

Article

Optimization of User Service Rate with Image Compression in Edge Computing-Based Vehicular Networks

Liujiang Zhang ^{1,2} , Jin Li ^{1,2,*} , Wenyang Guan ^{1,2} and Xiaoqin Lian ^{1,2}

¹ School of Artificial Intelligence, Beijing Technology and Business University, Beijing 100048, China; 2230601029@st.btbu.edu.cn (L.Z.); 20151005@btbu.edu.cn (W.G.); lianxq@th.btbu.edu.cn (X.L.)

² Key Laboratory of Industrial Internet and Big Data, China National Light Industry, Beijing Technology and Business University, Beijing 100048, China

* Correspondence: jinli@btbu.edu.cn

Abstract: The prevalence of intelligent transportation systems in alleviating traffic congestion and reducing the number of traffic accidents has risen in recent years owing to the rapid advancement of information and communication technology (ICT). Nevertheless, the increase in Internet of Vehicles (IoV) users has led to massive data transmission, resulting in significant delays and network instability during vehicle operation due to limited bandwidth resources. This poses serious security risks to the traffic system and endangers the safety of IoV users. To alleviate the computational load on the core network and provide more timely, effective, and secure data services to proximate users, this paper proposes the deployment of edge servers utilizing edge computing technologies. The massive image data of users are processed using an image compression algorithm, revealing a positive correlation between the compression quality factor and the image's spatial occupancy. A performance analysis model for the ADHOC MAC (ADHOC Medium Access Control) protocol is established, elucidating a positive correlation between the frame length and the number of service users, and a negative correlation between the service user rate and the compression quality factor. The optimal service user rate, within the constraints of compression that does not compromise detection accuracy, is determined by using the target detection result as a criterion for effective compression. The simulation results demonstrate that the proposed scheme satisfies the object detection accuracy requirements in the IoV context. It enables the number of successfully connected users to approach the total user count, and increases the service rate by up to 34%, thereby enhancing driving safety, stability, and efficiency.

Keywords: Internet of Vehicles; mobile edge computing; object detection; image compression

MSC: 94A08



Citation: Zhang, L.; Li, J.; Guan, W.; Lian, X. Optimization of User Service Rate with Image Compression in Edge Computing-Based Vehicular Networks. *Mathematics* **2024**, *12*, 558. <https://doi.org/10.3390/math12040558>

Academic Editors: Jonathan Blackledge and Radu Tudor Ionescu

Received: 31 December 2023

Revised: 29 January 2024

Accepted: 11 February 2024

Published: 12 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Recently, the application of intelligent transportation systems has increasingly gained prominence. Consequently, the provision of real-time and secure intelligent services to users with constrained computing resources (including bicyclists, pedestrians, and wheelchair users) has emerged as a significant area of research.

The intelligent connected vehicle (ICV), which encapsulates both Internet of Vehicles (IoV) and autonomous driving technologies, connects with sensor devices such as Road Side Units (RSUs) and edge servers. This enables it to analyze and process information about road conditions and other relevant data for controlling the vehicle's driving behavior [1]. Owing to their capability to effectively navigate complex urban road conditions, intelligent vehicles are increasingly becoming prevalent. Consequently, ensuring safe navigation within the IoV system is gaining paramount importance, with its primary objective being the perception of the surrounding environment and the extraction of critical information [2]. Target detection constitutes a fundamental perception challenge in autonomous driving systems. The literature [3] delves into and contrasts several cutting-edge algorithms in

target detection, including Faster R-CNN, Mask R-CNN, and Yolo v4, thereby underscoring the necessity of efficient and precise target detection algorithms within the IoV context. In autonomous driving systems, the paramount concern of object detection technology is the swift and accurate identification of road vehicles, crucial for enhancing driving safety and efficiency.

Vision provides the most critical information while driving, and cameras serve as sensors closely mimicking the human eye. The widespread availability of high-performance on-board cameras today enables users to effortlessly acquire detailed, high-resolution image data of the real-world environment, rich in information [4]. Owing to the swift advancements in image processing technology, it is now possible to identify specific objects within raw image data using algorithms. For instance, on-board cameras facilitate vehicular navigation by identifying target vehicles ahead. It is important to acknowledge that higher-quality image data not only occupy more space but also lead to increased processing delays and transmission burdens, thereby escalating the security risks associated with autonomous driving applications. As the number of users in the IoV continues to grow, the demands on Intelligent Transportation Systems (ITS) become increasingly stringent. These include the need for rapid and accurate target detection, the alleviation of data growth pressures, the enhancement of user service rates, and ensuring application security. Consequently, this necessitates substantial computational power in the IoV to cater to the expanding user requirements.

Presently, mobile edge computing (MEC), emerging as a novel computing paradigm, has come into focus for its ability to enhance computing efficiency and conserve computational resources. The advancement of 5G and 6G communication technologies has significantly facilitated the Internet of Things (IoT), albeit generating substantial data traffic [5]. As the user base within the IoV expands, there is an escalated demand for advanced object detection algorithms and enhanced security in intelligent transportation systems. To swiftly process the voluminous data necessitated for transmission in the IoV, MEC addresses the stringent demands of 5G and superior communication environments. This is achieved through the deployment of servers endowed with computing and storage capabilities at the network's edge, facilitating data processing proximate to the user and integrating technologies like compression and caching to expedite data flow [6]. Hui et al. [7] examined the average energy efficiency associated with computational offloading in both MEC and non-MEC networked vehicles. The findings reveal that MEC-integrated networked vehicles are capable of optimally selecting task sizes and transmission intervals, thereby maximizing energy efficiency, which is at least tenfold higher compared to non-MEC networked vehicles. However, current research often overlooks further processing of image data required for network transmission, relying instead on direct usage of high-definition images obtained by vehicle units. This approach results in an excessive data load and processing burden on these units. Currently, JPEG image compression technology is recognized as one of the most prevalent image compression methods globally, due to its significant capacity to reduce file sizes [8]. Therefore, this paper integrates MEC for image data processing, simultaneously ensuring accuracy in target detection. This approach aims to enhance data transmission efficiency, expand the user base of vehicle networking services, and consequently mitigate traffic congestion.

Autonomous driving technology, through its perception of the surrounding environment and subsequent data processing, not only facilitates convenience for drivers but also significantly enhances traffic safety [9]. It is evident that precision and rapidity in vehicle detection technology, coupled with a high user service rate, are imperative for driving safety. In this context, within the IoV environment and leveraging edge computing, the present study develops an ADHOC MAC (ADHOC Medium Access Control) protocol access performance analysis model. This model explores the mathematical correlations between frame length N , the number of service users, the compression quality factor, and the user service rate. A Yolov5 vehicle object detection model has been developed to facilitate complete target recognition in all image data, establishing a benchmark for recognition

accuracy. Furthermore, the JPEG compression algorithm is employed for optimization, examining the compression threshold and its associated user service rate in relation to accuracy requirements. Ultimately, the results of the system optimization are simulated and analyzed.

This paper is structured as follows: Section 2 analyzes the current state of research on resource optimization technologies in edge computing for IoV. Section 3 details the establishment of the system architecture and performance analysis model, along with theoretical derivation. Section 4 introduces the object detection and compression module for system optimization. Section 5 presents the final simulation results and evaluates the optimization outcomes. Finally, Section 6 provides a comprehensive summary of the entire paper.

2. Related Works

In this section, we present the current state of research in object detection, edge computing, and image processing technology within the context of the IoV, thereby establishing a foundation for the subsequent research work of this paper.

2.1. Object Detection Technology

In the evolution of intelligent transportation systems, vision-based driver assistance systems have become prevalent within the IoV context [10]. Early object detection techniques primarily concentrated on specific objects characterized by simplistic appearances and minimal variations, such as roads. However, while effective in certain scenarios, these techniques exhibit numerous limitations in practical vehicular applications, including the need for precise templates and rapid, accurate recognition. With the advancement of machine learning, a multitude of object detection algorithms have surfaced, achieving significant advancements. Dai et al. [11] focused primarily on the transmission delays induced by the copious amounts of redundant data within driver assistance systems. An enhanced Haar-like feature classification algorithm was utilized for object detection, and redundant video frames were eliminated to augment the transmission speed. The results demonstrated an approximate 84-fold increase in transmission speed following the filtration of 40% of similar frames. However, this methodology, while substantially enhancing the data transmission rate, compromises the accuracy of object detection. In autonomous driving systems, ensuring safety is paramount, necessitating the assurance of accurate vehicle target detection.

Following advancements in deep learning, the YOLO object detection algorithm has gained widespread adoption since its inception. This is attributed to its faster processing speed coupled with significant improvements in detection accuracy; for instance, YOLO processes approximately 300 times faster than Fast-RCNN while maintaining comparable accuracy [12]. Zineb et al. [13] presented an optimization of the YOLOv4 object detection model, enabling it to swiftly and accurately identify objects even amidst significant interference in road scene image data. A consistent public dataset was employed for testing, facilitating a comparison of the detection accuracies between the Faster-RCNN and EfficientDet models. Ultimately, the enhanced YOLOv4 detection model demonstrated marginally superior detection accuracy compared to the other two models. While the YOLOv4 model exhibits stable detection accuracy, it demands high-end configuration and deployment environments, resulting in limited flexibility. In contrast, YOLOv5s boasts the smallest network size, fastest processing speed, and ease of deployment, with its detection accuracy for large targets being on par with that of YOLOv4. Therefore, this paper selects the lightweight, easily deployable, and rapid YOLOv5s model as the object detection module for vehicle identification in image data.

2.2. Image Compression Technology

High-definition cameras in contemporary intelligent vehicles generate substantial volumes of high-definition image data, resulting in excessive data processing loads in

the cloud and strained computing resources. Consequently, research has intensified in visual data compression technology to mitigate storage and bandwidth resource consumption and enhance data transmission rates, focusing primarily on lossy and lossless compression methods [14]. Compared to lossless compression, lossy compression is less effective at removing redundant image information but offers a more pronounced compression effect, making it more suitable for enhancing data transmission efficiency in the IoV. Pantwa et al. [15] explored a machine learning-based method for visual data compression, employing an autoencoder for the image compression process. The results indicated that this approach preserved essential semantic features for tasks like image classification and detection, with a compression effect surpassing that of the DeepSIC method. While this research achieves bit rate reduction, it employs a simple uniform quantization approach in coding, resulting in the loss of significant information and a compromise in accuracy. Jalilian et al. [16] investigated the deep learning-based lossy compression of iris image data, revealing that, while the model exhibits high compression efficacy, it adversely affects recognition performance. Furthermore, the application of deep learning algorithms increases computational demands, particularly in IoV context, leading to an augmented computational load. Consequently, this paper utilizes the JPEG compression algorithm to process image data, effectively achieving a high compression ratio while ensuring the speed and accuracy of image detection tasks.

2.3. Mobile Edge Computing Technology

MEC-enabled vehicular networks offer advantages such as reduced response times, diversified services, the alleviation of substantial bandwidth pressures caused by big data, and proximity-based storage and services [17]. Consequently, edge computing technology has seen widespread application in the IoV domain in recent years. Executing object detection tasks on edge devices has been shown to enhance resource utilization more effectively. In [18], the authors employed the YOLOv3 model for detecting autonomous vehicles within an edge computing framework, demonstrating the technology's efficacy in conserving computational resources; however, there were shortcomings in detection accuracy. In [19], the authors juxtaposed the traditional autonomous driving environment detection methods with a novel approach, integrating the Collaborative Vehicle Infrastructure System (CVIS) and autonomous driving technology to propose a scalable 5G MEC-driven vehicle infrastructure collaborative system. The findings indicate that the detection accuracy of the fusion scheme exhibits an approximate 10% improvement over single viewpoint perception. Furthermore, collaborative systems facilitate the connection of distributed sensors, thereby enhancing the efficiency of autonomous driving. However, the vehicle detection image data utilized in this study were captured using fixed-angle cameras, without any subsequent processing of the image data. Given the dynamic nature of driving environments and the need for high flexibility, reliance on fixed cameras to capture road conditions could significantly compromise the real-time performance and safety of autonomous driving systems. Xiao et al. [20] introduced STAC, a deep neural network-driven compression scheme designed for edge-assisted semantic video segmentation, and proposed a spatiotemporal adaptive scheme to address the challenges related to varying spatial sensitivities and substantial bandwidth consumption. The results demonstrate that STAC, in comparison to leading-edge algorithms, can conserve up to 20.95% of bandwidth while maintaining accuracy. The adaptive compression strategy outlined in this paper, applied within an edge computing environment, demonstrates a commendable performance. However, in practice, this strategy continuously adjusts in response to changes in each video frame, leading to the issue of frequent strategy reconfigurations, inadvertently escalating computational power consumption.

Therefore, in synergy with edge computing, this paper utilizes vehicle-mounted cameras to capture road environment imagery, subsequently offloading tasks like target detection to edge servers, and thereby optimizing computational power usage and enhancing data transmission efficiency.

3. System Model

3.1. System Architecture

The ongoing evolution of vehicle applications has resulted in an immense influx of data, exerting significant pressure on network capacity and bandwidth. While cloud computing addresses the issue of limited vehicle resources, its deployment over long distances results in considerable latency and increased bandwidth burden [21]. Therefore, the system architecture under edge computing is built by deploying edge servers, as shown in Figure 1. Offloading the object detection task from the autonomous driving system to the MEC server, coupled with the integration of the JPEG compression algorithm prior to the transmission of image data, mitigates issues like delayed data processing and the excessive latency inherent in cloud computing.

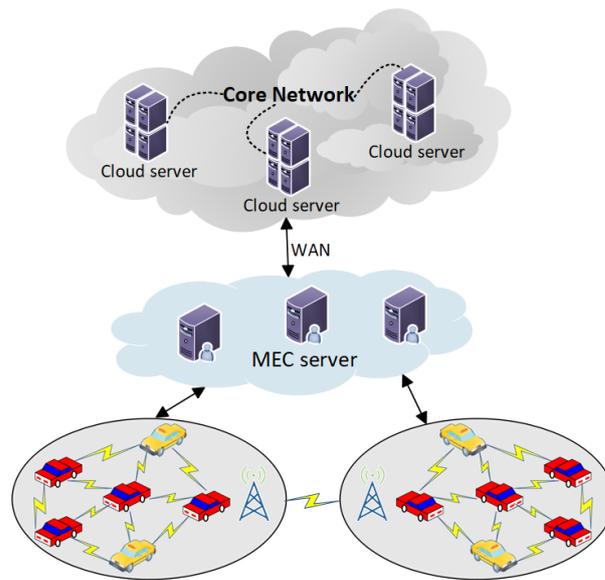


Figure 1. System network topology under MEC.

Figure 2 illustrates the flowchart of the entire system, encompassing image compression, the AD HOC MAC protocol, and the object detection module. Initially, image data reflecting road conditions are compressed by users within the IoV, thereby optimizing data storage and reducing transmission bandwidth requirements. Subsequently, the AD HOC MAC protocol is developed, and its performance evaluation model is established, leveraging a Markov chain approach. The processed data are then transmitted to the edge server to facilitate the object detection task. Ultimately, the resultant identification data are stored ready for subsequent dissemination.

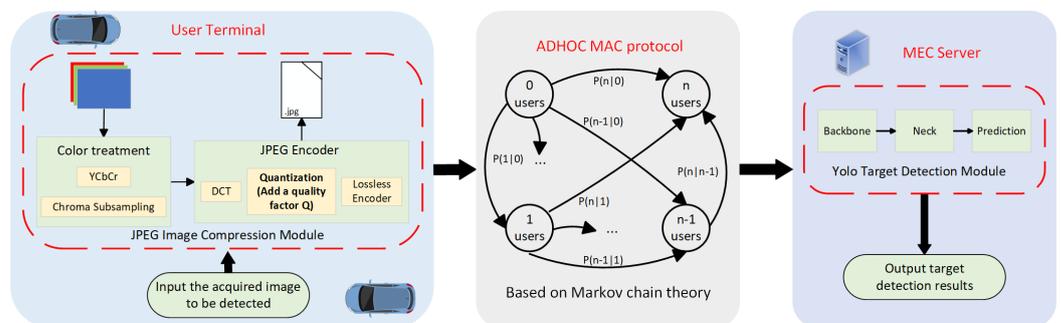


Figure 2. System structure.

Throughout the process, the principal components encompass the mathematical model analysis pertaining to the user service rate η of the system, the adjustment of the parameter

Q in the compression module (notably, the compression quality factor Q is an integer ranging from 0 to 100), and the formulation of standards for the target detection accuracy Acc . These components are further delineated in the subsequent sections of this article.

3.2. Ad Hoc MAC Protocol Performance Analysis Model

The AD HOC MAC protocol serves as a reliable broadcast MAC protocol specifically designed for vehicular ad hoc networks. This protocol is applicable in both single-hop and multi-hop communication environments [22]. To streamline the analysis, a multi-user network operating within a single-hop communication environment has been constructed, with each user gaining access to the network via a wireless link that connects to an AP (Access Point), as depicted in Figure 3. Given that the AD HOC MAC protocol adopts a time-division structure in real-world spatial networking applications, wherein multiple time slots collectively form time frames, N represents the number of available time slots in a frame ($N > 0$), and M denotes the number of vehicles in the network ($M > 0$).

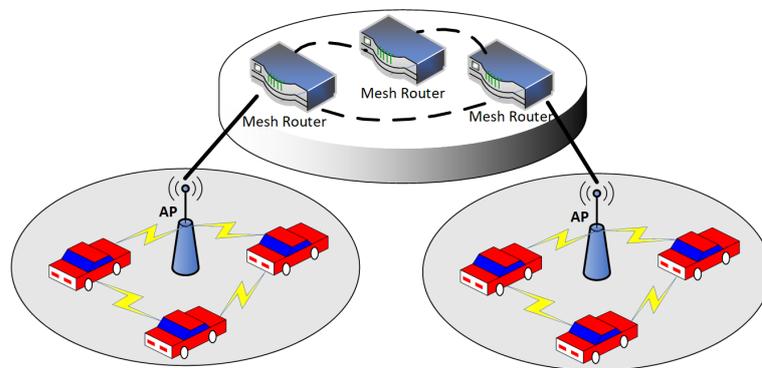


Figure 3. Single-hop communication environment.

For the sake of analytical simplicity, it is posited that, at the conclusion of each frame, every vehicle is able to discern whether a given time slot within the frame is occupied, and determines whether its attempt to acquire a time slot in that frame has been successful. Subsequently, any vehicle that fails in this attempt randomly selects a time slot from the remaining ones and endeavors to occupy it in the subsequent frame.

Define k as the number of vehicles when all users have successfully obtained a time slot, $k = \min\{M, N\}$. i represents the number of vehicles that successfully obtain a time slot in the initial frame, while j denotes the number of vehicles that successfully acquire a time slot by the end of the second frame. Under these conditions, the transition probability from state i to state j is denoted as $P(j|i)$, resulting in the formation of a stationary discrete Markov chain, as illustrated in Figure 4.

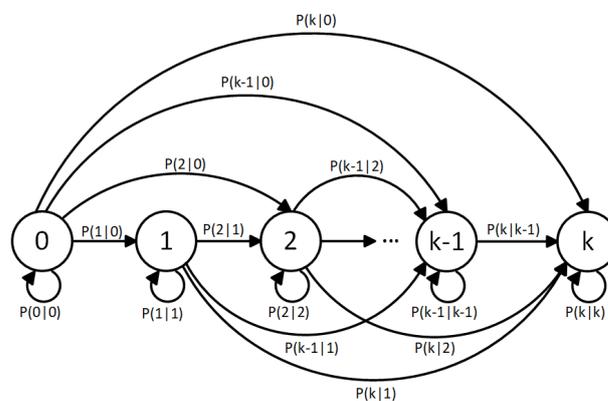


Figure 4. Markov chain state transition diagram.

3.2.1. State Transition Probability $P(j|i)$

Based on the aforementioned assumptions, at the conclusion of the first frame, $N - i$ time slots remain available for $M - i$ vehicles to attempt acquisition in the second frame. Given that a total of j vehicles successfully acquire a time slot by the conclusion of the second frame, it follows that $j - i$ vehicles will have successfully occupied a time slot by the end of the second frame.

Therefore, the state transition probability $P(j|i)$ can be defined as the probability that a $j - i$ vehicle successfully occupies a time slot when an $M - i$ vehicle randomly attempts to obtain one of the $N - i$ time slots.

Let $W(j - i, M - i, N - i)$ represent the number of cases in which $j - i$ vehicles successfully occupy a time slot when $M - i$ vehicles randomly try to obtain one of $N - i$ time slots, then the transfer probability $P(j|i)$ is defined as follows:

$$P(j|i) = \frac{W(j - i, M - i, N - i)}{(N - i)^{M - i}}, \quad j \leq k \tag{1}$$

Firstly, the derivation is carried out in the case of $M \leq N$. Since there are C_{M-i}^{j-i} ways to successfully obtain time slots by selecting $j - i$ cars from $M - i$ vehicles, there are C_{N-i}^{j-i} ways to select $j - i$ time slots from $N - i$ time slots, and there are $(j - i)!$ ways to align the remaining time slots. Therefore, it is not difficult to obtain a mathematical expression for $W(j - i, M - i, N - i)$:

$$W(j - i, M - i, N - i) = \begin{cases} C_{M-i}^{j-i} A_{N-i}^{j-i} [(N - j)^{M-j} - \sum_{l=1}^{M-j} W(l, M - j, N - j)], & 0 \leq j < M \\ A_{N-i}^{j-i}, & j = M \\ 0, & j > M \end{cases} \tag{2}$$

Similarly, the mathematical expression of $W(j - i, M - i, N - i)$ in the case of $M > N$ can be calculated:

$$W(j - i, M - i, N - i) = \begin{cases} C_{M-i}^{j-i} A_{N-i}^{j-i} [(N - j)^{M-j} - \sum_{l=1}^{N-j} W(l, M - j, N - j)], & 0 \leq j < N \\ 0, & j \geq N \end{cases} \tag{3}$$

The state transition probability is $P(j|i)$ when $M \leq N$ and $M > N$ can be obtained from Equation (1), Equation (2), and Equation (3), respectively. In order to simplify the derivation, the case of $M \leq N$ is discussed first.

By substituting Equation (2) into Equation (1), the state transition probability $P(j|i)$ for $M \leq N$ can be obtained. The calculation formula is as follows:

$$P(j|i) = \begin{cases} \frac{C_{M-i}^{j-i} A_{N-i}^{j-i}}{(N-i)^{M-i}} [(N - j)^{M-j} - \sum_{l=1}^{M-j} W(l, M - j, N - j)], & 0 \leq j < M \\ \frac{A_{N-i}^{j-i}}{(N-i)^{M-i}}, & j = M \\ 0, & j > M \end{cases} \tag{4}$$

In order to find out the general term of state transition probability, the relationship between $P(j|i)$ and $P(j|j)$ is first studied, and the following relationship can be found:

$$P(j|i) = \frac{C_{M-i}^{j-i} A_{N-i}^{j-i} (N - j)^{M-j}}{(N - i)^{M-i}} P(j|j) \tag{5}$$

According to the properties of Markov chains

$$\sum_{m=j}^M P(m|j) = 1 \tag{6}$$

$P(j|j)$ is obtained from Equation (6) and substituted into Equation (5). The calculation formula as follows:

$$P(j|i) = \frac{C_{M-i}^{j-i} A_{N-i}^{j-i} (N-j)^{M-j}}{(N-i)^{M-i}} \left[1 - \sum_{m=j+1}^M P(m|j) \right] \tag{7}$$

By expanding and simplifying the summation series in Equation (7), we obtain the formula is as follows:

$$P(j|i) = \frac{C_{M-i}^{j-i} A_{N-i}^{j-i} (N-j)^{M-j}}{(N-i)^{M-i}} \left\{ \begin{aligned} & 1 - \frac{C_{M-j}^1 A_{N-j}^1 [N-(j+1)]^{M-(j+1)}}{(N-j)^{M-j}} \\ & + \frac{C_{M-j}^1 A_{N-j}^1 [N-(j+1)]^{M-(j+1)}}{(N-j)^{M-j}} \sum_{s_1=j+2}^M P(s_1|j+1) \\ & - \frac{C_{M-j}^2 A_{N-j}^2 [N-(j+2)]^{M-(j+2)}}{(N-j)^{M-j}} \left[1 - \sum_{s_2=j+3}^M P(s_2|j+2) \right] \\ & - \frac{C_{M-j}^3 A_{N-j}^3 [N-(j+3)]^{M-(j+3)}}{(N-j)^{M-j}} \left[1 - \sum_{s_3=j+4}^M P(s_3|j+3) \right] \\ & - \dots - \frac{C_{M-j}^{M-j} A_{N-j}^{M-j} (N-M)^{M-M}}{(N-j)^{M-j}} \end{aligned} \right\} \tag{8}$$

From Equation (7), we obtain the formula as follows:

$$P(s_1|j+1) = \frac{C_{M-(j+1)}^{s_1-(j+1)} A_{N-(j+1)}^{s_1-(j+1)} (N-s_1)^{M-s_1}}{[N-(j+1)]^{M-(j+1)}} \left[1 - \sum_{s=s_1+1}^M P(s|s_1) \right] \tag{9}$$

By substituting Equation (9) into Equation (8), Equation (10) is simplified as follows:

$$P(j|i) = \frac{C_{M-i}^{j-i} A_{N-i}^{j-i} (N-j)^{M-j}}{(N-i)^{M-i}} \left\{ \begin{aligned} & 1 - \frac{C_{M-j}^1 A_{N-j}^1 [N-(j+1)]^{M-(j+1)}}{(N-j)^{M-j}} \\ & + \frac{C_{M-j}^2 A_{N-j}^2 [N-(j+2)]^{M-(j+2)}}{(N-j)^{M-j}} \\ & - \frac{C_{M-j}^2 A_{N-j}^2 [N-(j+2)]^{M-(j+2)}}{(N-j)^{M-j}} \sum_{s_2=j+3}^M P(s_2|j+2) \\ & + \left\{ \frac{C_{M-j}^3 A_{N-j}^3 [N-(j+3)]^{M-(j+3)}}{(N-j)^{M-j}} \right\} \left[1 - \sum_{s_3=j+4}^M P(s_3|j+3) \right] \\ & + \dots + \frac{C_{M-j}^1 C_{M-j}^{M-j} A_{N-j}^{M-j} (N-M)^{M-M}}{(N-j)^{M-j}} \end{aligned} \right\} \tag{10}$$

After iterating the above derivation process for $M - j$ times, the general term expression of $P(j|i)$, when $M \leq N$, can be obtained using the following formula:

$$P(j|i) = \frac{C_{M-i}^{j-i} A_{N-i}^{j-i}}{(N-i)^{M-i}} \sum_{l=0}^{M-j} (-1)^l C_{M-j}^l A_{N-j}^l [N-(j+l)]^{M-(j+l)} \tag{11}$$

Similarly, it is not difficult to derive the general term expression of $P(j|i)$ when $M > N$:

$$P(j|i) = \frac{C_{M-i}^{j-i} A_{N-i}^{j-i}}{(N-i)^{M-i}} \sum_{l=0}^{N-j} (-1)^l C_{M-j}^l A_{N-j}^l [N-(j+l)]^{M-(j+l)} \tag{12}$$

3.2.2. Number of Successful Service Users N_v and Service Rate η in the First Frame

Assume that, at the end of the first frame, there are j vehicles successfully occupying a time slot, and no vehicles initially occupy the time slot. Therefore, the transition probability $P(j|0)$ can be obtained as follows:

$$N_v = \sum_{j=0}^M jP(j|0) \tag{13}$$

By combining Equations (11) and (12), the mathematical expression of N_v can be obtained using the following formula:

$$N_v = \begin{cases} \sum_{j=0}^M \sum_{l=0}^{M-j} \frac{(-1)^l j C_M^j C_{M-j}^l A_N^{j+l} [N-(j+l)]^{M-(j+l)}}{N^M}, & M \leq N \\ \sum_{j=0}^N \sum_{l=0}^{N-j} \frac{(-1)^l j C_M^j C_{M-j}^l A_N^{j+l} [N-(j+l)]^{M-(j+l)}}{N^M}, & M < N \end{cases} \tag{14}$$

To simplify the computation of Equation (14), set $c = j + l$ to obtain the following formula:

$$N_v = \begin{cases} \sum_{c=0}^M \sum_{l=0}^c \frac{(-1)^l (c-l) C_M^{c-l} C_{M-(c-l)}^l A_N^c (N-c)^{M-c}}{N^M}, & M \leq N \\ \sum_{c=0}^N \sum_{l=0}^c \frac{(-1)^l (c-l) C_M^{c-l} C_{M-(c-l)}^l A_N^c (N-c)^{M-c}}{N^M}, & M < N \end{cases} \tag{15}$$

Next, the summation series in N_v is discussed in detail, and the cases when $c = 0$, $c = 1$, and $c > 1$ are calculated, respectively. After calculation, the mathematical expression of N_v can be derived as follows:

$$N_v = \frac{M(N-1)^{M-1}}{N^{M-1}}, \quad M \geq 1, N \geq 1 \tag{16}$$

The metric N_v derived above can be used to intuitively assess the change in average access users based on the number of users in the vehicular network. With the ongoing advancement in communication technology, the user service rate, particularly from the perspective of data transmission, has become a vital metric in evaluating the performance of the IoV. Considering the intuitive meaning of the user service rate, its calculation method can be expressed as follows:

$$\text{User Service Rate } \eta = \frac{\text{Number of users successfully served}}{(\text{served} + \text{not served}) \text{ Total number of users}} \tag{17}$$

From the perspective of information transmission within the IoV, a user being successfully serviced implies that the data intended for transmission by the user have been successfully sent. Consequently, the calculation method for the user service rate can be transformed into the ratio of data transmission time to total time.

When the user transmits data, information collision may occur after data are sent, so the user will wait for a period of time and retransmit the data several times until the transmission is successful. Therefore, the total time for successfully sending a frame of data includes the frame sending time T_0 , propagation delay t , and collision waiting time C_t . Assume that the data transmission rate is T_s , the channel bandwidth is B_w the size of the data to be transmitted is S (that is, the space occupied by the image), and the average

duration for each user to be successfully served is 2 s. Therefore, the user service rate is defined as the ratio of the data transmission time T_0 of the successfully served user end to the total transmission time. The formula is as follows:

$$\eta = \frac{T_0/2}{(t + C_t + T_0)/2} = \frac{T_0}{t + C_t + T_0} \tag{18}$$

T_0 , t , and C_t are calculated as follows:

$$T_0 = \frac{N}{T_s} \tag{19}$$

$$t = \frac{S}{B_w} \tag{20}$$

$$C_t = 2nt \tag{21}$$

where n is the number of contention periods when a collision occurs and N is the frame length.

4. Optimization Analysis

In this section, an extensive optimization analysis of the system is undertaken. Following the construction of the system model, we commence with the optimization problem aimed at maximizing the user service rate. Additionally, an image compression algorithm is introduced for further optimization, tailored to meet the requirements of object detection accuracy, Acc . Utilizing theoretical analysis and simulation verification, the effectiveness of the optimization is substantiated through a comparative analysis of results pre- and post-optimization.

4.1. Optimization Problem

According to the theoretical derivation of user service rate η in Section 3.2.2, we substituted Equation (18) into Equation (21) to obtain the formula as follows:

$$\eta = \frac{\frac{N}{T_s}}{\frac{S}{B_w} + 2n\frac{S}{B_w} + \frac{N}{T_s}} \tag{22}$$

Therefore, in order to maximize the user service rate, it is necessary to make the numerator of the fraction in Equation (22) as large as possible and the denominator as small as possible. However, the parameters of sending rate T_s , bandwidth B_w , and frame length N correspond to the communication technology and resources in the network, so these are difficult to optimize and adjust. Thus, it can be concluded that the controllable parameter is the transmitted data size S . In order to obtain η_{max} , we need to minimize S .

Based on the aforementioned succinct analysis, the optimization problem addressed in this paper can be formulated as follows:

$$\begin{aligned} & \max_{Q_{min}} \eta \\ & s.t. Acc_{max Q} \leq Acc_{original} + 0.3 \\ & Acc_{min Q} \geq Acc_{original} - 0.3 \end{aligned} \tag{23}$$

Due to the rapid development of vision devices, the quality of images obtained by users in the IoV is becoming higher and higher, so the image compression algorithm is introduced to minimize the size of the transmitted image data. However, for the object detection module, high image compression may lead to a reduction in detection accuracy and an increase in the recognition error rate. If the object detection requirements cannot be met, the driving safety will be seriously affected. Therefore, while optimizing based on the image compression algorithm, it is still necessary to meet the detection accuracy requirements as the benchmark, not only to achieve a significant compression effect to

maximize the user service rate, but also to meet the object detection requirements to ensure traffic safety.

4.2. Optimization Analysis Based on JPEG Image Compression Technology

The JPEG compression algorithm is employed to compress image data, thereby saving space and facilitating the faster transmission of a greater volume of image or video data per unit time [23]. The compression ratio achieved by the JPEG compression algorithm is unrivaled by other traditional compression algorithms. Furthermore, the resultant image file size is significantly reduced, thereby greatly minimizing the volume of data that require processing. This algorithm primarily consists of four steps: preprocessing, image segmentation and discrete cosine transform (DCT), followed by quantization and coding [24]. The detailed algorithmic flowchart is depicted in Figure 5.

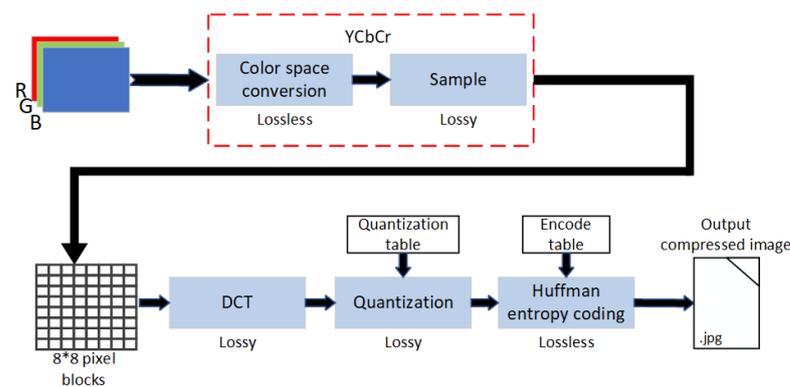


Figure 5. JPEG compression steps.

Upon the transformation of the image into 8×8 pixel blocks, these blocks typically exhibit low spatial frequency, indicating that the pixel values undergo gradual changes. Consequently, the application of the DCT results in the concentration of energy in these pixel blocks into specific low-frequency components. This approach not only eliminates data redundancy but also enables more efficient compression processing. DCT implementation is calculated as follows:

$$F(u, v) = \frac{1}{4}C(u)C(v) \left[\sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \right] \quad (24)$$

$$\text{where } x, y, u, v = 0, 1, \dots, 7, C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}}, & u, v = 0 \\ 1, & \text{else} \end{cases}$$

Because the high-frequency component of the image is the detailed part of the image, it has little effect on the whole. Therefore, quantization cannot only discard the information in the image that does not affect the visual effect, but can also ensure image quality. In this paper, the compression algorithm uses uniform quantization, according to the JPEG brightness quantization table and chroma quantization table, and the transformed DCT coefficients are divided by the corresponding compression quality factor, so as to achieve the purpose of compression. The quantization formula is as follows:

$$F'_{uv} = \left[\frac{F_{uv}}{Q} \right] \quad (25)$$

where F'_{uv} and F_{uv} are the DCT coefficients before and after quantization, Q is the compression mass factor (Q is an integer value between 0 and 100), and $[\cdot]$ represents rounding. After this, the low frequency component is retained for Huffman coding, and the high frequency component is removed to finally output the compressed result.

Because the corresponding quantization table is different under different Q values, the larger Q is, the smaller the quantization value is, that is, the smaller the loss is, and the correspondingly compressed image occupies more space. Therefore, the nonlinear mapping relationship between Q and the size of the image-occupied space can be obtained as follows:

$$S \propto f(Q) \quad (26)$$

Taking the real test image dorms.jpg as an example, a comparison between the original image and the image under different compression quality factors is shown in Figure 6. To compare image quality more closely, we zoom in on the red-framed area of the image. It can be seen that the image with a compression quality factor Q of 5 has obvious distortion and independent pixel blocks compared with the original image. However, when the compression quality factor Q is 50, there is almost no error between the image and the original image, and high image quality is maintained.

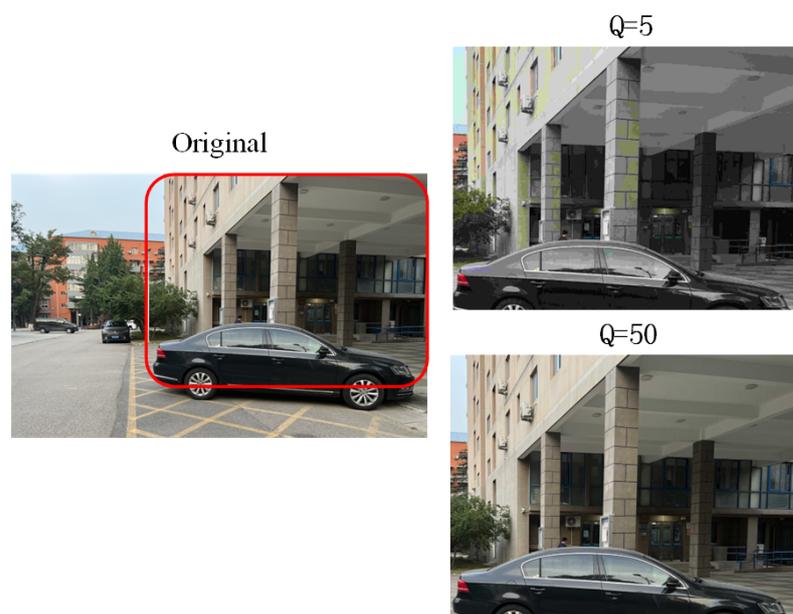


Figure 6. dorm.jpg original and compressed image comparison.

The test images utilized in this paper are presented in Figure 7. Following comprehensive testing across the entire range of compression quality factors, the relationship between the compression quality factor Q and the compression ratio r (where r represents the ratio of the original image size to the compressed image size) is illustrated in Figure 8. Furthermore, the results corroborate the validity of the positive nonlinear correlation as delineated in Equation (26).



Figure 7. (a) dorm.jpg. (b) test1.jpg. (c) test2.jpg.

In the object detection module, this paper employs the Yolov5 object detection algorithm, noted for its high precision, rapid processing speed, and broad applicability, to accomplish precise vehicle identification, utilizing both public datasets and a specially

selected test set for evaluation purposes. Yolov5s, a variant within the Yolov5 series, is a single-stage target detection algorithm characterized by minimal network parameters, while concurrently being optimized for speed and accuracy, making its detection speed particularly well-suited for vehicle target detection scenarios [25].

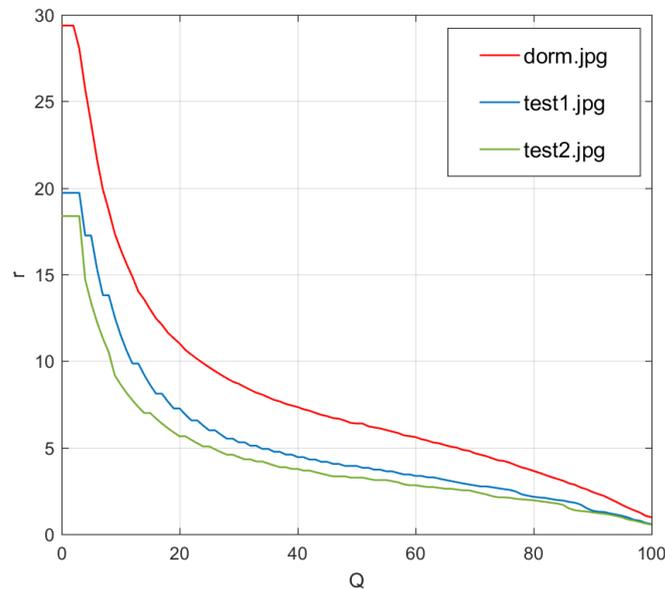


Figure 8. Compression quality factor Q and compression ratio r relationship.

The schematic diagram of the Yolov5s object detection principle is shown in Figure 9, and it mainly consists of three network layers: the backbone, neck, and head. The backbone network layer is the main part of the convolutional neural network, used for extracting image features; the neck layer is the module for feature fusion and dimension reduction; and the head layer is the module used for predicting object categories and locations.

In this paper, the Yolov5s object detection framework is carried out in Python environment, and network training and testing are deployed in the CPU environment. Thirty street-view images in the training set are selected from the public data set, CBCL StreetScenes Challenge Images. The test set is self-selected image data. In the performance evaluation metrics for the Yolov5s object detection model, the most critical ones include precision, recall, $mAP@0.5$, and $mAP@0.5:0.95$. Precision reflects the rate of false positives, with higher precision indicating fewer false detections; recall reflects the rate of false negatives, with higher recall indicating fewer missed detections; and $mAP@0.5$ and $mAP@0.5:0.95$ reflect the accuracy of detection under the Intersection over Union (IoU) thresholds. The performance of the model after the final training is shown in Figure 10. It can be seen that, after 150 iterations of the network, the indicators change slowly, and the average accuracy mAP is 0.95, which basically meets the accuracy requirements for object detection and recognition in this paper.

The trained model is used to detect and recognize the images of the test set, and the resulting graph is the output. The detection accuracy of the original image is used as the standard for subsequent judgment. The test set consists of three images, two of which are randomly selected from the public data set, CBCL StreetScenes Challenge Images, and the other is the image data of real scene shooting. The test results are shown in Figure 11.

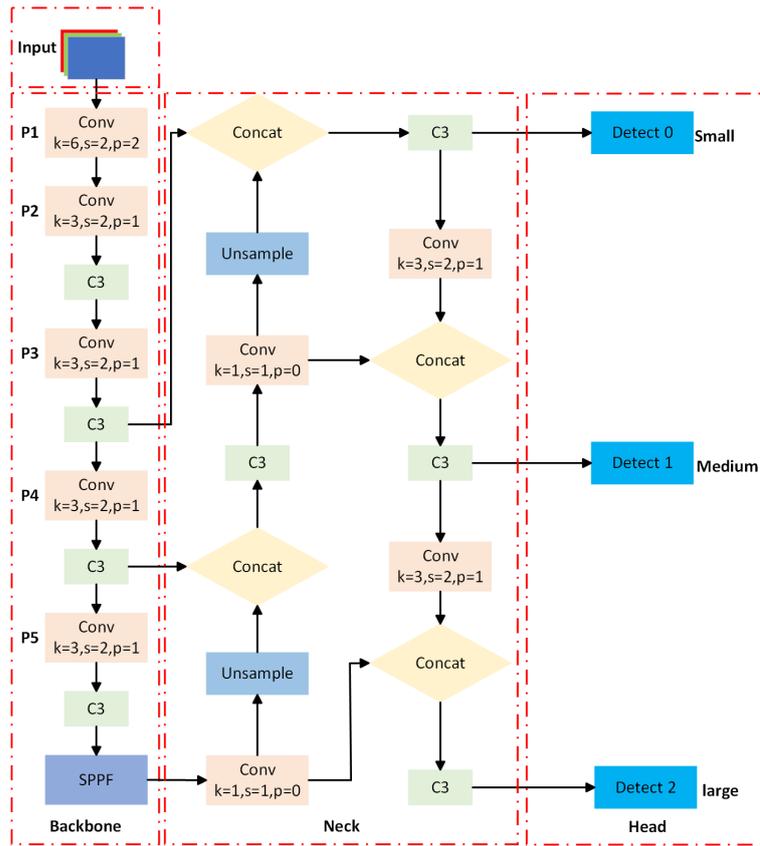


Figure 9. YOLOv5s network schematic.

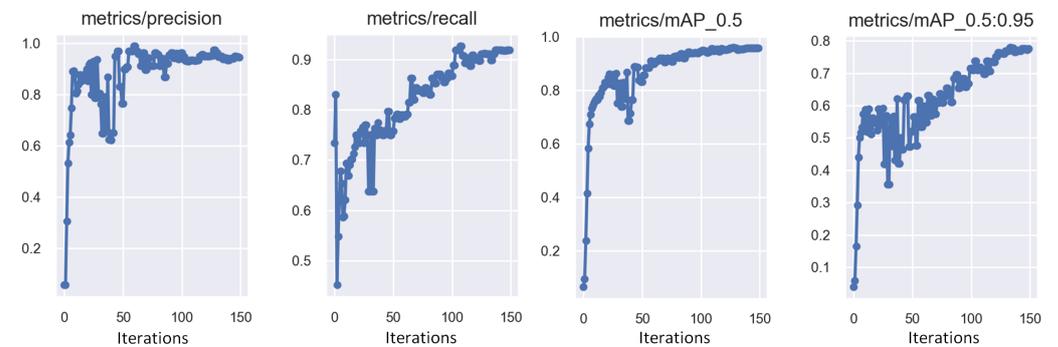


Figure 10. Yolo object detection model performance graph.

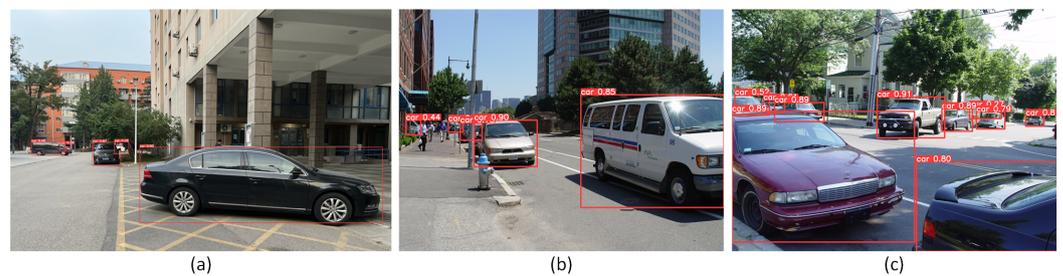


Figure 11. (a) dorm.jpg uncompressed test result. (b) test1.jpg uncompressed test result. (c) test2.jpg uncompressed test result.

Consequently, provided that the detection accuracy remains unaffected, a higher compression ratio proves increasingly beneficial for enhancing data transmission efficiency and user service rates within the IoV. Subsequently, the study will focus on determining

the maximum compression limit that does not compromise detection accuracy. Initially, the recognition accuracy of all compressed image data is evaluated, employing different serial numbers to distinguish various target vehicles in the test set images, as shown in Figure 12. And observing and charting the relationship between the detection accuracy value Acc and the compression quality factor Q , as shown in Figure 13.



Figure 12. (a) dorm.jpg vehicle serial number marking. (b) test1.jpg vehicle serial number marking. (c) test2.jpg vehicle serial number marking.

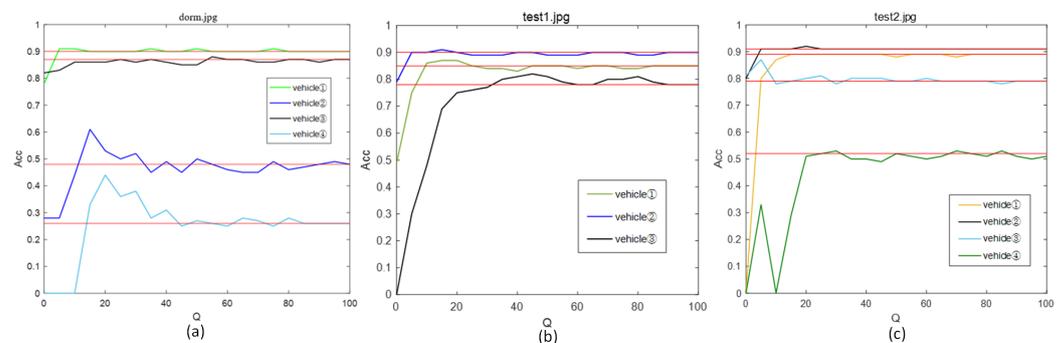


Figure 13. (a) dorm.jpg Acc-Q diagram. (b) test1.jpg Acc-Q diagram. (c) test2.jpg Acc-Q diagram.

The results of the Acc-Q relationship chart were analyzed, and the red horizontal line in each graph was the detection accuracy of the target vehicle in the original graph, which was used as the benchmark for further analysis. On the whole, when the compressed quality factor $Q > 20$, the detection accuracy of the vehicle basically fluctuates around the detection accuracy of the original image. Compared with the data, it can be seen that the floating range does not exceed 0.03 when the detection accuracy of the original image is not reached. When the compressed quality factor $Q < 20$, the detection accuracy of the target vehicle begins to decrease, and the phenomenon that the original target vehicle cannot be identified appears, which cannot meet the basic needs of target detection. Under different compression mass factors, the recognition accuracy of some target vehicles will be improved compared with the original figure, as shown in Figure 13a. This is because JPEG compression will eliminate the spatial redundancy information of the image, so that the frequency information of the detection target is relatively prominent; thus, the detection accuracy will be improved. Therefore, it can be determined that the compression limit meeting the requirement of target detection accuracy is $Q = 20$.

Based on the above analysis, the optimization algorithm can be summarized as follows (Algorithm 1).

Building upon the aforementioned analysis and the verification of simulation results, the efficacy of integrating a compression module is thoroughly demonstrated. Subsequently, Section 5 will present the simulation results and an analysis of the system optimization.

Algorithm 1: User service rate maximization algorithm based on image compression in IoV

```

Input: Input RGB image data image.jpg and the compression quality factor Q
Output: Output compression limit  $Q_{min}$ , maximum user service rate  $\eta_{max}$ 
% Detection and recognition of image target vehicle based on Yolo model
Detect (image.jpg);
Acc1 = results; % Save the detection accuracy results of the uncompressed image
Q = 0; % Initializes the compression quality factor
while(Q ≤ 100)
{
compression ('image.jpg', Q); %Compressed image corresponding to Q value
% Store the detection result under the corresponding Q value
Acc Q = results(image.jpg);
% Set the floating range of detection accuracy
If((Acc1 - 0.3) ≤ Acc[Q] || (Acc1 + 0.3) > Acc[Q])
{Q++;}
else
{ $Q_{min}=Q$ ;
I = imfinfo ('image.jpg');% Read the image data under the compression limit
S = I.FileSize;
 $\eta_{max}(S)$ ; % The maximum user service rate is calculated }
end
}

```

5. Simulation Results

Owing to the above analysis, the performance of the proposed optimization scheme is further evaluated through simulation.

Based on Equation (16) in Section 3.2, when M is constant, we can conclude that $N_v \propto N$. Next, we assume that each time slot is used to transmit data of the same size. We can establish the mathematical relationship as follows: $E = N \times S$, where E represents the total transmitted volume. We can therefore derive the mathematical relationship between N_v and S as follows:

$$N_v = M \left(1 - \frac{S}{E}\right)^{M-1} \quad (27)$$

Based on Equation (26), the relationship between N_v and Q can be obtained as follows:

$$N_v \propto \frac{1}{f(Q)} \quad (28)$$

By setting the total transmitted data amount $E = 1$ Mb, we can obtain the simulation results shown in Figure 14. An analysis of the results presented in Figure 14 reveals that a reduction in the compression quality factor, effectively increasing the compression ratio, results in a higher average number of vehicles.

Based on the analysis of service rate in Section 3.2, JPEG compression algorithm is added to optimize the user service rate. The optimization algorithm is shown in Algorithm 1. According to Equation (22), set the transmission rate $T_s = 80$ Mbps and $B_w = 100$ Mbps; the service rate of transmitting uncompressed original image data and sending compressed image data can be compared, as shown in Figure 15, which mainly compares the service rate when the compression quality factor is 20, 50, and 80. It can be seen from the results that different Q values have different degrees of influence on the service rate. Since Q value is proportional to the occupied space of the image, the smaller the Q value is, the smaller the occupied space of the compressed image data is, and the more obvious the service rate improvement is. Among them, the service rate of the test image dorm.jpg is increased by

34% at most, and the service rate of the two test images, test1.jpg and test2.jpg, randomly selected from the public data is increased by 11% at most.

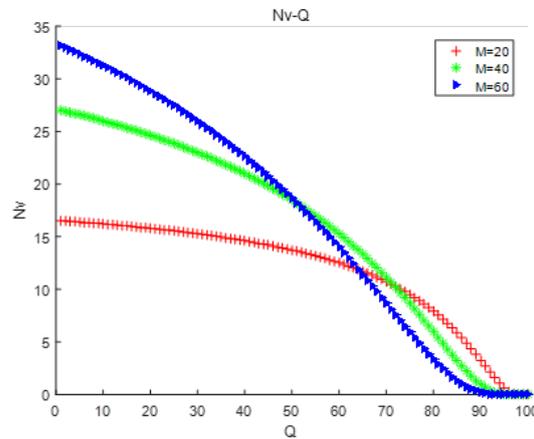


Figure 14. The relationship between the average number of users N_v and the compression quality factor Q .

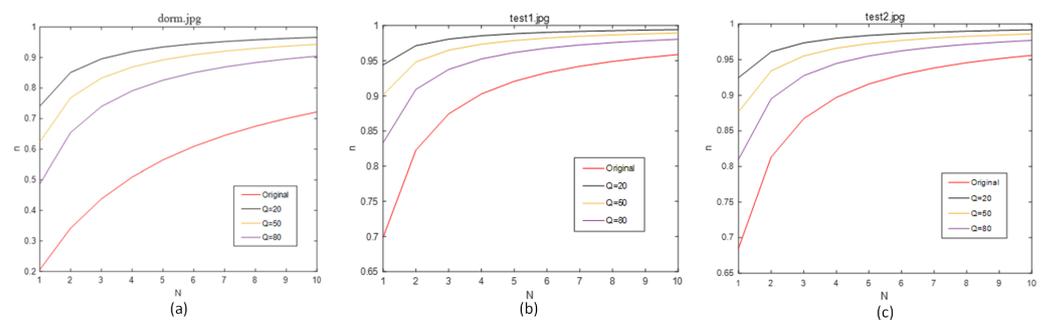


Figure 15. (a) dorm.jpg comparison of user service rate under different compression quality factors; (b) test1.jpg comparison of user service rate under different compression quality factors; and (c) test2.jpg comparison of user service rate under different compression quality factors.

Consequently, the resource optimization scheme that incorporates image compression technology within an edge computing framework is utilized to compress the image data prior to transmission, not only conserving data space but also significantly enhancing the average number of users accessing the IoV as well as the user service rate.

6. Conclusions

This study investigated the resource optimization issue for object detection in the IoV within the context of edge computing, aiming to enhance the service user rate and more effectively meet real-world application demands. Initially, the paper analyzed the system architecture both with and without edge computing, introducing the modules and operational flow of the optimized system. Subsequently, a performance analysis model for the AD HOC MAC protocol was established based on the Markov chain, elucidating the positive mathematical correlation between the average service users N_v and the frame length N . Additionally, this study explored the negative mathematical correlation between the user service rate η and the data size S . Following this, we introduced an image compression algorithm for further optimization. Utilizing the positively correlated nonlinear mapping relationship between the compression quality factor Q and the data size S , adjustments to Q values were made to facilitate image processing transmission under varying compression ratios, subsequently deriving the negatively correlated nonlinear mapping relationship between the user service rate η and the compression quality factor Q . Utilizing the Yolov5s object detection model, an evaluation was conducted to ascertain whether the

compression results satisfy the detection accuracy requirements, and the minimal compression limit Q_{min} and the corresponding maximum user service rate η_{max} under these accuracy requirements were then derived. The conclusive optimization results demonstrate that integrating edge computing with compression processing significantly enhances the user service rate. Specifically, the service rate for open datasets sees an increase of up to 11%, while, for real-scene shooting data, the increase reaches up to 34%.

In future research, certain idealized conditions will be taken into account, including an examination of the access performance of the AD HOC MAC protocol in a multi-hop network environment, the enhancement of the accuracy of the target detection model, and the impact on the service rate due to data collisions during information transmission.

Author Contributions: Conceptualization, L.Z.; methodology, L.Z.; software, J.L.; validation, W.G.; formal analysis, L.Z. and J.L.; investigation, W.G.; resources, X.L.; data curation, W.G.; writing—original draft preparation, L.Z.; writing—review and editing, X.L.; visualization, W.G.; supervision, X.L.; project administration, J.L.; funding acquisition, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was an R&D Program of the Beijing Municipal Education Commission (KM202310011002).

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Ding, Z.; Xiang, J. Overview of intelligent vehicle infrastructure cooperative simulation technology for IoV and automatic driving. *World Electr. Veh. J.* **2021**, *12*, 222. [[CrossRef](#)]
- Yurtsever, E.; Lambert, J.; Carballo, A.; Takeda, K. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access* **2020**, *8*, 58443–58469. [[CrossRef](#)]
- Zineb, S.; Lyamine, G.; Kamel, B.; Rezki, A. IoV Data Processing Algorithms for Automatic Real-Time Object Detection-Literature Review. In Proceedings of the 2023 International Conference on Inventive Computation Technologies (ICICT), Lalitpur, Nepal, 26–28 April 2023.
- Liu, Z.; Cai, Y.; Wang, H.; Chen, L.; Gao, H.; Jia, Y.; Li, Y. Robust target recognition and tracking of self-driving cars with radar and camera information fusion under severe weather conditions. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 6640–6653. [[CrossRef](#)]
- Chettri, L.; Bera, R. A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems. *IEEE Internet Things J.* **2019**, *7*, 16–32. [[CrossRef](#)]
- Gür, G.; Kalla, A.; De Alwis, C.; Pham, Q.V.; Ngo, K.H.; Liyanage, M.; Porambage, P. Integration of ICN and MEC in 5G and beyond networks: Mutual benefits, use cases, challenges, standardization, and future research. *IEEE Open J. Commun. Soc.* **2022**, *3*, 1382–1412. [[CrossRef](#)]
- Hui Ernest, T.Z.; Madhukumar, A.S. An Energy Efficiency Analysis of Computation Offloading in MEC-Enabled IoV Networks. In Proceedings of the 2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring), Florence, Italy, 20–23 June 2023.
- Bharadwaj, N.A.; Rao, C.S.; Gururaj, C. Optimized data compression through effective analysis of JPEG standard. In Proceedings of the 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 5–7 March 2021.
- Liu, J.; Zhang, R. Vehicle detection and ranging using two different focal length cameras. *J. Sens.* **2020**, *2020*, 4372847.
- Liang, X.; Du, X.; Wang, G.; Han, Z. A deep reinforcement learning network for traffic light cycle control. *IEEE Trans. Veh. Technol.* **2019**, *68*, 1243–1253. [[CrossRef](#)]
- Dai, C.; Liu, X.; Chen, W.; Lai, C.F. A low-latency object detection algorithm for the edge devices of IoV systems. *IEEE Trans. Veh. Technol.* **2020**, *69*, 11169–11178. [[CrossRef](#)]
- Diwan, T.; Anirudh, G.; Tembhurne, J.V. Object detection using YOLO: Challenges, architectural successors, datasets and applications. *Multimed. Tools Appl.* **2023**, *82*, 9243–9275. [[CrossRef](#)]
- Zhao, J.; Hao, S.; Dai, C.; Zhang, H.; Zhao, L.; Ji, Z.; Ganchev, I. Improved vision-based vehicle detection and classification by optimized YOLOv4. *IEEE Access* **2022**, *10*, 8590–8603. [[CrossRef](#)]
- Rahman, M.A.; Hamada, M.; Shin, J. The impact of state-of-the-art techniques for lossless still image compression. *Electronics* **2021**, *10*, 360. [[CrossRef](#)]
- Patwa, N.; Ahuja, N.; Somayazulu, S.; Tickoo, O.; Varadarajan, S.; Koolagudi, S. Semantic-preserving image compression. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 25–28 October 2020.

16. Jalilian, E.; Hofbauer, H.; Uhl, A. Iris Image Compression Using Deep Convolutional Neural Networks. *Sensors* **2022**, *22*, 2698. [[CrossRef](#)] [[PubMed](#)]
17. Hou, L.; Gregory, M.A.; Li, S. A Survey of Multi-Access Edge Computing and Vehicular Networking. *IEEE Access* **2022**, *10*, 123436–123451. [[CrossRef](#)]
18. Zeng, W.; Gao, Y.; Pan, F.; Yan, Y.; Yu, L.; Li, Z. Towards Real-time Object Detection on Edge Devices for Vehicle and Pedestrian Interaction Scenarios. In Proceedings of the 2022 41st Chinese Control Conference (CCC), Hefei, China, 25–27 July 2022.
19. Lian, Y.; Qian, L.; Ding, L.; Yang, F.; Guan, Y. Semantic fusion infrastructure for unmanned vehicle system based on cooperative 5G MEC. In Proceedings of the 2020 IEEE/CIC International Conference on Communications in China (ICCC), Chongqing, China, 9–11 August 2020.
20. Xiao, X.; Zhang, J.; Wang, W.; He, J.; Zhang, Q. Dnn-driven compressive offloading for edge-assisted semantic video segmentation. In Proceedings of the IEEE INFOCOM 2022-IEEE Conference on Computer Communications, London, UK, 2–5 May 2022.
21. Wang, X.; Duan, X.; Sun, T. Service-based network as a platform: Research on a new information communication network architecture. *Telecommun. Sci.* **2023**, *39*, 20.
22. Ma, M.; Liu, K.; Luo, X.; Zhang, T.; Liu, F. Review of MAC protocols for vehicular ad hoc networks. *Sensors* **2020**, *20*, 6709. [[CrossRef](#)]
23. Xiao, W.; Wan, N.; Hong, A.; Chen, X. A fast JPEG image compression algorithm based on DCT. In Proceedings of the 2020 IEEE International Conference on Smart Cloud (SmartCloud), Washington, DC, USA, 6–8 November 2020.
24. Dimililer, K. DCT-based medical image compression using machine learning. *Signal Image Video Process.* **2022**, *16*, 55–62. [[CrossRef](#)]
25. Zhang, Y.; Guo, Z.; Wu, J.; Tian, Y.; Tang, H.; Guo, X. Real-time vehicle detection based on improved yolo v5. *Sustainability* **2022**, *14*, 12274. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.