Blind Spot Warning System based on Vehicle Analysis in Stream Images by a Real-Time Self-Supervised Deep Learning Model

Arash Pourhasan Nezhad, Mehdi Ghatee, Hedieh Sajedi

Abstract—With the advent of intelligent systems, we are still facing a high number of fatal traffic accidents. Driver assistance systems can significantly reduce this rate. For example, when a driver uses a turn signal, driver assistance systems alert the object's presence in blind spot areas. Camera-based driver assistance systems for blind spots usually alert by detecting objects, including vehicles, in image frames. Based on a more dynamic dangerous situation classification for lane changing and turning to the sides, we propose an efficient blind-spot warning system that works with a single camera sensor for each side. Our contribution consists of two sections. First, we take a deeper look at classifying dangerous and safe situations in a dynamic environment with moving objects. Second, to distinguish dangerous situations from safe conditions, we install a pre-trained SOTA object detector to track vehicles in consecutive frames and then estimate the distances of tracked cars by a 6% mean percentage error rate. In addition, to detect objects in blind spots, the proposed system uses cars' relative velocity to warn dangerous situations. This classification process is not real-time. So, in the second section, we propose a tiny model as a driver assistance system for the blind spot that works in real-time. This tiny model feeds optical flow into CNN layers. This vision-based system uses self-supervised learning without the necessity of the labeled data. It shows 97% accuracy and can detect dangerous situations as a real-time system.

Index Terms—Driver Assistant System Blind Spot Warning System Image Processing Deep Learning Self-Supervised Learning.

I. INTRODUCTION

According to the Statistics Center of Iran, hundreds of thousands of accidents occur in Iran every year, which unfortunately causes thousands of deaths. Twelve percent of these accidents occur due to wrong turning and lane changes. With the advancement of technology, this rate can be reduced by advanced driver assistance systems (ADAS) that play an essential role in reducing human errors. There are several types of advanced driver assistance systems, each designed to improve driving and safety [1]. Also, different technologic approaches can be used based on image processing and machine learning to improve safety in transportation systems [2].

By default, mirrors are installed on both sides of all vehicles to widen the driver's viewing angle. However, one of the limitations of these side mirrors installed on the car is creating

Fig. 1: Possible blind spots in heavy vehicles and cars (Blind spots in heavy vehicles are more extended than in cars.) Blind spots are usually a subset of potential blind spots shown in the image. They depend on several factors, such as the type and angle of the side mirrors.

areas that are not visible, and usually, drivers are not aware of these areas. These areas are called blind spots, and vehicles' presence in these areas increases accidents. Blind spots are different for different types of vehicles; see Fig. 1 for example.

When turning sideways or changing lanes, limited viewing angles and blind spots are among the most common causes of car accidents. Various driver assistance systems have been developed for blind spots to prevent this type of accident. Sensors used in driver assistance systems can be divided into visual and non-visual types. For non-visual sensors, we can refer to radar-based systems that can measure distance with high accuracy. However, they are relatively expensive and have some limitations [3], [4]. In vision-based systems, a variety of cameras are commonly used as sensors. In this paper, similar to [4], [5], [6], a camera is used that is not expensive and provides a good field of view. Some systems have also used the fusion of several types of sensors [7], [8], [9], [10]. Other visual sensors, such as stereo cameras [11], [12], have also been used in some systems. In camera-based systems, the cameras are mounted on side mirrors to capture the images of blind spot areas.

Some camera-based driver assistant systems can estimate the distance of vehicles [13], [14], [15], [16], [17]. To accurately estimate vehicles' distance, using fusion with other sensors such as Lidar is more appropriate. The camera functions like the human's eyes. The distant cars in the image have similar dimensions in terms of the number of pixels. Therefore,

A. Pourhasan Nezhad and M. Ghatee are with the Department of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran, e-mails: arashphn@aut.ac.ir, ghatee@aut.ac.ir.

H.Sajedi is with School of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, Iran, email: hhsajedi@ut.ac.ir.

Manuscript received March 4, 2021; revised ?, ?.

as the distance of vehicles increases, the distance estimation error also increases.

The primary step for a robust camera-based blind spot driver assistant system is detecting vehicles in input images. Object detection is one of the main challenges in the digital image processing field [18]. With the help of a massive amount of annotated data, new deep artificial neural network models with variant architectures such as [19], [20], [21], [22], have ameliorated it. The object detection problem takes an image as input and returns the bounding boxes of all objects in the image and their object type. These deep neural networks have lots of layers, so they have a high computational cost.

The gap between the two groups of supervised and unsupervised learning is not very simple to categorize. For example, teacher-student learning [23], [24], [25], [26] and self-supervised learning [27], [28], [29] are new learning techniques that are used in a broad range of applications. The term self-supervised has not been used in the same way in different fields [30], [31], [32], [33]. Their common idea is the usage of unannotated data as a supervisory signal. This paper also utilizes the self-supervised learning approach to use a huge amount of raw data instead of limited human-labeled data.

II. RELATED WORKS

Almost all camera-based blind spot detection systems use an object detection algorithm and determine dangerous situations based on detected objects in the image. Traditional object detection algorithm in blind spot detection systems usually involves two main steps, feature extraction, and detection step. For feature extraction, various methods such as histograms of oriented gradients (HOG) and Gabor filters are represented [34], [35], [36], [37], [38], [5], [39], [40]. In detection step also a variety of classifier methods such as Support vector machine (SVM) [35] and AdaBoost [37] have been used. These traditional methods for detecting and localizing vehicles on the road suffer from a lack of robustness for complicated road scenes and environmental conditions.

New object detection algorithms based on deep learning improve robustness. With the recent advancement in computing capabilities and large data sets, deep learning methods have been used widely. As mentioned and experimented in [38] deep learning-based object detector models such as YOLOv2 [41] achieve better accuracy than traditional models. However, They are still not appropriate to use under limited computing resources. [4] designs an efficient deep learning-based object detector model that can be used as a real-time solution. One of the limitations of blind-spot detection systems is the lack of recognition of all types of vehicles. [37] just detects two-wheeler vehicles or [5] is reliable in detecting only sedan vehicles.

Table I shows the advantages of different works and a summary of their proposed method. [38] is the only system that refers to the relative moving direction of vehicles and does not consider the cars with backward relative moving direction as a potential cause of an accident. As a disadvantage, the camera-based blind-spot warning systems usually do not support all

types of vehicles. Also, they need labeled data. A comprehensive camera-based blind-spot system summarization is as the following:

- Considering the relative moving direction of vehicles as a factor of detecting the dangerous situation
- Cover a variety of vehicles such as cars, trucks, motorcycles, etc.
- Training and evaluation with large amounts of real-world data to cover a variety of possible scenarios

In what follows, we present a new system including **Dangerous Situation Detection** and **Blind Spot Warning** systems. To cover the first two items of mentioned characteristics, object detection, distance estimator, and object tracker subsystems are presented in Sections IV. To provide a distance estimator subsystem, firstly, in Section III, we create a new dataset including the front side of car images and their distance from the camera as the label. Then in Sections IV and V, we presents the details of these systems. The first one is used to detect vehicles in an image frame and track them in streams of images [44]. Then, using the object detector model's output, a neural network is designed to estimate the distance of cars in the image. In this component, we present a new method to determine dangerous situations to turn or change lanes. In the second system, we present our proposed system for a blind-spot warning system. The proposed system is needless to handmade labeled data. Using the analysis performed in the previous section and a large amount of raw data, we train a tiny neural network with reliable performance. Section VI presents the results of the proposed system. The final section ends the paper with a brief conclusion.

III. DATA SET

There are two common data sets for object detection; COCO [45] and VOC [46]. In this paper, we used the COCO data set to detect vehicles in images and extracted six classes of car, person, bicycle, motorcycle, bus, and truck useful in a driver assistance system. For driver assistance systems and smart car researches, we use KITTI [47] data set because, in addition to label object bounding boxes, it provides other information such as object spacing and image depth. However, it is not suitable for object detection tasks. The COCO data set has been used to build a robust object detector. It includes a significant number of data images and classes.

KITTI is a suitable data set for estimating the distance of objects. In addition to labeling them in 2- and 3-dimensional coordinates, it provides more information from other sensors such as Laser-scanner, stereo camera, and bird's eye view images to estimate depth and distance in the image. Most papers such as [48] are used to estimate the distance based on its images or used Lidar data [49].

It includes images with a resolution of 1392x512, taken from the camera mounted on the car. The size of objects in the image depends on the physical properties of the camera lens. Distance estimator models that train with KITTI, usually rely on the camera lens's physical properties and image quality. So its images are not similar to images taken by other cameras.

This paper collects a data set to train and evaluate car distance estimation from the camera. We try a very inexpensive

TABLE I: Camera Based blind-spot warning systems

Model	Advantages	Proposed method
[4]	<ul style="list-style-type: none"> Implemented as an embedded system application Robust model based on CNN 	They use a lightweight neural network structure to solve a specific classification task(Presence or absence cars in blind spot region) as a blind-spot warning system.
[37]	<ul style="list-style-type: none"> Support two-wheeler vehicles 	In this paper, the MCT features vector with an AdaBoost classifier is used. They also use a cascade classifier for processing more efficient
[36]	<ul style="list-style-type: none"> Using a tracker and reducing false alarm rate by controlling tracked reliability points 	They use a cascade classifier with HoG features periodically in the detection stage. Also, a tracker based on the Kalman filter is proposed to track detected cars and reduce the false alarm rate.
[42]	<ul style="list-style-type: none"> Support car and motorcycle detection Implemented as an embedded system application 	This paper proposes a system based on detecting any moving objects. Their detection algorithm uses optical-flow, and then the proposed algorithm can segment the detected moving objects from the background. Detected moving objects in the road (if not exist in the background) are assumed as vehicles.
[43]	<ul style="list-style-type: none"> Implemented as an embedded system application 	The authors, in comprehensive research, propose a fully connected network. Extracted HOG features feed into the network as input. The post-processing stage proposes a method based on thresholding and heat map to reduce the false-positive rate.
[38]	<ul style="list-style-type: none"> Not classify the situation as dangerous when detected vehicle relative moving direction is backward. Implemented as an embedded system application Use one single camera for both sides 	In this paper, a support vector machine (SVM) with HOG features used as a vehicle detector. The proposed vehicle detector is applied in two pre-defined fixed and small windows for each side. These small-sized windows help to reduce computational complexity. Also, estimation of moving direction of vehicles is dependent on the dangerous situation classification
[5]	–	The authors propose a matching mechanism with a scoring function to detect sedan-shaped vehicles. They also show that integrating two types of edge-based and appearance-based features leads to accuracy improvement of their proposed method.

way to provide a data set commensurate with our camera. The collected data set contains about 1000 car images that had a distance of 1 to 12 meters. Unlike the KITTI data set, our collected data set includes cars' images from the front view suitable for the blind spot detection system. We have tried to have images of diverse vehicles in this dataset. Also, the mean distance is 5, and the variance is 3 meters approximately. The size of objects in the image depends on the camera lens's physical properties, so estimating an object's distance is not the same for images taken from any camera.

The biggest problem in collecting data for distance estimation is using distances as labels. The best way to measure distances is to use other sensors, such as laser-scanner, which are very expensive and hard to set up. We test a new method to measure distance for the label of images by using artificial features like Apriltag2. Estimated distances by Apriltag2 are almost accurate in near ranges [50]. We use 15 different Apriltags with different physical sizes. Tags were attached to the front of different cars. We use an object detection model to distinguish the car's tag in the image. If the tag is inside the bounding box, it is attached to the seen car certainly. Then we can crop its bounding box. Finally, by checking the

cropped bounding boxes, we can easily ensure the collected data's suitability, see Fig.2.

IV. DANGEROUS SITUATION DETECTION

In this section, and just for state analysis purposes, we used the state-of-art level object detection YOLOv3-spp (improved version of YOLOv3 [20]) model, which also had a very high computational cost. The object detector model takes an image frame as input and returns all bounding boxes of objects and their types in an input image frame. Blind-spot systems are often used in dynamic environments, and objects in an input image are not fixed. To improve the analysis of vehicle context and detect the dangerous situation when turning to the sides, we try to estimate the relative distance and velocity of cars in an input image frame.

As mentioned earlier, most car distance estimation systems are based on unsupervised methods. Therefore, they are naive and hard to evaluate. The main issue in estimating the distance of cars from the camera based on the supervised techniques is the lack of suitable data sets and data gathering difficulty.

To estimate cars' distance, we detect the vehicles in an image with the object detector model. In this regard, we use

our object detector model. After executing the object detector model on the input image, we have all the bounding boxes containing them.

We use two different types of input to estimate the distance of cars from the camera. Like unsupervised car distance estimator methods, the first input type is a vector that includes the height, width, and position of the detected bounding box in the image. The second type of input is a 256x256 sized image. We crop and resize a shot in the detected bounding box to provide the second type of input for the distance estimator model.

The proposed distance estimator model is based on a feed-forward neural network. It takes two types of inputs and returns the predicted distance by the last neuron. We use a VGG like network [51] to extract image type input features. In the following, VGG like output is used next to the vector type input Fig. 3. Then, after several dense layers, the distance is estimated using a single neuron as the output layer. In the training stage, we also use horizontal flip augmentation and random painting augmentation methods[52]. In random painting augmentation and to cover the attached tags in the image, we paint a rectangle with the cars' tag coordinates. The length and width of the rectangle are obtained randomly based on the half-normal distribution. 2

Fig. 2: A data item of the collected data set. Subfigure (a) is an example of the captured image. The detected object is also shown, and the tag is inside the detected bounding box. (b, c, d) are the cropped bounding box of (a). Also, the tags are shown. We use some random erasing augmentation for the painting tag. As indicated in (b, c, d) the painted areas are different. With a probability of 0.5, we use horizontal flip augmentation as it used in (d).

The car distance is directly proportional to the amount of estimation error. Therefore, the mean absolute percentage error as a proper validation metric is 6% in our experimental images.

Fig. 3: The proposed model for distance estimation. (a) A VGG like CNN structure is installed to extract image features. Extracted image features with vector type features are fed to some dense layers(b) to predict the car distance.

Detecting objects and estimating vehicles' distances from the camera, use just from the appearance information in an individual frame. To use temporal information across some consecutive frames, an object tracker is a must. We use [53] to track cars from previous frames. It makes a sequence of estimated distances from each detected car. A Kalman filter is used to reduce the estimation error. Thus, we can calculate the velocity of vehicles; although this estimation is not very accurate, it is sufficient to understand the backwardness or forwardness movement of detected cars. Without using this object tracker, our proposed system can not work well in scenarios with multiple vehicles in an image.

Estimated car distances are defined based on the camera coordinate system. So we need to transform estimated distances into a proper coordinate system in the next step. We assume that all vehicles are located on one unique plane to use a 2D coordinate system for simplicity. Let X and Y be the axes of the mentioned proper coordinate system as indicated in Fig. 4.

Fig. 4: (a) It indicates the mounted camera coordinate system in the mentioned 2D coordinate system. (b) The two intended red and yellow areas based on the system parameters W_1, H_1, W_2 and H_2 . These parameters depend on the type and size of vehicles, and for larger vehicles, these parameters should be higher.

Earlier, we describe the object detector and car distance estimator models and use them to analyze the context around vehicles. We predict the accident in lane changing or turning because of a dangerous movement across consecutive frames. Two safe and dangerous classes are usually considered to classify an accident's occurrence while turning to the sides. Based on the analysis of the vehicle's environment and the type and size of the vehicle, we will provide a set of rules to classify the risk of an accident while turning to the sides. Vehicles from mini-cars or crossovers to trucks come in a variety of sizes. Depending on each vehicle's size, we consider two types of regions as indicated in figure 4.

Similar to [4], the existence of any object in the red region classifies as a dangerous situation. We also consider a yellow area. The presence of any car in this region is dangerous only if its distance reduces. Based on these regions and analyzing the vehicle's context, our rule-based inference engine classifies the situation into safe or dangerous.

V. BLIND SPOT WARNING

In the previous section, we presented a comprehensive and customizable method for detecting safe or dangerous situations. Unlike a blind-spot warning system, it needs lots of computation costs. Also, the real-time execution of this method is far from feasibility. In this section, we propose a blind-spot warning system based on self-supervised learning. Also, our proposed system is defined based on a tiny neural network. It is computationally efficient and executes real-time even on a CPU.

We resize all frames into 128x128 to reduce computation. Then, we calculate optical flow. The optical flow of frames helps us to extract temporal motion features across frames. We concatenate a grayscale frame and its optical flow to build an input tensor to fuse the frame's visual content and its temporal motion features. We propose a tiny neural network for a blind-spot warning system. The reference [12] takes the input tensor from a grayscale image and two-dimensional optical flow. The fusion of image frame and optical flow increases the model's robustness to prediction based on image temporal information. Optical flow can separate moving objects from the background image.

Providing a significant amount of labeled data to train the tiny model that use optical flow as the input is also costly. We train our proposed model by the self-supervised learning method. This learning method trains the tiny model with a massive amount of raw video data and labels generated with our rule-based inference engine Fig. 5.

The purpose of using this self-supervised learning method is to make the tiny model prediction similar to the rule-based inference engine.

VI. EXPERIMENTAL RESULTS

This part aims to evaluate the quality of SOTA object detection models on our collected data. As the first step of evaluation, we check the object detection model on the collected data. We have prepared the label only on 1000 sampled images. The label only includes the objects we need

TABLE II: Architecture of tiny model for blind spot warning system

	Layers	kernel size
1	Conv	(3, 3)
2	Conv	(3, 3)
3	Max Pooling	(2, 2)
4	Conv	(3, 3)
5	Conv	(3, 3)
6	Max Pooling	(2, 2)
7	Conv	(3, 3)
8	Conv	(3, 3)
9	Max Pooling	(2, 2)
10	Flatten	
11	32 x Dense	
12	16 x Dense	
13	16 x Dense	
14	1 x Dense

Fig. 5: The process of self-supervised learning method to train the tiny model for the blind-spot warning system. It also uses horizontal flip augmentation in the training stage.

on the road. In all cases, in addition to correctly identifying all the objects, other objects are also detected on the sidewalk.

The detection of these objects, which were not labeled due to lack of importance, indicates SOTA object detection models' robustness. [38] also gets similar results, and on their collected data set YOLOv2 achieved 100% accuracy. We show a sequence of frames with the indicated output of object detection and object tracker models in Fig. 6.

As indicated in the data set section, we collect a data set to train our proposed distance estimator model. About 1,000 labeled images were collected. We split them into 400 and 600 for test and train sets. For train set data, we use mentioned data augmentation methods like flipping to expand our data. The ADAM[54] optimizer has been used for the training process. We reduce the learning rate to decrease the loss.

The distance estimator model returns the car's distances in the meter. Mean absolute percentage error is chosen as the primary metric. At the end of the training, we achieved a 6% error for our testing set, as shown in Fig.7. We also have some experiments to apply other CNN architecture such as EfficientNet [55] and ResNet [56] to extract images' features. But we have not found any valuable improvement to present.

Fig. 6: In this figure, a sequence of frames from our collected data is shown. All detected bounding boxes of objects are drawn with yellow color. We indicate car, motorcycle, and people classes with the letters c, m, and p, respectively. Also, each class's detection confidence is written on the left top corner of the bounding box. The white color bounding boxes show the output of the object tracker. In the first frame, we use it to tracker our desired object. In the successive frames, we can follow this object with the unique id = 0.

TABLE III: Comparison of blind-spot warning systems (%)

Model	Cars	Two-wheeler	reported results
Sep-Res-SE [4]	*	-	Accuracy: 97.58 %
MobileNetV2 [4]	*	-	Accuracy: 95.35 %
VGG Block [4]	*	-	Accuracy: 98.01 %
Two-tier detector [37]	-	*	Precision: 95 - Recall: 93
Two-tier detector & wheel detector [37]	-	*	Precision: 94 - Recall: 98
HoG cascade classifier[36]	*	-	Detection rate: 96
Feature-based motion detection[42]	*	*	Accuracy: 87.92 %
Double-stage detection[57]	*	-	Detection rate: 99
Fully connected network[43]	*	*	Accuracy: 93.75 %
HoG, SVM and motion vector[38]	*	-	Precision: 97 - Recall: 99
YOLOv2 [38]	*	-	Precision: 100 - Recall: 100
Integrating appearance and edge features[5]	*	-	Precision: 90.7 - Recall: 82.1
ours	*	*	Accuracy: 97 %

Besides, we train the distance estimator model on the KITTI dataset to train the distance estimator model on the collected data set. In the KITTI data set, unlike our collected data, the cars' images are from behind. So we train our proposed distance estimator on the KITTI data set just for evaluation purposes. From the KITTI data set, only images have been selected that were not occluded. A laser scanner sensor has obtained the distance of cars from the camera in KITTI dataset, so vehicles with a considerable distance from the camera are also included. The average car distance in the training data is about 50 m, and the maximum distance is 86 m. In the end, MAE is 92 cm. It may seem that the result obtained on KITTI is very different from the result obtained on the collected data set. Also, as much as the distance of the cars increases, it

gets harder to estimate. Our collected data set is very suitable because all the images are provided when the cars in the blind spots are not far.

As we mentioned, we train our blind spot tiny model by self-supervised learning method and collect a vast amount of raw video. We used an RGB USB camera to record videos. We drove for several hours on highways and in the city with the camera mounted on the car's right side mirror to collect various data. We also record videos during the day with different amounts of traffic. We evaluate our tiny model for a blind-spot warning system with 50 thousand frames. Our evaluation is based on the generated label by the rule-based inference engine. The accuracy obtained in this evaluation is 97 percent. In Table III, we compare similar systems with

Fig. 7: Mean Absolute Error and Mean Absolute Percentage Error of distance estimator training process with our collected data set

the proposed model. Our comparison measures are the ability to detect cars, detect two-wheelers, and accuracy results. All reported results are based on their collected data, and it is not fair to evaluate them together because of differences in the variety and difficulty of the collected data sets.

VII. CONCLUSIONS, LIMITATIONS, AND FUTURE WORK

We proposed a tiny model for a blind-spot warning system to discuss dangerous turning sides or lane changes. Also, the code and more output from the models will be available at <https://github.com/arashphn/BlindSpot>.

We start our journey by investigating previous blind spot driver assistant systems. They have similar functionality to object detection models, which also have real-time execution capabilities. We think detecting objects in the blind spot is not a generic way to classify dangerous situations from safe ones, and there are scenarios it is not possible to cover. To have a more reliable system, we need more data to train and evaluate. Another limitation of these systems is that the lack

of data and collecting human-labeled data sets is challenging and expensive.

In the dangerous situation detection task, we tried to solve the problem without considering the computational constraints. More awareness of context is needed to detect a dangerous situation. So we use an object tracker to track objects in consecutive frames and estimate the distance of cars. Our proposed model estimates car distances from the camera with a mean absolute percentage error of 6%. This model is suitable for our mission. [42] well described how optical flow could separate moving objects from the background. Our results show that using optical flow and CNN can easily detect moving objects in a stream of frames. Also, 97 % compliance with the SOTA object detection model for our proposed tiny model proves it.

Camera-based Blind Spot Driver assistant systems usually can not work properly at night or in low light conditions. It is one of the most critical limitations of our system and similar systems. Using other visual sensors such as the night vision camera to solve the mentioned problem may be a must.

REFERENCES

- [1] A. Ziebinski, R. Cupek, D. Grzechca, and L. Chrusczyk, "Review of advanced driver assistance systems (adas)," in *AIP Conference Proceedings*, vol. 1906, no. 1. AIP Publishing LLC, 2017, p. 120002.
- [2] A. Pashaei, M. Ghatee, and H. Sajedi, "Convolution neural network joint with mixture of extreme learning machines for feature extraction and classification of accident images," *Journal of Real-Time Image Processing*, vol. 17, no. 4, pp. 1051–1066, 2020.
- [3] G. Liu, L. Wang, and S. Zou, "A radar-based blind spot detection and warning system for driver assistance," in *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. IEEE, 2017, pp. 2204–2208.
- [4] Y. Zhao, L. Bai, Y. Lyu, and X. Huang, "Camera-based blind spot detection with a general purpose lightweight neural network," *Electronics*, vol. 8, no. 2, p. 233, 2019.
- [5] B.-F. Lin, Y.-M. Chan, L.-C. Fu, P.-Y. Hsiao, L.-A. Chuang, S.-S. Huang, and M.-F. Lo, "Integrating appearance and edge features for sedan vehicle detection in the blind-spot area," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 737–747, 2012.
- [6] J. D. Alonso, E. R. Vidal, A. Rotter, and M. Muhlenberg, "Lane-change decision aid system based on motion-driven vehicle tracking," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 5, pp. 2736–2746, 2008.
- [7] B. Li, T. Zhang, and T. Xia, "Vehicle detection from 3d lidar using fully convolutional network," *arXiv preprint arXiv:1608.07916*, 2016.
- [8] F. Zhang, D. Clarke, and A. Knoll, "Vehicle detection based on lidar and camera fusion," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2014, pp. 1620–1625.
- [9] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. J. Nunes, "Multimodal vehicle detection: fusing 3d-lidar and color camera data," *Pattern Recognition Letters*, vol. 115, pp. 20–29, 2018.
- [10] L. Bombini, P. Cerri, P. Medici, and G. Alessandretti, "Radar-vision fusion for vehicle detection," in *Proceedings of International Workshop on Intelligent Transportation*, 2006, pp. 65–70.
- [11] P. Pyykonen, A. Virtanen, and A. Kyytinen, "Developing intelligent blind spot detection system for heavy goods vehicles," in *2015 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*. IEEE, 2015, pp. 293–298.
- [12] M. Bertozzi, A. Broggi, P. Medici, P. P. Porta, and A. Sjogren, "Stereo vision-based start-inhibit for heavy goods vehicles," in *2006 IEEE Intelligent Vehicles Symposium*. IEEE, 2006, pp. 350–355.
- [13] S.-H. Chen and R.-S. Chen, "Vision-based distance estimation for multiple vehicles using single optical camera," in *2011 Second International Conference on Innovations in Bio-inspired Computing and Applications*. IEEE, 2011, pp. 9–12.
- [14] B.-F. Wu, W.-H. Chen, C.-W. Chang, C.-J. Chen, and M.-W. Chung, "A new vehicle detection with distance estimation for lane change warning systems," in *2007 IEEE Intelligent Vehicles Symposium*. IEEE, 2007, pp. 698–703.

- [15] J. Han, O. Heo, M. Park, S. Kee, and M. Sunwoo, "Vehicle distance estimation using a mono-camera for fca/aeb systems," *International journal of automotive technology*, vol. 17, no. 3, pp. 483–491, 2016.
- [16] S.-M. Chang, C.-C. Tsai, and J.-I. Guo, "A blind spot detection warning system based on gabor filtering and optical flow for e-mirror applications," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2018, pp. 1–5.
- [17] C. Wu, C. Lin, H. Lin, and H. Chung, "Adjacent lane detection and lateral vehicle distance measurement using vision-based neuro-fuzzy approaches," *Journal of applied research and technology*, vol. 11, no. 2, pp. 251–258, 2013.
- [18] H. Yang, Q. Zhou, J. Ni, H. Li, and X. Shen, "Accurate image-based pedestrian detection with privacy preservation," *IEEE Transactions on Vehicular Technology*, 2020.
- [19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.
- [20] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
- [21] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.
- [22] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 781–10 790.
- [23] Z. Meng, J. Li, Y. Zhao, and Y. Gong, "Conditional teacher-student learning," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6445–6449.
- [24] J. Li, M. L. Seltzer, X. Wang, R. Zhao, and Y. Gong, "Large-scale domain adaptation via teacher-student learning," *arXiv preprint arXiv:1708.05466*, 2017.
- [25] J. Xie, B. Shuai, J.-F. Hu, J. Lin, and W.-S. Zheng, "Improving fast segmentation with teacher-student learning," *arXiv preprint arXiv:1810.08476*, 2018.
- [26] D.-H. Hwang, S. Kim, N. Monet, H. Koike, and S. Bae, "Lightweight 3d human pose estimation network training using teacher-student learning," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 479–488.
- [27] A. Baevski, S. Schneider, and M. Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," *arXiv preprint arXiv:1910.05453*, 2019.
- [28] P. Liu, M. Lyu, I. King, and J. Xu, "Selfflow: Self-supervised learning of optical flow," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4571–4580.
- [29] L. Chen, P. Bentley, K. Mori, K. Misawa, M. Fujiwara, and D. Rueckert, "Self-supervised learning for medical image analysis using image context restoration," *Medical image analysis*, vol. 58, p. 101539, 2019.
- [30] R. Hadsell, P. Sermanet, J. Ben, A. Erkan, M. Scoffier, K. Kavukcuoglu, U. Muller, and Y. LeCun, "Learning long-range vision for autonomous off-road driving," *Journal of Field Robotics*, vol. 26, no. 2, pp. 120–144, 2009.
- [31] A. Lookingbill, J. Rogers, D. Lieb, J. Curry, and S. Thrun, "Reverse optical flow for self-supervised adaptive autonomous robot navigation," *International Journal of Computer Vision*, vol. 74, no. 3, pp. 287–302, 2007.
- [32] A. Kolesnikov, X. Zhai, and L. Beyer, "Revisiting self-supervised visual representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1920–1929.
- [33] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," *arXiv preprint arXiv:1803.07728*, 2018.
- [34] X. Zhang, N. Zheng, Y. He, and F. Wang, "Vehicle detection using an extended hidden random field model," in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2011, pp. 1555–1559.
- [35] Z. Sun, G. Bebis, and R. Miller, "Monocular precrash vehicle detection: features and classifiers," *IEEE transactions on image processing*, vol. 15, no. 7, pp. 2019–2034, 2006.
- [36] J. W. Baek, E. Lee, M.-R. Park, and D.-W. Seo, "Mono-camera based side vehicle detection for blind spot detection systems," in *2015 Seventh International Conference on Ubiquitous and Future Networks*. IEEE, 2015, pp. 147–149.
- [37] J. W. Baek, B.-G. Han, H. Kang, and Y. Chung, "Fast and reliable two-wheeler detection algorithm for blind spot detection systems," in *2017 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2017, pp. 513–516.
- [38] K. H. Jung and K. Yi, "Vision-based blind spot monitoring using rear-view camera and its real-time implementation in an embedded system," *Journal of Computing Science and Engineering*, vol. 12, no. 3, pp. 127–138, 2018.
- [39] B.-F. Wu, H.-Y. Huang, C.-J. Chen, Y.-H. Chen, C.-W. Chang, and Y.-L. Chen, "A vision-based blind spot warning system for daytime and nighttime driver assistance," *Computers & Electrical Engineering*, vol. 39, no. 3, pp. 846–862, 2013.
- [40] H. T. Niknejad, A. Takeuchi, S. Mita, and D. McAllester, "On-road multivehicle tracking using deformable object model and particle filter with improved likelihood estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 748–758, 2012.
- [41] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.
- [42] K.-T. Song and H.-Y. Chen, "Lateral driving assistance using optical flow and scene analysis," in *2007 IEEE Intelligent Vehicles Symposium*. IEEE, 2007, pp. 624–629.
- [43] D. Kwon, R. Malaiya, G. Yoon, J.-T. Ryu, and S.-Y. Pi, "A study on development of the camera-based blind spot detection system using the deep learning methodology," *Applied Sciences*, vol. 9, no. 14, p. 2941, 2019.
- [44] G. L. Foresti, V. Murino, and C. Regazzoni, "Vehicle recognition and tracking from road image sequences," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 1, pp. 301–318, 1999.
- [45] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.
- [46] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.
- [47] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.
- [48] J. Yao, S. Ramalingam, Y. Taguchi, Y. Miki, and R. Urtasun, "Estimating drivable collision-free space from monocular video," in *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2015, pp. 420–427.
- [49] M. Rezaei, M. Terauchi, and R. Klette, "Robust vehicle detection and distance estimation under challenging lighting conditions," *IEEE transactions on intelligent transportation systems*, vol. 16, no. 5, pp. 2723–2743, 2015.
- [50] J. Wang and E. Olson, "Apriltag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4193–4198.
- [51] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [52] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 13 001–13 008.
- [53] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014.
- [54] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [55] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6105–6114.
- [56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [57] M. Á. Sotelo and J. Barriga, "Blind spot detection using vision for automotive applications," *Journal of Zhejiang University-Science A*, vol. 9, no. 10, pp. 1369–1372, 2008.

Arash Pourhasan Nezhad received the M.Sc. from the Department of Computer Science, Amirkabir University of Technology. He has a broad interest in applying computer science, and mathematics to problems of interest in the fields of NLP, image processing, deep learning and analytics. He is currently working as a Machine Learning Engineer at Cafe Bazaar.

Mehdi Ghatte is an Associate Professor with Department of Computer Science, Amirkabir University of Technology, Tehran, Iran. His major is ITS, Smartphone-based Systems, Neural Network and Fuzzy Systems. He has written more than 130 papers on national and international journals and conferences. He is the author of two textbooks on optimization and decision support systems and two book-chapters on intelligent transportation systems. Currently, he is Associate Dean for Undergraduate Affairs of Faculty of Mathematics and Computer

Science and director of NORC.

Hedieh Sajedi is an Associate Professor with the School of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, Iran. Her major is Machine Learning, Artificial Intelligence, and Image Processing. She has written more than 132 papers on national and international journals and conferences.