Vision-Based Blind Spot Warning System by Deep Neural Networks

Víctor R. Virgilio G.¹(✉) , Humberto Sossa^{1,2} , and Erik Zamora¹

¹ Instituto Politécnico Nacional - CIC, Av. Juan de Dios Batiz S/N,
Gustavo A. Madero, 07738 Ciudad de Mexico, Mexico
vvirgiliog@gmail.com, hsossa@cic.ipn.mx, ezamorag@ipn.mx

² Tecnológico de Monterrey, Unidad Guadalajara, Av. Gral. Ramón Corona 2514,
45201 Zapopan, Jalisco, Mexico

Abstract. Traffic accidents represent one of the most serious problems around the world. Many efforts have been concentrated on implementing Advanced Driver Assistance Systems (ADAS) to increase safety by reducing critical tasks faced by the driver. In this paper, a Blind Spot Warning (BSW) system capable of virtualizing cars around the driver's vehicle is presented. The system is based on deep neural models for car detection and depth estimation using images captured with a camera located on top of the main vehicle, then transformations are applied to the image and to generate the appropriate information format. Finally the cars in the environment are represented in a 3D graphical interface. We present a comparison between car detectors and another one between depth estimators from which we choose the best performance ones to be implemented in the BSW system. In particular, our system offers a more intuitive assistance interface for the driver allowing a better and quicker understanding of the environment from monocular cameras.

Keywords: ADAS (advanced driver-assistance systems) · BSW (blind spots warning) · SIDE (single image depth estimation) · Object detection · Neural networks

1 Introduction

Transport can be involved in daily traffic accidents which are one of the most serious problems currently facing modern societies. According to 2017 figures from the World Health Organization (WHO), each year around 1.3 million people die in road accidents worldwide, and between 20 and 50 million suffer non-fatal injuries that cause disabilities [17]. According to data from the National Institute of Public Health (INSP), Mexico ranks seventh in the world and third in the Latin American region in terms of road deaths, with 22 deaths of young people between 15 and 29 years of age per day [9].

Road specialists and road safety experts report that behind every vehicle accident the human factor is involved in 90% [6]. So for several years, car manufacturers have implemented technologies such as ADAS which assist the driver in

the driving process. ADAS goal is to increase automobile safety and road safety in general using Human-Machine Interfaces (HMI). These systems use multiple sensors (radar, lidar, camera, GPS, etc.) to identify the environment with which the vehicle interacts.

When driving a vehicle the driver depends on rear-view mirrors and body movements to observe other vehicles approaching, however, this practice represents risks due to the generation of areas where vision is partially or completely occluded, these areas are called “blind spots”. Due to the large number of accidents caused by this situation, BSW systems have been developed which provide the driver with information about the vehicles around him to avoid possible collisions.

This document is organized as follows. Section 2 describes details of the existing State-of-the-Art (SOTA) of BSW systems and the type of processing they perform. Section 3 presents the different techniques and technologies implemented in the proposed BSW system, such as the neural models, the applied transformations and the visualization platform used. Section 4 shows the results obtained qualitatively and quantitatively from the implemented technologies. Section 5 the conclusions obtained with the development of this work are presented.

2 Related Work

Although the objective is the same (alerting the driver to the presence of vehicles in occlusion areas) BSW systems can be developed from different technologies and implement different sensors such as: ultrasonic, optical, radar, cameras, etc; in addition, they can provide visual (e.g. outside image), audio (e.g. voice prompt) or tactile (e.g. steering wheel vibration) information to indicate that it is not safe to change lanes. Typically there are two basic approaches to obtain and processing information: range-based and vision-based.

Works such as presented in [14,16,22,23] describe range-based systems that implement ultrasonic or radar devices mounted around the vehicle to estimate the distance of approaching objects, subsequently alert the driver by means of indicators on the side mirrors.

Vision-based systems aim to obtain information from the environment using cameras and then perform image analysis for obstacles detection while driving. Most BSW systems employ classic image processing techniques for their development. Such as [3,13,18,19,23] histogram of oriented gradients (HOG), filters for edge detection, entropy, optical flow, Gabor’s filter, among others are used to extract useful information and techniques such as clustering and vector support machines [13,18] to classify where vehicles are. In [11] the concept of depth estimation is implemented to determine whether a vehicle is near or far from the driver’s vehicle, they make use of features such as texture and blur in the image, and techniques such as principal component analysis (PCA) and discrete cosine transformation.

In recent years, neural models have been implemented for the classification and detection of objects in images due to good performance obtained. In [15,21]

fully connected neural networks (FCN) are used for vehicle detection in blind spot areas, in addition to techniques such as HOG, heat mapping and threshold levels for pre-processing of images.

Other types of BSW systems have been developed with more complex neural models; such is the case of [26] where first the objects are located by classic image segmentation, then the candidates are classified with a Convolutional Neural Network (CNN) and the vehicle is tracked using optical flow analysis. On the other hand in [27] blind spot vehicles are treated as a classification problem in which a CNN takes full responsibility for classifying whether or not a vehicle exists in the predetermined area.

Lastly, in [19] a BSW system is developed implementing multi-object tracking (MOT) from a fusion of sensors, including cameras, LIDAR, among others; in addition, techniques such as decision by Markov models and reinforcement learning for information processing are applied.

3 Proposed Method

We propose a BSW system capable of providing a driver assistance interface that virtualizes the cars around him on a 3D platform. The system contains (i) a neuronal model for car detection, (ii) a neuronal model for depth estimation, (iii) a processing module to generate car location and (iv) a graphical interface module to visualize the cars, as illustrated in Fig. 1.

The presented system was implemented using monocular images from the KITTI database [8]. KITTI provides stereoscopic images (1242×375) of front view using cameras mounted on top of the vehicle at a rate of 10 frames per second. All scenes are recorded in similar weather conditions during the day.

3.1 Car Detection

For car detection in the images, two very popular neural architectures were tested: YOLOv3 and Detectron2.

YOLOv3 [20] is a neural model for object detection that processes approximately 30 images per second in COCO test-set obtaining an average precision of 33% and consists of 53 convolutional layers (Darknet 53). This model has several advantages over systems based on classifiers and sliding window, for example, it examines the entire image at the time of inference so that predictions have information about the overall context of the image. In addition, it develops the predictions with a single evaluation of the image which makes it a very fast network.

Detectron2 is a neural model developed by Facebook AI Research that implements SOTA object detection algorithms. It is a rewrite of the previous version, Detectron, and originates from the benchmark Mask R-CNN [12]. The average precision of this model is 39.8% obtained in COCO test-set.

Fig. 1. Proposed system diagram. The images are passed through the detector to infer areas where there are cars, subsequently distance is estimated in the previously detected areas using the neuronal model (depth, Z axis) and the BEV transformation (horizontal, X axis). Later, the information is given to the 3D graphical interface to visualize the cars.

3.2 Depth Estimation

Considering that car detecting in images does not give us clear information about the distance they are, which is fundamental for the understanding of a scene, a single-image depth estimation (SIDE) has been implemented to know the distance in the Z-axis (deep). Different neuronal models were considered.

DenseDepth [2] is a model that consists of a convolutional neural network for computing a high-resolution depth map given a single RGB image. Following a standard encoder-decoder architecture, they leverage features extracted using high performing pre-trained networks when initializing the encoder along with augmentation and training strategies that lead to more accurate results.

MonoDepth2 [10] is a depth estimation network is based on the general U-Net architecture with skip connections, enabling to represent both deep abstract features as well as local information. They use a ResNet18 as encoder, unlike the larger and slower DispNet and ResNet50 models used in existing SOTA.

monoResMatch [24] is a deep architecture designed to infer depth from a single input image by synthesizing features from a different point of view, horizontally aligned with the input image, performing stereo matching between the two cues. In contrast to previous works sharing this rationale, this network is the first trained end-to-end from scratch.

3.3 Car Location

Following the steps described in [25], and using OpenCV, we apply a bird's eye view transformation (BEV) to estimate the distance of the vehicles in the X axis (horizontal). Then we organize and give the detections and estimated distances to virtualization platform.

3.4 3D Graphical Interface

In this module, we generate a 3D graphical interface to achieve a more natural and intuitive interface for the driver. Unlike the typical 2D interface, which BEV is presented, UBER’s interface [1] virtualizes the cars in 3D which represents an environment similar to the one humans face daily, so it directly impacts on the speed of assimilation/interpretation of the environment.

4 Results

In this section we present the qualitative and quantitative results obtained by the neuronal models for car detection and depth estimation, as well as ones obtained by the BEV transformation. In addition, the final results of the BSW system are shown through the interface generated by the UBER platform.

Although the BSW system aims to process complete information on the environment around the vehicle, the results presented are the first tests carried out using the KITTI database. However, the system could be evaluated with another database that offers images of the complete environment using cameras located at different points of the vehicle as well as scenes recorded in more challenging weather conditions.

This work aims to demonstrate the feasibility of using deep neural models in BSW systems, the experiments were individually carried out offline using a Tesla P100 (16 GB) GPU.

Car Detection. Following [4], we evaluate car detectors using 3,769 images for validation set at KITTI 2D detection benchmark [8]. Evaluation is done for car class in three regimes: Easy, Moderate and Hard, which contain objects of different box sizes, and different levels of occlusion and truncation. The results in Table 1 show that, in general, car detection is feasible even in high complexity situations such as moderate and hard KITTI levels, with 0.07 s for images with few detected cars (less than 5) and 0.1 s for many cars (more than 10). Figure 2 shows some results of detectors in the validation set.

The main reason why the AP is below 50% is because both models have not been retrained in the KITTI database; instead, these models have been pre-trained in the COCO database with almost 100 classes. In addition, both neural models are the most popular and intuitive to implement but not the best performing in the SOTA. Based on the experimental results we conclude that Detectron2 is a better choice for this type of problem in a BSW system.

Table 1. Performance on KITTI validation set for Car class using the KITTI standar metric, Average Precision Metric (AP).

	AP (%)			Time (s)	
	Easy	Moderate	Hard	min	max
YOLOv3	47.3	41.4	30.4	0.07	0.1
Detectron2	46.1	49.5	39.9	0.07	0.1

Fig. 2. Groundtruth (first row), YOLOv3 detection (second row), Detectron2 detection (third row) are presented. It is possible to observe that Detectron2 adjusts bounding boxes better and detects cars that Yolov3 does not.

Depth Estimation. SOTA single-image depth estimators were compared in KITTI's benchmark [7]. Table 2 shows that neural models compared present a very good and similar performance, which demonstrates that they are a good alternative to the problem of depth estimation; it is worth mentioning that MonoDepth2 processes information in a considerably less amount of time than the other methods, which would be important when testing the system on embedded hardware.

Implementing SOTA depth estimation models allows us to obtain more precise information about the location of previously detected trolleys. Based on the experimental results we conclude that DenseDepth is the best option to depth estimation problem in a BSW. Figure 3 shows some results of depth estimators.

BEV Transformation. Some results of the BEV transform are presented in Fig. 4. Later, the information was organized to be sent to the graphic interface platform.

Table 2. Quantitative evaluation on the test set of KITTI dataset [7] using the standard six metrics used in [5], maximum depth: 80 m.

	Higher is better			Lower is better				
	$\delta 1$	$\delta 2$	$\delta 3$	rel	sq. rel	rms	log10	Time (s)
DenseDepth	0.886	0.965	0.986	0.093	0.589	4.170	0.171	0.08
MonoDepth2	0.890	0.961	0.981	0.096	0.673	4.351	0.184	0.007
monoResMatch	0.876	0.958	0.980	0.106	0.806	4.630	0.193	0.66

Fig. 3. The original image (first row), depth estimation by DenseDepth (second row), depth estimation by MonoDepth2 (third row), depth estimation by MonoResMatch (fourth row) are presented. It is possible to observe that, in the case of defined shapes (such as cars and people), DenseDepth has a higher level of detail than the rest.

Blind Spot Warning System. To test the BSW system we use the previously chosen neural models, then we apply the BEV transformation and give the data to the UBER platform to generate the 3D graphical interface.

Figure 5 shows the result of the BSW system, where different 3D views generated by the graphical interface are displayed in addition to the indication (by color) of the closest cars, which offers greater assistance and comfort to the driver in terms of how he or she perceives the environment.

Fig. 4. BEV transformation results. The original image (first row) and Bird's Eye View transformation (second row) are presented.

Fig. 5. The original image (top left), driver's view (center left), bird's view (bottom left) and perspective view (right) generated in the graphical interface are presented.

5 Conclusion

A single-image BSW system was developed based on artificial intelligence technologies such as neural models for object detection and depth estimation. In addition, the visualization system development on a 3D graphics platform offers the driver a much more intuitive interface than SOTA BSW systems and presents a much faster way to understand the behavior of the vehicles around.

This work shows a BSW system with complex interfaces through the unique use of vision sensors, which represents a cost reduction compared to range-based systems that employ sensors such as LIDAR or radar. Finally, the presented system contributes to the approach of understanding the scene, since it offers an alternative of car virtualization that works as a reference for the perception of the environment.

Acknowledgments. V. Virgilio acknowledges CONACYT for the scholarship granted towards pursuing his postgraduate studies. H. Sossa and E. Zamora would like to acknowledge the support provided by SIP-IPN (grant numbers 20190166, 20190007, 20200651 and 20200630).

References

1. Chen, X., Lisee, J., Wojtaszek, T., Gupta, A.: Introducing AVS, an open standard for autonomous vehicle visualization from uber. Accessed, February 2020. <https://eng.uber.com/avs-autonomous-vehicle-visualization/>
2. Alhashim, I., Wonka, P.: High Quality Monocular Depth Estimation via Transfer Learning (2018). <http://arxiv.org/abs/1812.11941>

3. Chang, S.M., Tsai, C.C., Guo, J.I.: A blind spot detection warning system based on gabor filtering and optical flow for e-mirror applications. In: Proceedings - IEEE International Symposium on Circuits and Systems 2018-May, pp. 1–5 (2018). <https://doi.org/10.1109/ISCAS.2018.8350927>
4. Chen, X., Kundu, K., Zhu, Y., Ma, H., Fidler, S., Urtasun, R.: 3D object proposals using stereo imagery for accurate object class detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(5), 1259–1272 (2018). <https://doi.org/10.1109/TPAMI.2017.2706685>
5. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. *Adv. Neural Inf. Process. Syst.* **3**(January), 2366–2374 (2014)
6. Fundación Carlos Slim: Las causas más comunes en accidentes de tránsito - Seguridad Vial. Accessed, February 2020. <http://fundacioncarlosslim.org/12022-2/>
7. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **32**(11), 1231–1237 (2013). <https://doi.org/10.1177/0278364913491297>
8. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the KITTI vision benchmark suite. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 3354–3361 (2012). <https://doi.org/10.1109/CVPR.2012.6248074>
9. Gobierno de México: Accidentes viales, primera causa de muerte en los jóvenes. Accessed, February 2020. <https://www.gob.mx/salud/prensa/accidentes-viales-primer-causa-de-muerte-en-los-jovenes>
10. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.: Digging into self-supervised monocular depth estimation (1), 3828–3838 (2018). <http://arxiv.org/abs/1806.01260>
11. Guo, Y., Kumazawa, I., Kaku, C.: Blind spot obstacle detection from monocular camera images with depth cues extracted by CNN. *Automot. Innov.* **1**(4), 362–373 (2018). <https://doi.org/10.1007/s42154-018-0036-6>
12. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **42**(2), 386–397 (2020). <https://doi.org/10.1109/TPAMI.2018.2844175>
13. Jung, K.H., Yi, K.: Vision-based blind spot monitoring using rear-view camera and its real-time implementation in an embedded system. *J. Comput. Sci. Eng.* **12**(3), 127–138 (2018). <https://doi.org/10.5626/JCSE.2018.12.3.127>
14. Kedarkar, P., Chaudhari, M., Dasarwar, C., Domakondwar, P.B.: Prevention device for blind spot accident detection and protection. *Int. Res. J. Eng. Technol. (IRJET)* **6**(1), 624–627 (2019). <https://www.irjet.net/archives/V6/i1/IRJET-V6I1112.pdf>
15. Kwon, D., Malaiya, R., Yoon, G., Ryu, J.T., Pi, S.Y.: A study on development of the camera-based blind spot detection system using the deep learning methodology. *Appl. Sci.* **9**(14), 2941 (2019). <https://doi.org/10.3390/app9142941>. <https://www.mdpi.com/2076-3417/9/14/2941>
16. Liu, G., Wang, L., Zou, S.: A radar-based blind spot detection and warning system for driver assistance. In: Proceedings of 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference, IAEAC 2017, pp. 2204–2208 (2017). <https://doi.org/10.1109/IAEAC.2017.8054409>
17. Organización Mundial de la Salud: Accidentes de tránsito. Accessed February 2020. <https://www.who.int/es/news-room/fact-sheets/detail/road-traffic-injuries>
18. Ra, M., Jung, H.G., Suhr, J.K., Kim, W.Y.: Part-based vehicle detection in side-rectilinear images for blind-spot detection. *Expert Syst. Appl.* **101**, 116–128 (2018). <https://doi.org/10.1016/j.eswa.2018.02.005>

19. Rangesh, A., Trivedi, M.M.: No blind spots: full-surround multi-object tracking for autonomous vehicles using cameras & LiDARs, pp. 1–12 (2018). <http://arxiv.org/abs/1802.08755>
20. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement (2018). <http://arxiv.org/abs/1804.02767>
21. Rusiecki, A., Roma, P.: Framework of blind spot information system using feedforward neural networks, March 2016. <https://doi.org/10.13140/RG.2.1.3252.3921/1>
22. Sheets, D.: Semi-Truck Blind Spot Detection System Group **32** (2016)
23. Tigadi, P., Gujanatti, P.B., Patil, R.: Survey on blind spot detection and lane departure warning systems. *Int. J. Adv. Res. Eng.* **2**(5), 2015 (2015). <https://pdfs.semanticscholar.org/aae1/85ec8c8caef29389d5b6253c0b2c9bdf4a0c.pdf>
24. Tosi, F., Aleotti, F., Poggi, M., Mattoccia, S.: Learning monocular depth estimation infusing traditional stereo knowledge. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2019-June, pp. 9791–9801 (2019). <https://doi.org/10.1109/CVPR.2019.01003>
25. Tuohy, S., O’Cualain, D., Jones, E., Glavin, M.: Distance determination for an automobile environment using inverse perspective mapping in OpenCV. In: IET Conference Publications 2010(566 CP), pp. 100–105 (2010). <https://doi.org/10.1049/cp.2010.0495>
26. Wu, L.T., Lin, H.Y.: Overtaking vehicle detection techniques based on optical flow and convolutional neural network. In: VEHITS 2018 - Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems 2018-March(Vehits), pp. 133–140 (2018)
27. Zhao, Y., Bai, L., Lyu, Y., Huang, X.: Camera-based blind spot detection with a general purpose lightweight neural network. *Electronics* **8**(2), 233 (2019). <https://doi.org/10.3390/electronics8020233>