. Reference [13] also shows YOLO's advantage over GMM in terms of efficiency and performance.

II. Related Work

Several studies have demonstrated traffic congestion classification in traffic CCTV video by analyzing traffic features. Wei and Hong-ying [11] developed an algorithm for traffic congestion classification based on image texture analysis to estimate traffic density level. The energy and entropy properties were extracted from gray level co-occurrence matrix (GLCM) to represent the vehicle density level in the traffic. The algorithm developed in [11] can classify traffic congestion with an accuracy of 99%. Yang et al. [12] developed a traffic congestion classification method by analyzing speed feature of the traffic in a CCTV video. The position of each vehicle was obtained by utilizing YOLO algorithm to detect vehicle in each video frame. The positions

obtained from vehicle detection process were used as tracking points for speed estimation process with Lucas-Kanade optical flow. This method can classify traffic congestion well in various weather conditions. Ke et al. [7] developed a traffic congestion classification method in traffic video based on four traffic features analysis, including traffic flow, occupancy, density, and speed. The information of the vehicles in the video frame was first extracted with gaussian mixture model (GMM) and convolutional neural network (CNN). The traffic flow feature was determined by the number of the vehicles and the traffic occupancy feature was determined by the ratio of the vehicle pixels to the background pixels. The traffic density feature was extracted from contrast property of the GLCM, while the traffic speed feature was estimated with pyramid Lucas-Kanade optical flow with vehicles' corner points as tracking points. This method can classify traffic congestion with precision, recall, and F1-score of 96%, 94%, and 95%, respectively.

III. Design and Implementation

Fig. 1 shows the architecture design of the proposed traffic congestion classification method. In essence, road and vehicle information in a traffic CCTV video input will first be extracted to calculate the value of traffic flow, occupancy, density, and speed. These features are then used to classify the status of the traffic, indicating whether there is congestion occurring or not. To improve the efficiency of the traffic congestion classification method, the processing is performed on only one frame per second as the information contained in a frame is not significantly different from the information contained in adjacent frames.

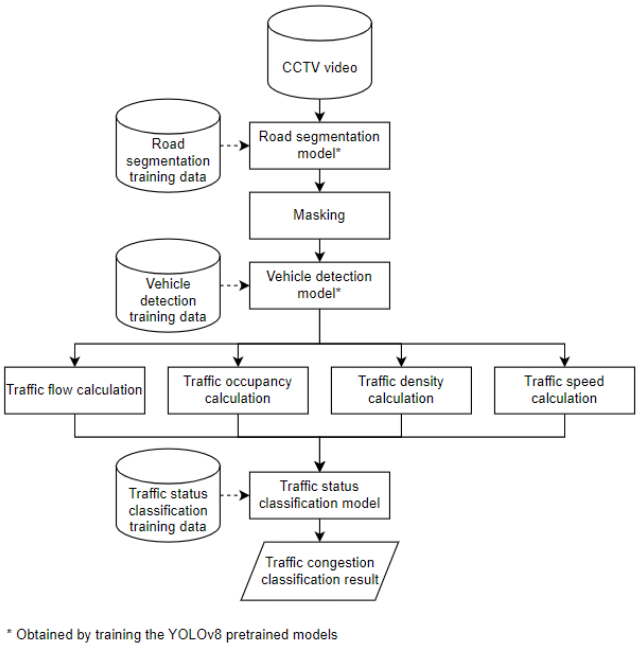

Fig. 1. Proposed traffic congestion classification architecture design

A. Data Preparation

There are three models involved in the proposed traffic congestion classification method: the road segmentation model, the vehicle detection model, and the traffic status classification model. Data needed to train these models were acquired from Pelindung website and Road Vehicle Images Dataset. The Pelindung website provides users with access to traffic CCTV cameras installed in Bandung, Indonesia. Frames were sampled from those accessible CCTV videos for models training and testing. Additional data were acquired from filtered Road Vehicle Images Dataset. Data filtering was performed so that the data used for training and testing only include traffic images taken from CCTV camera because the images in the dataset were obtained from multiple camera sources. This data acquisition process resulted in 669 images for vehicle detection model and 341 images for road segmentation model. These images were then annotated according to each model's needs. On the other hand, data required for traffic status classification model training and testing were acquired by passing traffic CCTV videos through the feature calculation modules. The data were then labeled according to the traffic status in the CCTV videos. Labeling was done into three classes, i.e. free, slow, and congested. This resulted in 1935 rows of labeled data.

B. Road Segmentation

Road segmentation process aims to determine the region of interest (ROI) in a traffic CCTV video. This process results in binary masks which indicate road areas in a traffic video frame. Road segmentation is performed by using YOLO segmentation model. There are five pretrained YOLO segmentation model provided by Ultralytics, i.e. YOLOv8n-seg, YOLOv8s-seg, YOLOv8m-seg, YOLOv8l-seg, and YOLOv8x-seg. Those pretrained models were trained using the acquired and annotated images. Beforehand, the 341 images had been seperated into 261 train data, 40 validation data, and 40 test data. The training was conducted for 100 epochs and each batch's size was 16. Road segmentation is only done once for each traffic CCTV video. The masks produced from road segmentation process on the first frame of the video are used for all remaining frames in the video.

C. Masking

Masking process utilizes masks produced by the road segmentation process. The output of the masking process is an image containing only the road area, while the pixel value of the non-road area is set to 0. This process aims to minimalize the noise resulted from irrelevant area of the image.

D. Vehicle detection

The purpose of vehicle detection process is to produce bounding boxes indicating the position of vehicles in a masked traffic video frame. These bounding boxes are used as inputs for feature calculation processes. This vehicle detection process is performed by using YOLO object detection model. Similar to the road segmentation process, there are five pretrained YOLO object detection models provided by Ultralytics, i.e. YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. Those pretrained models were trained using the acquired and annotated images. Before the training, 669 annotated images had been seperated into 469 train data, 100 validation data, and 100 test data. The training was conducted for 100 epochs and each batch's size was 16.

E. Traffic Flow Calculation

The traffic flow feature is calculated from the number of vehicles in a traffic video frame. This information is obtained from the number of detections yielded from the vehicle detection process using the trained vehicle detection model.

F. Traffic Occupancy Calculation

The traffic occupancy feature is calculated from the ratio of the vehicle pixels to the road area pixels. The number of vehicle pixels can be obtained from the sum of each bounding

box area from the vehicle detection result. Meanwhile, the number of road area pixels can be calculated from the binary masks obtained from the road segmentation result.

G. Traffic Density Calculation

To obtain the traffic density feature, the gray level co-occurrence matrix (GLCM) of the input traffic frame that has been converted into grayscale image should first be computed. The spatial relationship between pixel pairs used for GLCM computation is at 1 pixel distance with angels of 0°, 45°, 90°, and 135°. The value used to represent the traffic density feature is the reciprocal of correlation property from GLCM. The correlation property shows the degree of correlation between a pixel's value and its neighbouring pixels. In a traffic frame with low traffic density, the majority part of the frame is filled with uniform road pixels. Therefore, the correlation between each pixel and its neighbouring pixels is high. Meanwhile, for a traffic frame with high traffic density, the majority part of the frame is filled with diverse vehicle pixels. Therefore, the correlation between each pixel and its neighbouring pixels is low. Because the correlation property is inversely proportional to the traffic density level, the reciprocal of the correlation property is used to represent the traffic density value.

H. Traffic Speed Calculation

Traffic speed calculation is done by utilizing pyramidal Lucas-Kanade optical flow method. The tracking points used for optical flow calculation are the center pixel of each bounding box resulting from the vehicle detection process. The Lucas-Kanade optical flow outputs displacement vector for each of the tracking points which estimates the movement of each detected vehicle from one traffic video frame to another. Each displacement vector can be interpreted as the velocity of vehicle in units of pixels per n/N, where n is the distance between the frames used for optical flow calculation and N is the frame rate of the video. The value of n was chosen to ensure that the displacement vectors can be estimated well, that is, when the movement of the vehicles is not too little and too large. Based on experiment, the displacement vectors can be estimated well when 2 is used as the value of n. The average magnitude of velocity vectors obtained from the optical flow calculation is used as the speed feature value. The pyramidal optimization is used to optimize the Lucas-Kanade optical flow method. The number of pyramid levels used is 3 and the window size for movement estimation is 25×25.

I. Traffic Status Classification

Traffic status classification is carried out by an artificial neural network based on four traffic features, i.e. traffic flow, occupancy, density, and speed. The artificial neural network architecture consists of one input layer, three hidden layers, and one output layer. The input layer contains 4 nodes, while the output layer contains 3 nodes. Meanwhile, the first, second, and third hidden layer contain 64, 32, and 16 nodes, respectively. The activation function associated with each hidden layer is ReLU and the activation function associated with the output layer is softmax. The neural network was trained using the labeled data. Before the training was done, the data had been splitted and preprocessed. The data were splitted based on proportions of 80% train data and 20% test data. Afterwards, the preprocessing was performed. The preprocessing steps involved oversampling the data with non-majority class and feature scaling. The training was then conducted for 100 epochs and each batch's size was 32.

IV. Experiment and Analysis

A. Vehicle Detection Testing

The vehicle detection testing was done on all five trained vehicle detection models: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. The data used for evaluation had been allocated before training. The testing data consist of 100 images, which contain 1,151 vehicle instances. Table I shows the testing result of vehicle detection models. The metrics used for model evaluation were precision, recall, mAP50, and mAP50-95.

TABLE I. Testing result of vehicle detection models

Model	Precision	Recall	mAP50	mAP50-95
YOLOv8n	91.3%	86.2%	94.1%	65.6%
YOLOv8s	92.9%	90.6%	95.9%	67.9%
YOLOv8m	91.6%	92.3%	96.4%	68.6%
YOLOv8l	92.6%	90.9%	96.2%	68.5%
YOLOv8x	90.9%	91.7%	96.1%	68.0%

As shown in Table I, an increase in model size tends to improve the performance on YOLOv8n, YOLOv8s, and YOLOv8m models. Meanwhile, for YOLOv8l and YOLOv8x models, the model performance is not significantly impacted by the increase in model size. The best trained vehicle detection model is YOLOv8m as it has the highest scores in recall, mAP50, and mAP50-95.

Fig. 2 shows the examples of vehicle detection result with trained model.

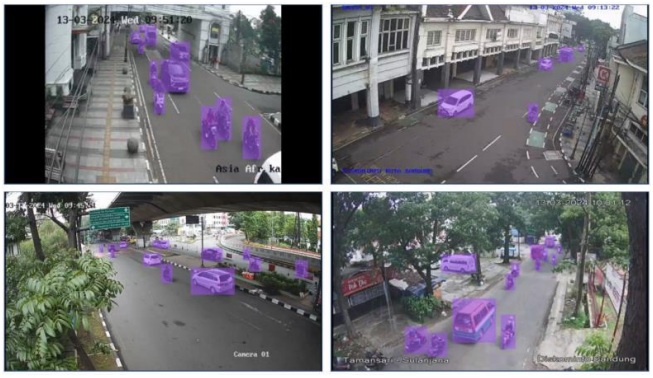

Fig. 2. Examples of vehicle detection result

B. Road Segmentation Testing

Just like the vehicle detection, there are five models trained for road segmentation, including YOLOv8n-seg, YOLOv8s-seg, YOLOv8m-seg, YOLOv8l-seg, and YOLOv8x-seg. Those models were tested with 40 test images which had been allocated before the training process. Those test images contain 67 road instances. The metrics used for models evaluation were precision, recall, mAP50, and mAP50-95. Table II shows the testing result of all five trained road segmentation models.

As shown in the test result, the model performance is not much influenced by the model size. The best trained road segmentation model is YOLOv8s-seg as it has the highest

scores in precision, recall, and mAP50. The mAP50-95 score of YOLOv8s-seg is lower than the mAP50-95 score of YOLOv8n-seg.

TABLE II. TESTING RESULT OF ROAD SEGMENTATION MODELS

Model	Precision	Recall	mAP50	mAP50-95
YOLOv8n-seg	88.9%	95.4%	94.2%	73.2%
YOLOv8s-seg	97.1%	98.3%	97.9%	72.5%
YOLOv8m-seg	92.0%	94.0%	94.8%	62.8%
YOLOv8l-seg	93.6%	92.5%	92.0%	65.4%
YOLOv8x-seg	95.1%	91.0%	93.6%	68.6%

Fig. 3 shows the examples of road segmentation result.

Fig. 3. Examples of road segmentation result

The segmentation process results in binary masks which are used in the masking process to isolate the region of interest, that is, the road area of the image. Fig. 4 shows the examples of mask obtained from road segmentation process and the masking result using the masks obtained.

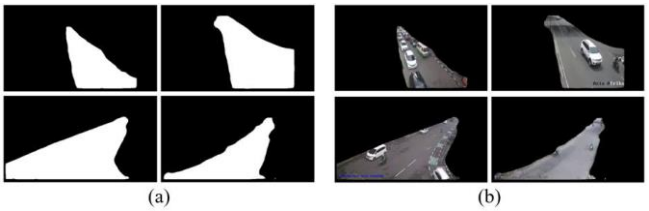

Fig. 4. Examples of (a) binary mask obtained from segmentation process (b) masking result utilizing the masks obtained

C. Traffic Features Analysis

Traffic features analysis was done based on labeled data used for training and testing of traffic status classification model. For each class, the mean value of each traffic feature was calculated. Table III shows the result of the average value calculated.

TABLE III. AVERAGE OF TRAFFIC FEATURE VALUE FOR EACH CLASS

Class	Traffic flow	Traffic occupancy	Traffic density	Traffic speed
free	6.150	0.181	1.012	9.027
slow	13.412	0.454	1.014	5.490
congested	18.228	0.673	1.038	1.937

Based on Table III, it can be observed that the congestion level is directly proportional to the traffic flow, occupancy, and density, while inversely proportional to the traffic speed.

D. Traffic Status Classification Testing

Trained traffic status classification model was tested with 387 test data allocated before training. The data consist of 201 data labeled as free, 117 data labeled as slow, and 69 data labeled as congested. The evaluation metrics used were accuracy, precision, recall, and F1-score. The testing result shows that the model is able to classify the traffic status with an accuracy of 84.75%, weighted precision of 84.66%, weighted recall of 84.75%, and weighted F1-score of 84.69%. Table IV shows the overall accuracy of the model along with the precision, recall, and the F1-score values for each class.

TABLE IV. TESTING RESULT OF TRAFFIC STATUS CLASSIFICATION MODEL

Class	Accuracy	Precision	Recall	F1-score
free	84.75%	89.81%	92.04%	90.91%
slow		75.86%	75.21%	75.54%
congested		84.62%	79.71%	82.09%

Based on the testing result shown, the highest precision, recall, and F1-score belong to the free class. In other words, the false negative and false positive values are relatively low compared to the true positive for the free class. The testing result also shows that the precision, recall, and F1-score of the slow class are the lowest. This implies that the false negative and false positive values are relatively high compared to the true positive for slow class.

E. Traffic Congestion Classification Time Evaluation

The traffic congestion classification time evaluation aims to determine the time needed to process a traffic video frame. A frame processing includes masking, vehicle detection, traffic flow calculation, traffic occupancy calculation, traffic density calculation, traffic speed calculation, and traffic status classification. The road segmentation process is only carried out once, namely for the first frame of the video. The binary masks resulting from the road segmentation process of the first frame were used for the masking process of the remaining frames. The evaluation was done with the acquired traffic CCTV videos and with two execution environments, that is, Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz and NVIDIA GeForce GTX 1650 Ti. Table V shows the testing result of traffic congestion classification time depending on which vehicle detection model used as each model has different inference time which will affect the processing time.

TABLE V. TESTING RESULT OF TRAFFIC CONGESTION CLASSIFICATION TIME

Vehicle Detection Model	CPU processing time (ms)	GPU processing time (ms)	CPU processing speed (fps)	GPU processing speed (fps)
YOLOv8n	198.06	128.49	5.05	7.78
YOLOv8s	265.60	147.96	3.77	6.76
YOLOv8m	403.48	162.62	2.48	6.15
YOLOv8l	629.81	177.07	1.59	5.65
YOLOv8x	859.83	181.91	1.16	5.50

Based on the testing result shown, the larger the model used, the slower the processing speed. The best processing speed was obtained when the program was executed on GPU with the YOLOv8n model, namely 7.78 fps. Meanwhile, the worst speed was obtained when the program was executed on CPU with the YOLOv8x model, namely 1.16 fps. Because the processing is done only on a frame at every second, all models meet the requirement for traffic congestion classification process, whether using the CPU or the GPU.

V. Conclusion

Traffic congestion classification from CCTV videos can be done by analyzing traffic features using YOLO algorithm to provide information about region of interest and vehicles in a traffic video frame. There are four traffic features, that is, traffic flow, traffic occupancy, traffic density, and traffic speed. The traffic flow can be obtained from the number of vehicles detected in a frame and the traffic occupancy can be obtained by calculating the ratio of the vehicle pixels to the road area pixels. Meanwhile, the traffic density is determined by the reciprocal of the correlation property from GLCM and the traffic speed can be obtained by utilizing pyramidal Lucas-Kanade optical flow method. The result of the traffic features calculation can then be used to classify the traffic status using an artificial neural network. Based on experiment, the trained neural network is able to classify the traffic status with an accuracy of 84.75%, precision of 84.66%, recall of 84.75%, and F1-score of 84.69%.

References

[1] M. Barth and K. Boriboonsomsin, "Real-world carbon dioxide impacts of traffic congestion," *Transportation Research Record*, vol. 2058, no. 1, pp. 163–171, Jan. 2008, doi: 10.3141/2058-20.

[2] I. D. Greenwood, R. C. M. Dunn, and R. R. Raine, "Estimating the effects of traffic congestion on fuel consumption and vehicle emissions based on acceleration noise," *Journal of Transportation Engineering*, vol. 133, no. 2, pp. 96-104, Feb. 2007, doi: 10.1061/(ASCE)0733-947X(2007)133:2(96).

[3] S. A. C. S. Jayasooriya and Y. M. M. S. Bandara, "Measuring the economic costs of traffic congestion," *2017 Moratuwa Engineering Research Conference (MERCon)*, May 2017, doi: 10.1109/mercon.2017.7980471.

[4] D. A. Hennessy and D. L. Wiesenthal, "Traffic congestion, driver stress, and driver aggression," *Aggressive Behavior*, vol. 25, no. 6, pp. 409-423, Jan. 1999, doi: 10.1002/(SICI)1098-2337(1999)25:6<409::AID-AB2>3.0.CO;2-0.

[5] K. N. Qureshi and A. H. Abdullah, "A survey on intelligent transportation systems," *Middle East Journal of Scientific Research*, vol. 15, no. 5, pp. 629–642, Aug. 2013, doi: 10.5829/idosi.mejsr.2013.15.5.11215.

[6] J. Kurniawan, S. G. S. Syahra, C. K. Dewa, and N. Afiahayati, "Traffic congestion detection: Learning from CCTV monitoring images using convolutional neural network," *Procedia Computer Science*, vol. 144, pp. 291–297, Jan. 2018, doi: 10.1016/j.procs.2018.10.530.

[7] X. Ke, L. Shi, W. Guo, and D. Chen, "Multi-dimensional traffic congestion detection based on fusion of visual features and convolutional neural network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2157–2170, Jun. 2019, doi: 10.1109/tits.2018.2864612.

[8] H. Cui, G. Yuan, N. Liu, M. Xu, and H. Song, "Convolutional neural network for recognizing highway traffic congestion," *Journal of Intelligent Transportation Systems*, vol. 24, no. 3, pp. 279–289, Apr. 2020, doi: 10.1080/15472450.2020.1742121.

[9] D. Ding, J. Tong, and L. Kong, "A deep learning approach for quality enhancement of surveillance video," *Journal of Intelligent Transportation Systems*, vol. 24, no. 3, pp. 304–314, Oct. 2019, doi: 10.1080/15472450.2019.1670659.

[10] P. Chakraborty, Y. O. Adu-Gyamfi, S. Poddar, V. Ahsani, A. Sharma, and S. Sarkar, "Traffic congestion detection from camera images using deep convolution neural networks," *Transportation Research Record*, vol. 2672, no. 45, pp. 222–231, Jun. 2018, doi: 10.1177/0361198118777631.

[11] L. Wei and D. Hong-Ying, "Real-time road congestion detection based on image texture analysis," *Procedia Engineering*, vol. 137, pp. 196–201, Jan. 2016, doi: 10.1016/j.proeng.2016.01.250.

[12] X. Yang, F. Wang, Z. Bai, F. Xun, Y. Zhang, and X. Zhao, "Deep learning-based congestion detection at urban intersections," *Sensors*, vol. 21, no. 6, p. 2052, Mar. 2021, doi: 10.3390/s21062052.

[13] Q. Peng et al., "Pedestrian detection for transformer substation based on gaussian mixture model and YOLO," *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, Aug. 2016, doi: 10.1109/ihmsc.2016.130.