Transforming Traffic Safety: Detection of Car-Pedestrian Contact Using Computer Vision Technologies

Vesna Radojcic[1], Milos Dobrojevic[2]
[1]*University Sinergija, Faculty of Computing and Informatics, Bijeljina, Bosnia and Herzegovina, vradojcic@sinergija.edu.ba, 0000-0001-7826-1081*
[2]*University Sinergija, Faculty of Computing and Informatics, Bijeljina, Bosnia and Herzegovina, mdobrojevic@sinergija.edu.ba, 0000-0003-3798-312X*

Original scientific paper
https://doi.org/10.7251/JIT2402104R UDC: 656.1.08:625.712.34

Abstract: This paper explores the integration of computer vision technologies to enhance traffic safety through the effective detection of car-pedestrian interactions. As urban environments become more congested, pedestrian safety remains a critical concern. The system's performance was evaluated using real-life footage from vehicle-mounted cameras, as well as images and videos sourced from online platforms. These real-world scenarios enabled a detailed assessment of the system's accuracy and efficiency in practical conditions. The study highlights the potential for significant improvements in traffic safety, particularly in Bosnia and Herzegovina, where over 38% of registered vehicles are older than 23 years, and nearly 62% exceed 14 years. The aging vehicle fleet heightens the risk of accidents, underscoring the need for advanced detection methods. The proposed system automates the identification of hazardous situations on roads, allowing timely responses from relevant authorities.

Keywords: Traffic safety, YOLOv8, Computer vision, OpenCV, Object detection

Introduction

Computer vision technologies have demonstrated considerable potential in improving traffic safety, especially in detecting and analyzing interactions between pedestrians and vehicles. These systems facilitate automated identification of traffic conflicts and violations, delivering crucial data for safety assessments and the development of preventive measures [1]. Within cities, traffic congestion has emerged as the primary challenge. [2]. Monitoring pedestrian movements and their interactions with vehicles, commonly referred to as object tracking [3], can significantly enhance pedestrian safety. By analyzing pedestrian behavior, recognizing traffic patterns, and predicting potential conflicts, authorities can implement various measures, such as pedestrian crossing signals, improved signage, and traffic calming techniques, to boost overall pedestrian safety. According to the latest statistics on traffic accidents, their causes, and consequences in Bosnia and Herzegovina, 31,321 traffic accidents were reported on Bosnian-Herzegovinian roads in 2022. Of these, 7,230 resulted in casualties and injuries, while 24,091 cases involved material damage. The total number of accidents marked an increase of 1,083 compared to 2021, highlighting a persistent upward trend in traffic incidents each year. [4]. Given these alarming statistics regarding traffic incidents in Bosnia and Herzegovina, this research aims to provide practical solutions for enhancing traffic safety.

Literature Review

Recent advancements in computer vision technologies have significantly improved traffic safety and surveillance systems. These systems utilize object detection and recognition techniques to identify vehicles, pedestrians, and traffic signs. [5]. Despite challenges like varying weather conditions and lighting changes, these technologies continue to evolve, contributing to the development of intelligent transportation systems and autonomous vehicles.

Patil et al. [6] propose a real-time traffic sign detection system utilizing deep learning models, which improves road safety by recognizing traffic signs and aiding traffic management. Similarly, S. S et al. [7] introduce an IoT-based system that employs YOLO for real-time traffic monitoring and accident detection, optimizing traffic signal timings to alleviate congestion. Chadha et al. [8] focus on driver safety by developing a fatigue detection system using MediaPipe and OpenCV, which monitors driver attention and alerts them to prevent accidents. Kushwaha et al. [9] address lane detection challenges, proposing a system that identifies lane boundaries under varying conditions, thereby enhancing driver awareness and reducing accidents.

Rocky et al. [10] have tackled the necessity for a dependable system to detect high-risk incidents. They highlight the significance of enabling self-driving vehicles to function autonomously for prolonged durations without human oversight. Their review centers on utilizing dashboard cameras (dashcams) as a cost-effective means to improve the safety of autonomous vehicles during accident scenarios. The authors provide a thorough overview of the evolution of concepts in this field, classifying methods into supervised, self-supervised, and unsupervised learning. They meticulously analyze evaluation criteria and available datasets, shedding light on the advantages and drawbacks of various approaches.

These studies illustrate the transformative potential of computer vision in improving road safety and traffic management.

Methods and materials

The preparation for this experimental study involved collecting various video materials and photographs obtained from real-life situations as well as online sources, ensuring an authentic representation of traffic scenarios (Figure 1). The dataset used in the research includes 150 images and 50 video clips, as well as additional data from the NYC Motor Vehicle Collisions to Person dataset available on Kaggle [11]. This additional dataset enhanced the analysis, providing a broader spectrum of traffic scenarios and increasing the validity of the findings. The goal was to provide deeper insights into the system's detection efficiency and assess its applicability in real-world conditions.

Image Processing

Image processing encompasses the manipulation of digital data in the form of images and videos, allowing for the extraction of information regarding shape, position, orientation, size, dimensions, and color [12]. When individuals view images or videos, they can easily identify objects, gauge distances between them, and discern colors and positions. The main goal of digital image processing is to extract and interpret this information. Our image processing workflow consists of several steps, including loading, resizing, edge detection, object detection, converting to grayscale, and removing noise.

Loading an image

When loading an image, the initial step is to import the necessary library. Following this, the cv2.imread() function is invoked to load the image, and the resulting data is stored in the variable img. This function's parameters include the file path of the image and an integer that specifies whether the image should be loaded in color or grayscale (stored in the variable img_1). To display the loaded image in a new window, the cv2.imshow() function is called, which takes as parameters the name of the window and the variable containing the image data. Next, the waitKey() function is executed, which temporarily halts program execution until a key is pressed. Once a key is detected, the destroyAllWindows() function is called to close all open windows.

Figure 1. *Image loaded in its original format*

Resizing the image

To resize an image by its width and height in pixels, the resize() function from the OpenCV library (cv2) is employed. The image dimensions can be adjusted according to specific requirements. In this example, the image size is reduced by 50%, and the new dimensions are calculated accordingly. If only one dimension is known, the other can be determined based on the aspect ratio of the original image.

Here's a breakdown of the process [13]: The cv2.imread() function reads the specified file in cv2.IMREAD_UNCHANGED mode, returning a NumPy array containing the pixel values. The variable scaling_percentage is set to 50, indicating that the image will be reduced to 50% of its original dimensions (both width and height). img.shape[1] retrieves the width of the original image. int(img.shape[1] * scaling_percentage / 100) computes the new width, which represents 50% of the original width (Figure 2). The new size of the image is established with these calculated dimensions. The cv2.resize() function resizes the image img to the new dimensions stored in the variable new_image, returning a NumPy array. Finally, the resized image is saved to disk with the name "traffic_resize.jpg".

Figure 2. *Size comparison: 50% reduction (right) compared to the original size (left)*

Converting an RGB image to grayscale

To create a grayscale image, you can either read the file directly in grayscale mode or if an RGB image is already loaded, convert it to grayscale using the cvtColor method from the OpenCV library (Figure 3).

Figure 3. *Original RGB image (left) converted to grayscale (right)*

Noise removal

One of the key challenges in image processing and computer vision is the removal of noise from images. The "denoising" process entails estimating the original image by minimizing the noise that may be present (Figure 4). Noise can originate from various sources, including sensors and environmental factors, making it often unavoidable in real-world applications.

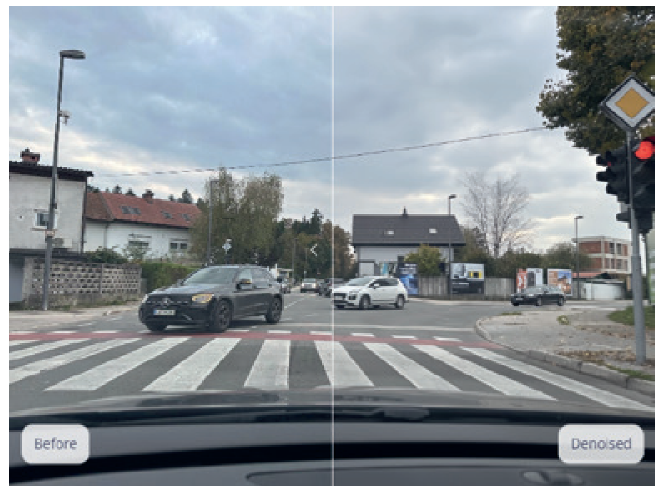

Figure 4. *Result of noise removal from the image*

Edge detection

Edge detection is a critical and essential function in computer vision and image processing, with numerous applications [14]. Its primary objective is to identify significant variations in grayscale images and to understand the physical phenomena that cause these changes.

This process involves locating the boundaries or edges of objects by detecting sudden shifts in shading within the image [15]. This technique is instru-

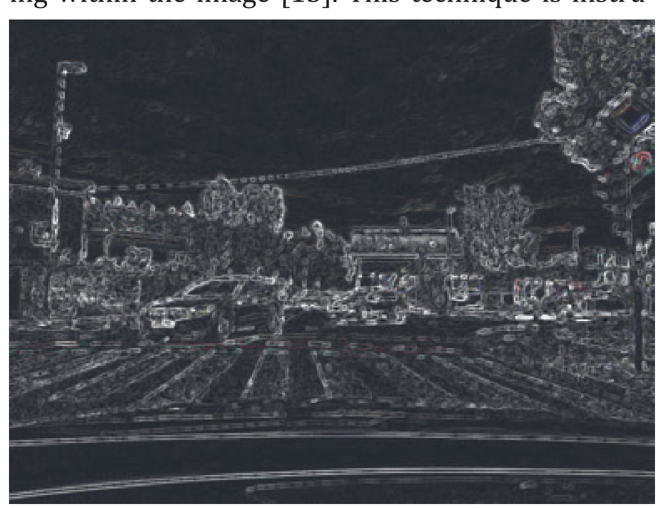

Figure 5. *Result of edge detection on the image*

mental in extracting structural information about the objects represented in the image. Various algorithms are available for edge detection, given their broad applicability, with one notable example being the Canny Edge Detection algorithm (Figure 5).

Characteristics of vehicle-pedestrian collisions

Vehicle-pedestrian collision contacts occur at various points depending on factors such as the speed of the vehicle, the height of the pedestrian, and the type of vehicle involved. These contact points typically include the front bumper, hood, and windshield of the vehicle, which often result in injuries to the lower limbs, torso, and head of the pedestrian. The dynamics of such collisions are crucial for understanding injury mechanisms and improving vehicle safety design. Figure 6 illustrates the typical contact positions during these types of collisions.

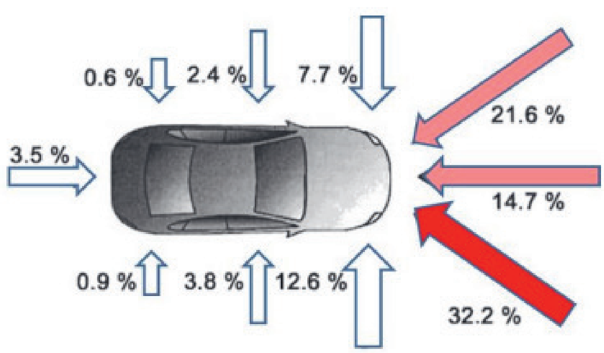

Figure 6. *Positions of vehicle-pedestrian collision contacts [16]*

Results

Global Workflow Architecture for Object Detection and Collision Analysis

A systematic workflow was designed to implement object detection and collision analysis using advanced computer vision techniques (Figure 7). The workflow consists of four main stages:

Preprocessing, Object Detection, Collision Detection, and Postprocessing and Validation. Each stage plays a crucial role in ensuring the robustness and accuracy of the system.

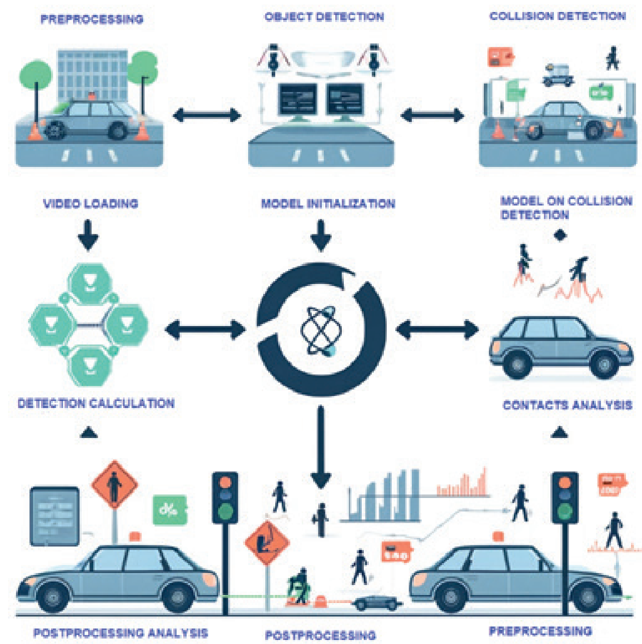

Figure 7. *Global Workflow Architecture for Object Detection and Collision Analysis*

Object detection

Object detection, a crucial task in computer vision, has undergone significant advancements with the emergence of deep learning techniques. This technology enables machines to analyze and identify thousands of individual objects in real-time as they move past a sensor. [17].

This technology finds broad application in areas such as surveillance, autonomous vehicles, medical imaging, and beyond. By continuously scanning and analyzing multiple objects in real-time, machines can accurately detect and classify them, unlocking vast opportunities for automation and intelligent decision-making. Object detection leverages computer vision and image analysis to identify specific elements within images and video content. Using the YOLOv8 algorithm for object detection, various objects in the image were successfully identified (Figure 8). The detected objects included individuals, cars, and traffic lights, all of which were identified with high precision and speed.

For this detection, the command yolo predict source=viber_image_traffic.jpg model=yolov8n.pt was used.

Figure 8. *Object Detection using YOLOv8 Algorithm*

Pedestrian-Vehicle Contact Detection

The development of the "Pedestrian-Vehicle Contact Detection" application in Python significantly improves pedestrian safety in traffic. The application aims to assist in accident prevention by using advanced computer vision techniques, specifically YOLO (You Only Look Once) version 8, to detect potential collisions between vehicles and pedestrians in real-time.

In the initial phase of the code, the necessary libraries for video processing are imported. These libraries include OpenCV for handling video files, Torch for deep learning model operations, and NumPy for numerical data manipulation. A function called load_yolov8_model() is defined to load the YOLOv8 model for object detection. This function utilizes the torch.hub.load method to download the model from the popular Ultralytics/YOLOv8 repository. If the model fails to load, an appropriate error message is displayed.

The path to the video file, defined as video_file_path, specifies the input for processing. The main video processing function is responsible for detecting objects and potential collisions. It starts by loading the YOLOv8 model, followed by opening the specified video file for processing and initializing variables to track object and collision data. In the main loop, each frame of the video is read and preprocessed before being passed to the detection model. Detected objects are classified into categories (vehicles and pedestrians), and the average size of vehicle bounding boxes is calculated. Bounding boxes are then drawn on the frame to visualize detected objects.

Afterward, collisions between objects are detected, and collision data is overlaid on the frame for enhanced visualization. The processed video is displayed, and the loop continues until the user decides to exit, typically by pressing 'q' on the keyboard.

$$Detection\ Ratio = \frac{Detected\ accident\ cases}{Total\ accident\ cases\ in\ the\ dataset} \times 100 \quad (1)$$

$$False\ Alarm\ Rate = \frac{Patterns\ where\ false\ alarm\ occurs}{Total\ number\ of\ patterns} \times 100 \quad (2)$$

Equations (1) and (2) represent the detection rate and false alarm rate, respectively [18]. This application framework achieved a detection rate of 79%, calculated using Equation (1), and a false alarm rate of 10.3%, based on Equation (2). These results demonstrate the system's solid performance, considering multiple factors contributing to collision detection.

DISCUSSION

Although the system has achieved a high detection rate, certain challenges have been identified. In conditions of heavy traffic, the system occasionally fails to detect certain vehicles, resulting in a higher number of false positives. In situations where detected objects in the image partially overlap in daylight, the objects are identified, and their relative positions indicate contact. However, during the processing of footage recorded at night, when street lighting is minimal, various issues may arise.

In such conditions, the following can occur:

- Non-detection of objects (vehicles or pedestrians).
- Inaccurate detection of objects.
- Misclassification of objects.

For example, if a vehicle and a pedestrian are in an initial position of mutual contact, the system may assign the wrong class to the vehicle, recognizing it as a pedestrian, which leads to a lack of contact detection.

To improve detection accuracy in these challenging conditions, spatial depth compensation could be implemented [19], which, by comparing the relative positions of vehicles and pedestrians, may help reduce classification and detection errors. This strategy could potentially optimize the system's performance in various weather and lighting conditions, ensuring more reliable collision detection.

Conclusion

By exploring the application of computer vision and utilizing OpenCV and YOLOv8 for detecting traffic accidents, we have opened the door to innovative solutions that can significantly improve traffic safety. Early detection of hazardous situations and enabling quick responses from relevant authorities can drastically reduce traffic accidents and save lives. The combination of real-world scenarios and online video materials allowed for extensive testing of the system in realistic conditions, providing valuable insights into the effectiveness of various detection methods.

However, certain limitations of the research must be acknowledged. The dataset used for testing was relatively small. The system's performance in low-light conditions or heavy traffic remains an area for improvement.

Further research should focus on expanding the dataset to include a wider range of traffic and weather conditions. This initiative not only enhances road safety in Bosnia and Herzegovina but also lays the foundation for further technological advancements in the field of traffic safety.

References

[1] M. Zaki, S. T., T. Ahmed, and H. Mohamed, "Application of Computer Vision to Diagnosis of Pedestrian Safety Issues," *Transportation Research Record: Journal of the Transportation Research Board,* vol. 2393, no. 1, pp. https://doi.org/10.3141/2393-09, 2013.

[2] J. W. R. Salem, K. Rajkumar, A. M., and A. Parameswari, "Efficient Intelligent Smart Ambulance Transportation System using Internet of Things," *Technical gazette,* vol. 1, no. https://doi.org/10.17559/TV-20230726000829, pp. 171-177, 2024.

[3] M. N. Favorskaya and V. V. Andreev, "THE STUDY OF ACTIVATION FUNCTIONS IN DEEP LEARNING FOR PEDESTRIAN DETECTION AND TRACKING," *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-2/W12,* pp. 53–59 doi:10.5194/isprs-archives-XLII-2-W12-53-2019, 2019.

[4] S. Z. I. I. D. (SID), "BIHAMK - BOSANSKOHERCEGOVAČKI AUTO-MOTO KLUB," March 2023. [Online]. Available: https://bihamk.ba/statistika/statistike. [Accessed 2024].

[5] S. Gaurav and S. S. Prof., "Advanced Systems for Detecting and Recognizing Traffic Objects," *International Journal of Scientific Research in Computer Science Engineering and Information Technology,* p. https://doi.org/10.32628/cseit24104111, 2024.

[6] P. Dr. Rahul, A. Dr. Prashant, B. Dr. Kalyan, P. Dr. Abhijit, D. P. Dr. Pramod, B. Saomya, D. Resham, B. Gautam and S. D. Bikramjeet, "Real-Time Traffic Sign Detection and Recognition System using Computer Vision and Machine Learning," *J. Electrical Systems,* pp. 2244-2254 https://doi.org/10.52783/jes.1991, 2024.

[7] S. S. S., H. S and B. M. T, "IoT Based Traffic Congestion Management and Accident Detection System," *Journal of Electrical Engineering and Automation,* p. 10.36548/jeea.2024.1.005, 2024.

[8] S. C. R., Jugesh and S. Jasmehar, "Enhancing Road Safety: A Driver Fatigue Detection and Behaviour Monitoring System using Advanced Computer Vision Techniques," *Journal of Ubiquitous Computing and Communication Technologies,* p. https://doi.org/10.36548/jucct.2024.2.004, 2024.

[9] K. K. Gopal, K. Chandrika, G. Kapil and H. Khushi, "Review on Computer Vision Techniques to enhance Road Lane Detection by using Open CV," *International Journal of Innovative Research in Advanced Engineering,* p. https://doi.org/10.26562/ijirae.2024.v1102.07, 2024.

[10] R. Arash, J. W. Qingming, and Z. Wandong, "Review of Accident Detection Methods Using Dashcam Videos for Autonomous Driving Vehicles," *IEEE Transactions on Intelligent Transportation Systems,* p. doi: 10.1109/TITS.2024.3354852, 2024.

[11] KAKUROO3, "NYC Motor Vehicle Collisions to Person," kaggle, 2021. [Online]. Available: https://www.kaggle.com/datasets/kukuroo3/nyc-motor-vehicle-collisions-to-person.

[12] B. Wilhelm and J. B. Mark, Digital Image Processing An Algorithmic Introduction, Springer International Publishing, 2022.

[13] V. Radojcic, "UPOTREBA RAČUNARSKOG VIDA ZA PREPOZNAVANJE PJEŠAKA U SAOBRAĆAJU," *Master thesis, Univerzitet Sinergija,* 2021.

[14] X. Ronghao, X. Xin, P. Hong, W. Jun, R. d. A. M. Antonio, and Y. Qian, "Feature fusion method based on spiking neural convolutional network for edge detection," *Pattern Recognition,* vol. 147, no. https://doi.org/10.1016/j.patcog.2023.110112, p. 110112, 2024.

[15] V. Radojcic, A. S. Cvetkovic, M. Dobrojevic, P. Spalevic, and M. E. Jalal, "Advancements in Computer Vision Applications for Traffic Surveillance Systems," in *Sinergija University Scientific Conference with International Participation*, Bijeljina, 2023.

[16] D. Mikulić, R. Spudić and B. Katana, "ZAŠTITA PJEŠAKA OD NALETA MOTORNIH VOZILA," in *Zbornik radova: 11. Dani kriznog upravljanja 2018 (DKU-2018)*, Brijuni, Hrvatska, 2018.

[17] V. Radojcic, A. S. Cvetkovic and M. Dobrojevic, "The Use of Computer Vision in Precision Agriculture," in *Sinteza 2023 - INTERNATIONAL SCIENTIFIC CONFERENCE ON INFORMATION TECHNOLOGY, COMPUTER SCIENCE, AND DATA SCIENCE*, Belgrade https://doi.org/10.15308/Sinteza-2023-60-65, 2023.

[18] E. P. Ijjina, D. Chand, S. Gupta and K. Goutham, "Computer Vision-based Accident Detection in Traffic Surveillance," in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Kanpur, India pp. 1-6, doi: 10.1109/ICCCNT45670.2019.8944469, 2019.

[19] D. Jokanovic, "Detekcija saobraćajnih nezgoda na pješačkim prelazima pomoću računarskog vida," *Master thesis, Univerzitet Sinergija,* 2023.

[20] H. Narit, C. Pitchaya, M. Nagorn, N. Chayawat, and I. Pawinee, "The video-based safety methodology for pedestrian crosswalk safety measured: The case of Tham-

masat University, Thailand," *Transportation Research Interdisciplinary Perspectives,* p. 101036 https://doi.org/10.1016/j.trip.2024.101036, 2024.

[21] T. Raju, K. Kopparapu, S. R. Borugadda, K. Konka, and V. S. Alapati, "IoT Based Automatic Vehicle Accident Detection and Rescue System," *International Journal for Modern Trends in Science and Technology,* vol. 10, no. 3 https://doi.org/10.46501/IJMTST1003016, pp. 94 - 99, 2024.

[22] V. Radojcic and M. Dobrojevic, "ANALYSIS OF VEHICLE-PEDESTRIAN CONTACT USING COMPUTER VISION AND OPENCV," in *ITeO - Informacione tehnologije za e-obrazovanje M e d j u n a r o d n i n a u c n o - s t r u c n i s k u p*, Banja Luka, 2024.

Received: October 22, 2024
Accepted: November 19, 2024

ABOUT THE AUTHORS

Vesna Radojcic is a Ph.D. student in Computer Systems Security at the University Sinergija, where she works as an assistant at the Faculty of Computing and Informatics. She holds a Master's degree in Contemporary Information Technologies, and her research interests include the application of artificial intelligence and computer vision technologies. She has participated in numerous scientific conferences.

Milos Dobrojevic, Ph.D., is an Associate Professor at the Faculty of Computing and Informatics, University Sinergija, Bijeljina, Republic of Srpska, Bosnia and Herzegovina. He earned his Ph.D. in 2006 from the Faculty of Mechanical Engineering, University of Belgrade, Republic of Serbia. He has published over 50 scientific papers, with a research focus on software engineering, artificial intelligence, and smart IoT applications.

FOR CITATION

Vesna Radojcic, Milos Dobrojevic, Transforming Traffic Safety: Detection of Car-Pedestrian Contact Using Computer Vision Technologies, *JITA – Journal of Information Technology and Applications, Banja Luka*, Pan-Europien University APEIRON, Banja Luka, Republika Srpska, Bosna i Hercegovina, JITA 14(2024)2:104-110, (UDC: 656.1.08:625.712.34), (DOI: 10.7251/JIT2402104R, Volume 14, Number 2, Banja Luka, December (89-188), ISSN 2232-9625 (print), ISSN 2233-0194 (online), UDC 004