Indonesian Journal of Electrical Engineering and Computer Science
Vol. 36, No. 3, December 2024, pp. 1532~1542
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v36.i3.pp1532-1542

Enhanced driving assistance: automated day and night vehicle detection system utilizing convolutional neural networks

Abdelmoghit Zaarane, Ibtissam Slimani, Mourad Elhabchi, Issam Atouf
Faculty of Sciences Ben M'sick, Hassan II University of Casablanca, Casablanca, Morocco

Article Info

Article history:

Received Apr 29, 2024
Revised Aug 24, 2024
Accepted Aug 31, 2024

Keywords:

Convolutional neural network
Day-time detection
Driving assistance systems
Image processing
Night-time detection
Two-dimensional discrete wavelet transform
Vehicle detection

ABSTRACT

This paper presents an enhanced real-time vehicle detection system using convolutional neural networks (CNNs) for both daytime and night-time conditions. Initially, the system determines the time of capture by analyzing the upper part of input images. For daytime detection, it uses normalized cross-correlation and two-dimensional discrete wavelet transform (2D-DWT) techniques. Night-time detection involves identifying vehicle lamps through color thresholding and connected component techniques, followed by symmetry analysis and CNN classification. The dataset for training includes images from the Caltech Cars, AOLP, KITTI Vision, and night-time vehicle detection datasets, ensuring robust performance across various lighting conditions. Experiments demonstrate the system's high accuracy, achieving 99.2% during the day and 98.27% at night, meeting real-time requirements and enhancing driving assistance systems' reliability.

Corresponding Author:

Abdelmoghit Zaarane
Faculty of Sciences Ben M'sick, Hassan II University of Casablanca
Av Driss El Harti Sidi Othmane Casablanca B.P 7955, Morocco
Email: z.abdelmoghit@gmail.com

1. INTRODUCTION

Recently, the driving assistance systems (DAS) has gained a great consideration in the research field, in light of the urgent need to automate driving systems due to the strong growth in the accident numbers while most of these accidents are caused by human causes. Vehicle detection [1] is one of the most challenging tasks in the DAS where it can be used for several purposes, such as calculating the distances and speeds of other vehicles [2], [3] and the possibility of overtaking, in order to warn the driver to slow down and avoid collisions. However, there are several difficulties and constraints in vehicle detection systems due to the dynamic environment and the complex and rapid movement of objects interfering in the road. Several researches are done to develop efficient vehicle detection systems [4]. However, most of researchers carried out their researches to deal with vehicle detection only in day-time in suitable conditions, where the assistance is more needed in night-time.

Concerning the work related to the detection during day-time. Ruan *et al.* [5] present a system for vehicle detection using wheel part identification. Their proposed method consists of two main parts, feature extraction and vehicle detection. Then they detect and locate vehicles using three different submodules: the region of interest (ROI) segmentation module, the wheel identification module, and the vehicle detection module. Wei *et al.* [6] suggest a vehicle recognition method based on Haar and histogram of oriented gradients (HOG) features, where they segment the regions of interest based on a combination between Haar and HOG features then they classify them using AdaBoost and support vector machines (SVMs) classifier.

However, the most recent work has always been focused on the use of YOLO algorithms for detection, you only look once, which is a deep learning method. This YOLO algorithm was proposed by

Redmon *et al.* [7]. It is a single-step approach that translates an object's boundary box position problem directly into an object's regression problem without generating candidate boxes. In the YOLO framework, the image is divided into a certain number of grids. Each grid has the responsibility to estimate the objects within the grid that have a center. Some of the authors who have used YOLO, Chen and Li [8] use on the YOLOv3 algorithm and the single-shot detector (SSD) algorithm for vehicle detection. They first process the input image for training by using k-means for clustering the bounding box of the candidates. Then, they trained the vehicle detection models by using YOLOv3 and SSD algorithms. Also [9] are developed a YOLOV3 algorithm that has optimal performance relative to the performance of deep learning algorithms, for autonomous vehicle image recognition.

Concerning the work related to the detection during night-time. Most existing vehicle detection systems and methods focus mainly on detecting vehicles in daylight conditions. However, statistics show that more than half of all dangerous road accidents occur at night. Hence, more attention needs to be paid to work on vehicle detection at this time of day [10].

Among the works focusing on night detection, we find [11]. They propose a front vehicles detection system during the night-time on the basis of both laser and video information. Firstly, they pre-processed both laser information and video frames using the region growth and threshold area expulsion algorithm. Next, they extracted the characteristics of the frontal vehicles using the Gabor filter which uses the principle of uncertainty, and they obtained the frontal vehicle distances via the laser point cloud. Lastly, they classify the detected front vehicles based on the SVM algorithm.

Kuang *et al.* [12] a night-time method is proposed for detecting vehicles using the combination of two different methods to increase the accuracy of the system. A region of interest extraction approach to detect vehicle light and a night-time image improvement approximation on the basis of enhanced multi-scale retinex (MSR) for extracting a precise region of interest and improving the images to accurately detect vehicles at night. Hemmati *et al.* [13] present a novel method for night-time vehicle detection based on deep belief networks (DBN) for detecting backlights in image thresholds. And then some detected taillights are selected and fed to an SVM classifier on the basis of the size characteristics retrieved and their distance to localize the vehicle.

Combining the MobileNet v2 and YOLO v3, Huang *et al.* [14] propose M-YOLO which is a novel deep neural network model. Initially, they extract the characteristics of M-YOLO by using the lightweight network MobileNet v2. Subsequently, they use the algorithm “K-means” to reorganize the database and get suitable and appropriate anchor boxes. Third, they use the EIoU loss function to optimize the model continuously.

Despite advancements in previous works, several problems remain unresolved, the main unresolved problem is the inconsistency in detection performance across different lighting scenarios [15]. Our work aims to fill this gap by developing a system that maintains high detection accuracy regardless of ambient light. Specifically, we focus on improving detection rates and reducing false positives, thereby enhancing overall system reliability.

Within this regard, the goal of this paper is to propose an enhanced driving assistance for vehicle detection during day and night. The work presented in this paper is based on a different efficient algorithm such as normalized cross-correlation, two-dimensional discrete wavelet transform (2D-DWT), and convolutional neural networks (CNNs). The contributions of this paper are multifaceted, resides in addressing several gaps in the existing literature on automated vehicle detection systems. In this paper, we present an enhanced driving assistance system that leverages a CNN to detect vehicles with high precision and recall in both day and night conditions. Unlike previous works that often focus on either daytime or night-time detection, our approach provides a comprehensive solution that excels in varying lighting conditions.

For clarity of presentation, the paper follows a structured organization. Section 1 provides the introduction. Section 2 describes the proposed vehicle detection method. Section 3 provides the results of the experiments, while section 4 offers the concluding remarks for the paper.

2. METHOD

The proposed method is a pure image processing system based on many experiments and researches in this field. In this article, we present our enhanced system for detecting vehicles in day-time and night-time. The goal is to obtain an efficient vehicle detection system during day and night with highly reliable results. The principle of this system is described in the overall flow diagram presented in the Figure 1. To detect vehicles, we need to determine when the input images are captured. Therefore, we first perform processing on the upper part of the input images, which represents the sky, in order to detect whether they were captured during the day or at night, note that the elevation angle of the camera equals zero. The idea behind this processing is to compute the average values of the intensities of the upper third of the input images for each adjustable number of images. Where the input images are considered as daytime scenes, if their average is greater than a predefined threshold T1, we perform the daytime detection algorithm. Conversely, if their average is less than other predefined threshold T2, they are considered as night scenes, so we perform the night detection algorithm (where T1>T2). Any other scenario indicates that the nature of the images remains unchanged.

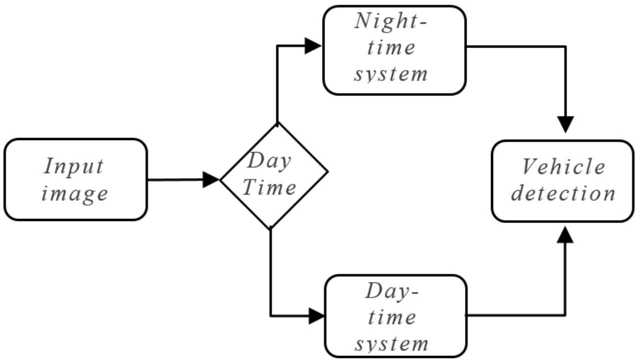

Figure 1. Overall flow diagram of the proposed system

2.1. Day-time vehicle detection method

The first part of this section focuses on vehicle detection during the day, highlighting the process's two main stages: hypothesis generation and hypothesis verification. This two-step approach ensures that the vehicle detection system is both efficient and reliable, providing accurate results under daylight conditions. By breaking down the task into these stages, the system can more effectively manage the complexities of detecting vehicles in varied environments.

2.1.1. Hypothesis generation stage

The performance of this stage is the key to achieve the efficiency of the entire system, where the good generation of hypothesis has a direct impact on the result of the verification stage. For this reason, we have performed several algorithms in order to correctly generate all the hypotheses. In this stage we based on pattern detection in order to locate the potential vehicles (hypotheses). Initially, preprocessing is conducted on input images by converting them from red, green and blue (RGB) space to hue saturation value (HSV) space as shown in Figure 2 to segregate luminosity information from color information. Subsequently, histogram equalization is applied to the luminosity information (V channel) to enhance contrast, as depicted in Figure 3. Figure 4 presents the resulting image after histogram equalization juxtaposed with the original image. As evident from both Figures 3 and 4, a substantial enhancement in contrast is observed, which is expected to facilitate subsequent processes.

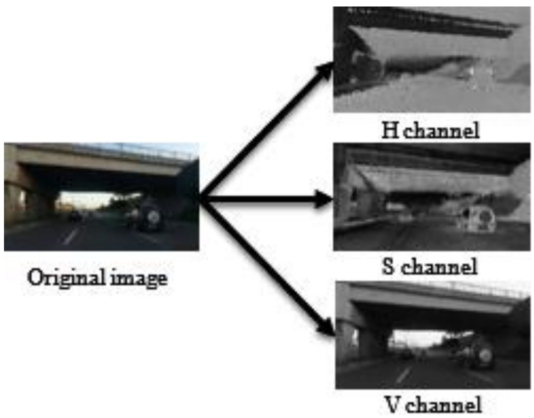

Figure 2. RGB to HSV conversion

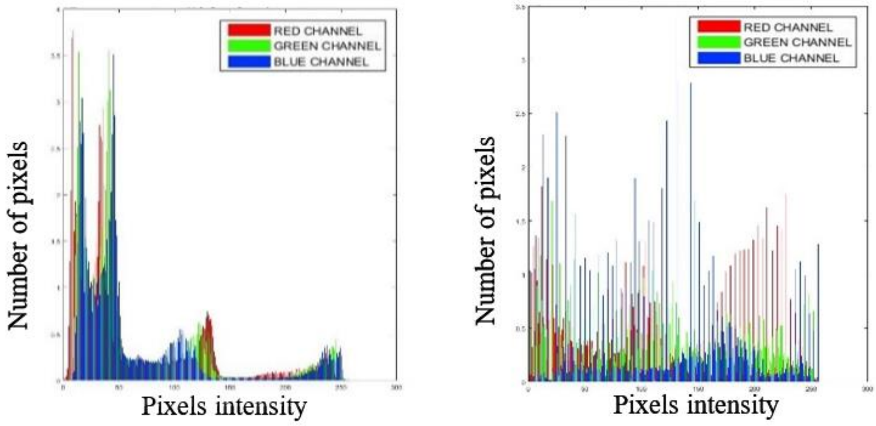

Figure 3. Histogram equalization

Figure 4. Image resulted after applying histogram equalization

Following this, key features, specifically edges, are extracted using the Canny edge detector [16] to expedite processing time and enhance pattern detection accuracy. Subsequent to edge extraction, the technique of fast normalized cross-correlation is employed to identify vehicle patterns within the image. This technique involves applying fast normalized cross-correlation to the filtered input image and model images, after subjecting them to the Canny edge detector. Normalized cross-correlation [17] is a widely utilized method for quantifying the resemblance between two images, exhibiting notable efficacy in pattern detection, particularly when combined with the Canny edge detector. Consequently, potential vehicles within the input images are identified based on the degree of similarity between model images and test images. An illustrative example is depicted in Figure 5. The best match occurs when the model and test images have a maximum correlation value. Employing this technique enables the discovery of multiple potential vehicle locations.

Figure 5. Image model examples

2.1.2. Hypothesis verification stage

Hypothesis verification stage is not less important than the first stage which is responsible for the classification of the hypotheses generated in the previous stage. This stage is composed of two principal parts, feature extraction and classification. In the first part, we extract the important characteristics using the third level of the two-dimensional discrete wavelet transformation on the generated candidates, where we have used Haar wavelet transform. This transformation is a very important and powerful technique to represent input data at different scales and frequencies. The 2D-DWT [18] as a mathematical tool is utilized to decompose an image into four elements and extract the characteristics of the image using the most significant element of these four elements. It splits the image to four parts, three parts with high frequencies and one part with low frequency. In this part, we relied on the low frequency sub-images to feed the classification part. For the classification part, we based on the CNN to verify the generated hypotheses based on their extracted features for classifying them into true or false hypotheses (vehicles or non-vehicles).

CNNs [19] for classification are the currently best performing models and consist of two different parts. The input is an image given as a matrix of pixels. This matrix contains two elements for a grayscale image. The third element is the color. The initial part of a CNN is the convolutional part. It works a feature extraction system of images. The result of the convolutional part is fed into the second part, which consists of fully connected layers. This part's task consists of connecting the features obtained previously in order for classifying the images. The result is a last layer with only a neuron for each class. The obtained results are typically normalized to 0 and 1, with the sum of these value equal to 1, in order to generate a probability distribution over the classes.

Building a novel CNN is very intensive in terms of knowledge, hardware, and volume of labeled datasets. For this reason, there are specialist researchers who focus on enhancing CNNs. They make their technological improvements public, together with information about the trained networks in the database, which can be exploited for transfer training. There are several powerful models that could be used for transfer learning such as the model used in this work, Inception v4 [20]. This model trained by google using 1.2 million images classified in 1,000 categories. We chose to use this model because of its low consumption of time and resources, and also because of its high accuracy and high performance of classification.

2.2. Night-time vehicle detection method

The second part of this section focuses on vehicle detection at night, which is done in two main stages: lamp candidate generation and lamp candidate verification. First, the system identifies possible vehicle lights in the image by detecting bright spots. Next, it verifies these potential lights to confirm they are actual vehicle lamps. This process, shown in Figure 6, helps improve the accuracy of detecting vehicles in low-light conditions.

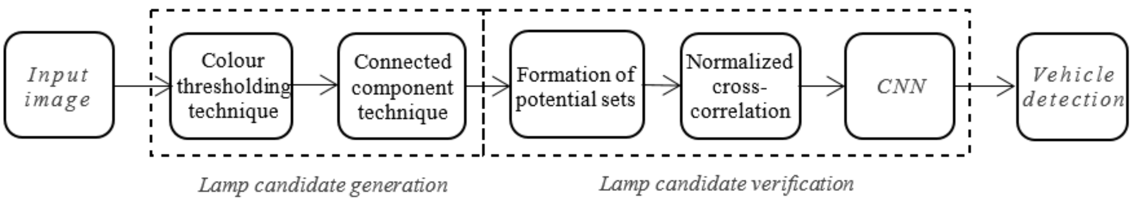

Figure 6. General flow chart of the vehicle detection system at night

2.2.1. Lamp candidate generation stage

To generate candidate lamps, two essential techniques are employed: the color thresholding technique and the connected components technique. Initially, significant features, namely colors, are extracted using a color thresholding method. This method involves thresholding the input images based on the color intensities of image pixels using predefined thresholds. Specifically, bright white regions representing potential headlamps and bright red regions representing potential rear lamps are preserved, while all other regions are disregarded. The process begins by converting the input images from the RGB color space to the HSV color space, which is optimal for color thresholding. Subsequently, color thresholding is applied pixel by pixel in the HSV color space to retain only the relevant regions. Pixels are classified as potential parts of red regions if the hue (H) falls within the range of 345 to 10, saturation (S) falls within 55 to 100, and value (V) falls within 50 to 100. Conversely, pixels are identified as potential parts of white regions if S falls within 0 to 20 and V falls within 90 to 100, regardless of the H value. Some regions may contain gaps that could fragment them into multiple regions. To address this issue, morphological operations are performed to fill and restore missing pixels within these gaps. Specifically, dilation with a 3x3 kernel followed by erosion with a 3×3 kernel is employed.

Subsequently, each remaining region is extracted as a component (candidate) using the connected components technique [21]. This technique involves scanning the binary image containing the regions pixel by pixel to group all adjacent pixels of each bright region into one connected component and provide its information, including size and coordinates. Finally, the extracted components undergo filtration to eliminate excessively large or small components that are unlikely to represent headlamps or rear lamps.

2.2.2. Lamp candidate verification stage

In order to verify the generated candidates, we first group them into sets of pairs according to specific criteria, where potential lamps, which have the same color and are relatively close to each other and aligned horizontally, are gathered in the same set. Vehicle lamps have different shapes that vary from one vehicle to another, making them quite difficult to classify. However, the pair of lamps in each vehicle is symmetrically identical. Therefore, to pre-classify candidate lamps as potential vehicles, we pair the most symmetrically identical lamp candidates by performing a normalized cross-correlation technique between a mirrored lamp candidate and the other lamps candidates in the same set. Pairs that are highly correlated are considered as potential vehicle.

Potential vehicles are classified as either vehicles or non-vehicles using the CNN classifier. The CNN is retrained by a dataset that contains positive samples as well as negative samples. The positive samples contain images of vehicles taken from different videos in different locations, and the negative samples contain different images of various elements or areas of the natural scene.

2.3. Retraining CNN model

2.3.1. Data collection and preprocessing

The dataset used for retraining the used CNN model is a custom-built vehicle detection dataset that includes diverse day and night conditions. The dataset was compiled from several publicly available datasets, ensuring a comprehensive collection of vehicle images. Specifically, it includes images from the Caltech Cars dataset [22], the AOLP dataset [23], the KITTI Vision Benchmark Suite [24], and the night-time vehicle detection dataset [25]. In total, the dataset comprises 10,000 images, with equal distribution between daytime and night-time scenes, that ensures a robust model performance across different lighting scenarios.

Data preprocessing was critical to ensure high-quality inputs for the Inception V4 model. The following steps were undertaken:

- Image resizing: all images were resized to 299×299 pixels to match the input dimensions required by the Inception V4 model.
- Normalization: pixel values were normalized to the range [0, 1] to facilitate faster convergence during training.
- Data augmentation: techniques such as horizontal flipping, rotation, and brightness adjustment were applied to augment the dataset and improve model robustness.

2.3.2. Inception v4 architecture

We employed the Inception V4 model, utilizing transfer learning to leverage pre-trained weights on the ImageNet dataset. The Inception V4 architecture is known for its efficiency and high performance in image classification tasks. The model was fine-tuned on our custom vehicle detection dataset.

- Input layer: accepts 299×299 RGB images.
- Pre-trained layers: all layers up to the final inception module were frozen to retain the learned features from ImageNet.
- Fully connected layers: added custom fully connected layers with 1,024 and 512 neurons respectively, followed by rectified linear unit (ReLU) activation.
- Output layer: a softmax layer with two neurons for binary classification (vehicle present or not).

This model was trained using the Adam optimizer with a learning rate of 0.0001. The loss function used was binary cross-entropy, and the model was trained for 30 epochs with a batch size of 32. To retrain this model, we split the dataset into training (70%), validation (15%), and test (15%) sets. Training involved iteratively updating the model weights using backpropagation. Early stopping was employed to prevent overfitting, based on the validation loss.

The choice of Inception V4 was motivated by its proven efficacy and efficiency in image classification and object detection tasks. Transfer learning was employed to leverage pre-trained weights, significantly reducing training time and enhancing performance. Data augmentation was used to address the variability in real-world driving conditions, thereby enhancing the model's generalization capabilities.

3. EXPERIMENT RESULTS

3.1. Performance metrics

In order to evaluate the proposed approach, we gathered six video sequences of real traffic scenes in different environments with a camera mounted on the rear-view mirror of the hosting car. Three videos were collected in daytime and the remaining three in night-time. The experiments were performed using C++, OpenCV and TensorFlow on a 1.2 GHz dual-core hardware processor system ARM Cortex-A9 (HPS) operating under an LXDE desktop with 1.0 GB DDR3 memory. This hardware processor system is housed within a VEEK-MT2S which includes the MTLC2 module and the DE10 standard FPGA supplied by TERASIC.

In order to evaluate the performance of the proposed system, we used precision and recall as the primary metrics. Precision measures the ratio of true positive detections to the total detections, while recall measures the ratio of true positive detections to the total actual positives. These measures were computed as shown (1) and (2):

$$Precision = \frac{TPs}{TPs+FPs} \tag{1}$$

$$Recall = \frac{TPs}{TPs+FNs} \tag{2}$$

with FNs is false negatives, FPs is false positives and TPs is true positives.
These metrics were calculated for both daytime and night-time datasets, providing a comprehensive evaluation of the model's performance.

3.2. Evaluation results

Our experiments demonstrated that the proposed system significantly improves vehicle detection precision under varying lighting conditions. For instance, as shown in Tables 1 and 2, the system achieved an precision rate of up to 99.2% during daytime and up to 97.57% at night. These results indicate that the combination of normalized cross-correlation, 2D-DWT, and CNN provides robust performance across different scenarios. Where some of its results are shown in Figures 7 and 8.

Table 1. Evaluation results of the three vehicle detection methods during day

Videos	1		2		3	
Methods	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)
Ruan *et al.* [5]	95.11	94.86	96.23	96.03	96	95.89
Wei *et al.* [6]	97.58	97.27	97.97	98.01	97.68	97.11
Our proposed work	98.86	98.93	99.2	98.75	99.05	98.82

Table 2. Evaluation results of the three vehicle detection methods during night

Videos	1		2		3	
Methods	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)
Kuang *et al.* [12]	94.3	94.05	94.83	94.17	95.11	94.06
Hemmati *et al.* [13]	96.81	95.9	96.32	95.93	95.75	96.1
Our proposed work	97.57	97.01	97	96.86	96.19	96.5

To properly evaluate the system that we have proposed, we used four different works from the literature to evaluate the proposed system against. We used two works to evaluate the day-time detection method. The first work, presented by Ruan *et al.* [5], introduces a vehicle detection system targeting vehicle wheel identification, comprising feature extraction and vehicle detection stages. It incorporates ROI segmentation, wheel identification, and vehicle detection. A key drawback is its dependence on wheel visibility, prone to obstruction or occlusion in real-world scenarios, impairing accuracy. Furthermore, the method may struggle with complex backgrounds and varying lighting conditions during the day. The second work, by Wei *et al.* [6], proposes a vehicle recognition method using Haar and HOG features for ROI segmentation and AdaBoost/SVM classifiers for classification. The performance of this method suffers in real-world settings due to diverse vehicle shapes, sizes, and colors. Combining multiple features and classifiers also hinders real-time implementation in DAS.

The comparison between the proposed method and those by Ruan *et al.* [5] and Wei *et al.* [6], as detailed in Table 1, illustrates significant improvements in both precision and recall. As mentioned earlier, our proposed method achieves precision of up to 99.2% and recall of 98.93%. In contrast, Ruan *et al.* [5] achieve a precision of up to 96.23% and a recall of up to 96.03%, while Wei *et al.* [6] achieve a precision of up to 97.97% and a recall of up to 98.01%. Although Wei *et al.* [6] method performs better than Ruan *et al.* [5] both fall short of the precision and recall achieved by our proposed method. These higher values demonstrate the superior ability of our method to accurately detect and classify vehicles during daytime, underscoring its robustness and effectiveness across various daytime scenarios.

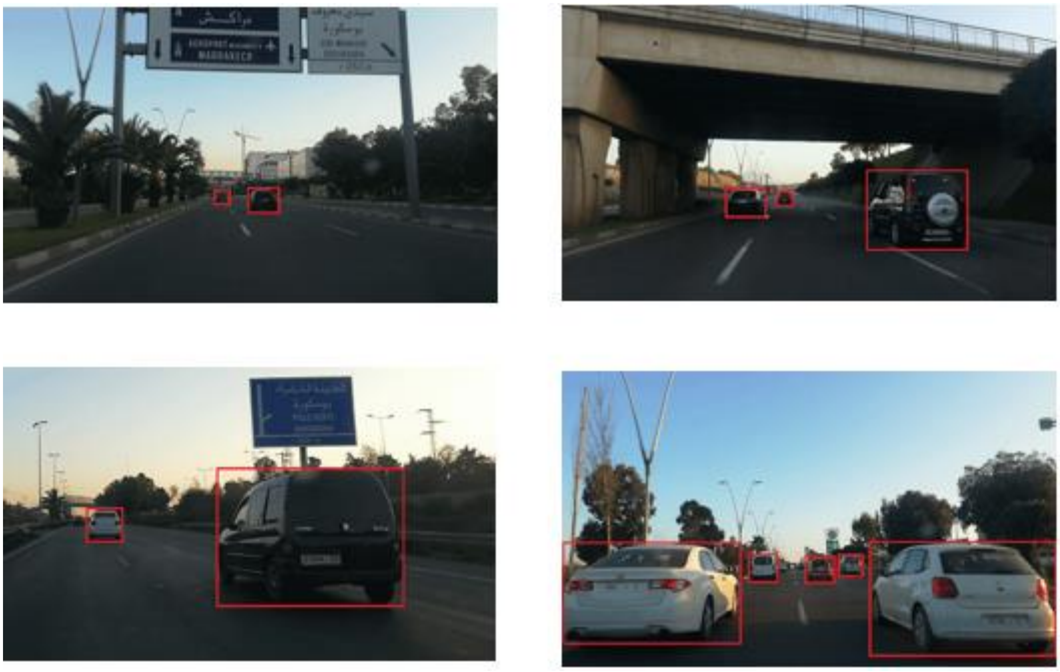

Figure 7. Some results of day-time vehicle

Figure 8. Some results of night-time vehicle detection

To evaluate the night-time detection method, we used the other two works. The first work, by Kuang *et al.* [12], presentes a night-time vehicle detection system integrating ROI extraction with image enhancement techniques to improve visibility under low-light conditions. However, its efficacy relies heavily on preprocessing quality, potentially leading to false positives or missed detections. Adaptability to varied night-time conditions and camera sensors is also limited by its reliance on specific enhancement techniques. The second work, by Hemmati *et al.* [13], proposes an adaptive system for real-time autonomous driving, focusing on detecting vehicle headlights at night. This approach identifies bright regions as headlight, effective for vehicles with visible headlights but less so for others. It may produce false positives from non-vehicle bright objects like streetlights, impacting detection accuracy.

The results for night-time vehicle detection, presented in Table 2, also highlight the superiority of the proposed method compared to the works by Kuang *et al.* [12] and Hemmati *et al.* [13]. The proposed method achieves precision of up to 97.57% and recall of up to 97.01% across the three videos. In comparison, Kuang *et al.* [12] method shows lower performance with precision of up to 95.11% and recall of up to 94.17%. Hemmati *et al.* [13] approach performs better than Kuang *et al.* [12] but still lags behind the proposed method, with precision of up to 96.81% and recall of up to 96.1%. The proposed method's higher precision and recall rates reflect its enhanced capability to accurately detect vehicles under low-light conditions, thus offering more reliable performance in night-time scenarios and demonstrating significant advancements over existing methods.

Our purpose is to design a system that not only has a better successful detection rate, but it should also be as fast as possible. Figure 9 shows the speed of our system in the three experiments. Where the CNN-based daytime vehicle detection system could process up to 21.64 frames per second, at the same time the average frames per second processed for the three videos is 20.72 frames per second. In addition, the night-time vehicle detection system could process up to 23.09 frames per second, while 22.65 frames per second is the average frames per second processed for the three videos. These results are sufficient for real-time processing.

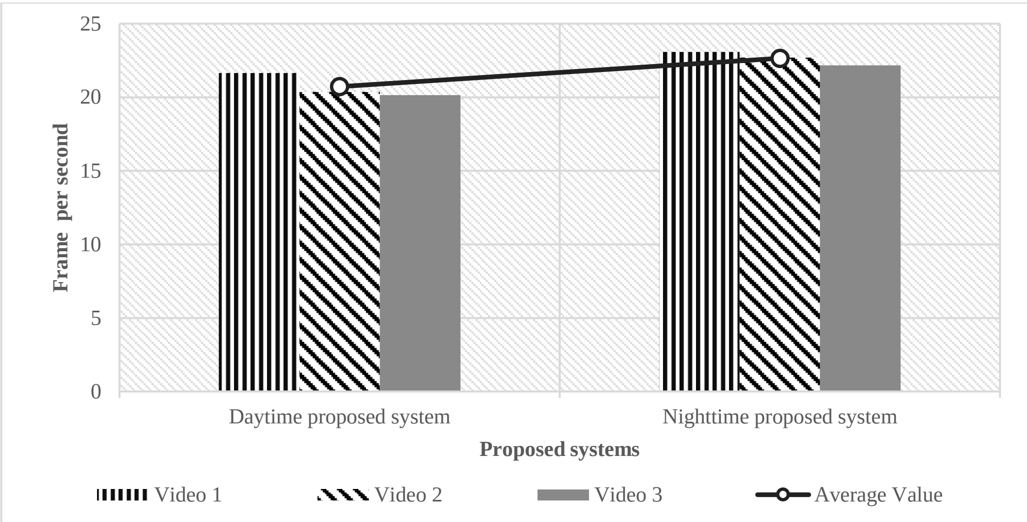

Figure 9. Speed of our system in the three experiments

One of the main strengths of our method is its adaptability to different lighting conditions without requiring additional hardware, such as lasers. This is a significant improvement over methods that depend on laser and video information [11]. However, our system's performance slightly drops in extremely low-light conditions, which we plan to address in future work. Unexpectedly, the system performed better in detecting smaller vehicles at night compared to larger ones, which we attribute to the specific training data used.

4. CONCLUSION

This paper proposes an automated vehicle detection system for driver assistance systems (DAS), where vehicles are detected in real time during day and night. The proposed system first processes the upper part of the input videos at each period to detect whether the videos were recorded during day or night. Then, if day-time is detected, we perform the day-time vehicle detection method, which consists of two major parts: the hypothesis generation part, and the hypothesis verification part. At the first part, the hypotheses are generated by detecting the vehicle patterns using a normalized cross-correlation after converting the input images to HSV color space, performing histogram equalization and running the Canny edge detector. At the second part, we verify the hypotheses generated and classify them as vehicles or non-vehicles using the CNN classifier after extracting their features with the third level of 2D-DWT. And if night-time is detected, we apply the night-time vehicle detection method, which is divided into two main stages: the generation of lamp candidates and the verification lamp candidates. At the first stage, we generate the candidates by first performing the color threshold technique on input images in HSV color space to extract the red and white bright regions (rear-lamp candidates and headlamp candidates), then we extract each bright region as a component using the connected component technique after performing some morphological functions. At the second stage, we verify the generated candidates by first pairing them into sets. Then, we keep only symmetrically identical pairs using the normalized cross-correlation and classify them as vehicles or non-vehicles using the CNN classifier. Experiments have shown that the proposed system works well and detects vehicles with a high accuracy. In addition, a comparative study was carried out between the proposed system and other works in the literature in order to obtain a good evaluation. The obtained results prove the high accuracy of the proposed method. The implications of these findings suggest that our method can be integrated into various driving assistance systems, improving road safety and reducing accident rates. Future research will focus on refining the system's performance in low-light conditions and expanding the dataset to include a wider variety of vehicle types and environmental scenarios.

REFERENCES

[1] I. Slimani, A. Zaarane, W. A. Okaishi, I. Atouf, and A. Hamdoun, “An automated license plate detection and recognition system based on wavelet decomposition and CNN,” *Array,* vol. 8, p. 100040, Dec. 2020, doi: 10.1016/J.ARRAY.2020.100040.

[2] A. Zaarane, I. Slimani, W. A. Okaishi, I. Atouf, and A. Hamdoun, “Distance measurement system for autonomous vehicles using stereo camera,” *Array,* vol. 5, p. 100016, Mar. 2020, doi: 10.1016/J.ARRAY.2020.100016.

[3] A. Zaarane, I. Slimani, A. Hamdoun, and I. Atouf, “Vehicle to vehicle distance measurement for self-driving systems,” 2019 6th *International Conference on Control, Decision and Information Technologies, CoDIT* 2019, pp. 1587–1591, Apr. 2019, doi: 10.1109/CODIT.2019.8820572.

[4] C. Meng, H. Bao, and Y. Ma, “Vehicle detection: a review,” *Journal of Physics: Conference Series,* vol. 1634, no. 1, p. 012107, Sep. 2020, doi: 10.1088/1742-6596/1634/1/012107.

[5] Y. S. Ruan, I. C. Chang, and H. Y. Yeh, “Vehicle detection based on wheel part detection,” *2017 IEEE International Conference on Consumer Electronics - Taiwan, ICCE-TW 2017*, pp. 187–188, 2017, doi: 10.1109/ICCE-China.2017.7991058.

[6] Y. Wei, Q. Tian, J. Guo, W. Huang, and J. Cao, “Multi-vehicle detection algorithm through combining Harr and HOG features,” *Mathematics and Computers in Simulation,* vol. 155, pp. 130–145, Jan. 2019, doi: 10.1016/J.MATCOM.2017.12.011.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “(YOLO) you only look once,” *In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition,* pp. 779–788, Dec. 2016, doi: 10.1109/CVPR.2016.91.

[8] Y. Chen and Z. Li, “An effective approach of vehicle detection using deep learning,” *Computational Intelligence and Neuroscience,* vol. 2022, no. 1, p. 2019257, Jan. 2022, doi: 10.1155/2022/2019257.

[9] S. Karl, Y. Donzia, Y.-P. Geum, and H.-K. Kim, “A study on autonomous driving adaptive simulation system using deep learning model YOLOV3,” *Covenant Journal of Informatics and Communication Technology*, vol. 9, no. 2, pp. 1–15, Dec. 2021, Accessed: Aug. 07, 2024. [Online]. Available: https://journals.covenantuniversity.edu.ng/index.php/cjict/article/view/2794

[10] N. Arora and Y. Kumar, “Automatic vehicle detection system in day and night mode: challenges, applications and panoramic review,” *Evolutionary Intelligence,* vol. 16, no. 4, pp. 1077–1095, Aug. 2023, doi: 10.1007/S12065-022-00723-0.

[11] R. H. Zhang, F. You, F. Chen, and W. Q. He, “Vehicle detection method for intelligent vehicle at night time based on video and laser information,” *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 32, no. 4, Apr. 2018, doi: 10.1142/S021800141850009X.

[12] H. Kuang, L. Chen, F. Gu, J. Chen, L. Chan, and H. Yan, “Combining region-of-interest extraction and image enhancement for nighttime vehicle detection,” *IEEE Intelligent Systems*, vol. 31, no. 3, pp. 57–65, 2016, doi: 10.1109/MIS.2016.17.

[13] M. Hemmati, M. Biglari-Abhari, and S. Niar, “Adaptive vehicle detection for real-time autonomous driving system,” *Proceedings of the 2019 Design, Automation and Test in Europe Conference and Exhibition, DATE 2019,* pp. 1034–1039, May 2019, doi: 10.23919/DATE.2019.8714818.

[14] S. Huang *et al.,* “M-YOLO: a nighttime vehicle detection method combining mobilenet v2 and YOLO v3,” *In Journal of Physics: Conference Series,* vol. 1883, no. 1, p. 012094, 2021, doi: 10.1088/1742-6596/1883/1/012094.

[15] Z. Yang and L. S. C. Pun-Cheng, “Vehicle detection in intelligent transportation systems and its applications under varying environments: a review,” *Image and Vision Computing*, vol. 69, pp. 143–154, Jan. 2018, doi: 10.1016/J.IMAVIS.2017.09.008.

[16] J. Canny, “A computational approach to edge detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. PAMI-8, no. 6, pp. 679–698, 1986, doi: 10.1109/TPAMI.1986.4767851.

[17] S. D. Wei and S. H. Lai, “Fast template matching based on normalized cross correlation with adaptive multilevel winner update,” *IEEE Transactions on Image Processing,* vol. 17, no. 11, pp. 2227–2235, 2008, doi: 10.1109/TIP.2008.2004615.

[18] I. Slimani, A. Zaarane, and A. Hamdoun, “Convolution algorithm for implementing 2D discrete wavelet transform on the FPGA,” *Proceedings of IEEE/ACS International Conference on Computer Systems and Applications,* AICCSA, vol. 0, Jul. 2016, doi: 10.1109/AICCSA.2016.7945831.

[19] K. Nogueira, O. A. B. Penatti, and J. A. dos Santos, “Towards better exploiting convolutional neural networks for remote sensing scene classification,” *Pattern Recognit,* vol. 61, pp. 539–556, Jan. 2017, doi: 10.1016/J.PATCOG.2016.07.001.

[20] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, Inception-ResNet and the impact of residual connections on learning,” *Proceedings of the AAAI Conference on Artificial Intelligence,* vol. 31, no. 1, pp. 4278–4284, Feb. 2017, doi: 10.1609/AAAI.V31I1.11231.

[21] R. C. Gonzales and P. Wintz, *Digital image processing*, 2nd ed. Addison-Wesley Longman Publishing Co., 1987. Accessed: Aug. 07, 2024. [Online]. Available: https://dl.acm.org/doi/abs/10.5555/22881.

[22] “Caltech cars dataset,” [Online]. Available: http://www.vision.caltech.edu/archive.html

[23] G. Hsu, J. Chen, Y. C.-I. transactions on vehicular, and undefined 2012, “Application-oriented license plate recognition,” *IEEE transactions on vehicular technology,* vol. 62, no. 2, 2013, doi: 10.1109/TVT.2012.2226218.

[24] A. Geiger, P. Lenz, and R. Urtasun, "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*, 2012, pp. 3354–3361, doi: 10.1109/CVPR.2012.6248074.

[25] J. Currie, B. Penn, and D. Barnes, "Nighttime vehicle detection dataset." github, 2018, [Online]. Available: https://github.com/ntnu-arl/vehicles-nighttime.

BIOGRAPHIES OF AUTHORS

Abdelmoghit Zaarane is a distinguished researcher and educator specializing in embedded systems and computer vision. He holds the position of assistant Professor and Researcher at Hassan II University of Casablanca, Faculty of Sciences Ben M'sick. He has published extensively in renowned journals and conferences, making significant contributions to fields such as advanced driving assistant systems, traffic engineering computing, and image segmentation. Prof. Zaarane's dedication to advancing technology and educating future researchers underscores his prominent role in the academic community. He can be contacted at email: z.abdelmoghit@gmail.com.

Ibtissam Slimani is a Moroccan Ph.D. Her researches focus on traffic monitoring and autonomous vehicles. She is part of a research group at Hassan II University in Casablanca, Morocco, where she and her collaborators have developed many algorithms in the transportation sector. She is a temporary professor and member of the Robotics and Internet of Things Club, at the Ben M'Sick Faculty of Sciences, Hassan II University of Casablanca, Morocco. She is also a member of the review board of the International Journal of Electrical and Computer Engineering (Hindawi) and the International Journal of Artificial Intelligence (IJ-AI). She can be contacted at email: ibtissamslimani7@gmail.com.

Mourad Elhabchi is an accomplished assistant professor in the Faculty of Sciences Ben M'sick Casablanca, specializing in Electronic Engineering, Telecommunications Engineering, and Communication Engineering. With a wealth of knowledge and expertise in these fields, he has made significant contributions to the academic and research community he published more than 14 papers. Their research and teaching focus on cutting-edge technologies, with a particular emphasis on antennas. He possesses a deep understanding of antenna design, optimization, and performance evaluation, making them a sought-after expert in this area. He can be contacted at email: mouradelhabchi@gmail.com.

Issam Atouf has been a professor at Casablanca's Ben M'Sick Faculty of Sciences since 2004, specializing in Electronics; Embedded Systems Electronics; Information Processing; Artificial Neural Networks. He has supervised several research projects as part of his theses or final year projects for Master's students. His work has been the subject of some twenty articles published in specialized journals. He can be contacted at email: issamatouf@gmail.com.