SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I6P103

Volume 11 Issue 6, 22-30, June 2024

Original Article

Cloud Computing Framework for Vehicle Multimedia with Dynamic Priority-based Efficient Resource Allocation

Mohammad Khaja Nizamuddin[1], Zunaira Begum[2], C. Atheeq[3], Ruhiat Sultana[4], Ahsan Saud Qadri[5]

[1, 3]Department of CSE, GITAM University, Telangana, India.
[2] Department of CSM, Lords Institute of Engineering and Technology, Telangana, India.
[4] Department of IT, Lords Institute of Engineering and Technology, Telangana, India.
[5]Sheikh Abdullah Technical Institute, Ministry of Education Bahrain, Bahrain.

[3]Corresponding Author : atheeq.prof@gmail.com

Received: 02 April 2024 Revised: 03 May 2024 Accepted: 02 June 2024 Published: 29 June 2024

Abstract *- Intelligent transportation systems rely on smart vehicles equipped with a diverse array of sensory devices to deliver a spectrum of multimedia applications, including driving assistance, traffic status updates, weather forecasts, safety alerts, and entertainment features. However, the substantial volume of multimedia data generated by these vehicles overwhelms the processing due to their restricted processing speed and storage capacity, standalone onboard computer systems. Consequently, a shift in networking and computational paradigms is imperative to accommodate these multimedia services effectively. Cloud computing emerges as a viable solution for seamlessly integrating vehicles into the cloud infrastructure. Nevertheless, challenges related to multimedia content processing, encompassing resource costs, swift service response times, and optimized user experiences, can significantly influence vehicular communication performance. We provide effective resource allocation and computation architecture designed specifically for vehicle multimedia cloud computing to overcome these issues. Through the Cloudsim simulator, this framework's performance is thoroughly assessed with an emphasis on user experience, service response times, and resource costs.*

Keywords *- Intelligent transportation system, Smart vehicles, Efficient resource allocation, Cloud computing, QoS.*

1. Introduction

An idea for effectively managing resources in vehicular multimedia cloud computing settings is Vehicle Multimedia Cloud Computing: Dynamic Priority-based Efficient Resource Allocation and Computing Framework. The strategy is intended to offer dynamic resource allocation depending on the importance of cloud-based multimedia apps. While vehicular technology and cloud computing have both advanced substantially, the integration of these domains continues to grapple with persistent inefficiencies in resource allocation that meet the real-time requirements of vehicular multimedia applications. This gap highlights the urgent need for a framework capable of accommodating the specific demands for rapid processing and extensive bandwidth inherent in real-time vehicular multimedia tasks.

The framework consists of several crucial parts, including modules for resource allocation, priority-based scheduling, load balancing, and Quality of Service (QoS) monitoring. According to the priority levels of the various multimedia apps running in the cloud, the resource allocation module is in charge of assigning resources to them. The load balancing module aids in distributing the workload equally across the cloud computing environment, while the priority-based scheduling module makes sure that higher-priority applications are given preference when resources are allotted [1]. The performance of the multimedia apps is tracked using the QoS monitoring module, which also makes sure that the necessary QoS levels are maintained.

The suggested method includes a mathematical foundation for resource distribution. This methodology takes into account multimedia applications' hierarchical importance, cloud resources, and QoS requirements [2]. Using an optimization technique, the mathematical model allocates resources to maximize cloud-hosted multimedia application QoS. This technique, called the Vehicle Multimedia Cloud Computing Dynamic Priority-based Efficient Resource Allocation and Computing Framework, attempts to improve resource allocation in vehicular multimedia cloud computing settings while meeting multimedia application QoS requirements [3, 4].

Autonomous vehicles, developed by the auto industry and academics, require a fast internet connection. These vehicles have sensors that record high-resolution photos and videos and process enormous volumes of sensory data for safe and comfortable driving. They also use roadside infrastructure to interchange traffic load, road safety, map location, automatic parking, cooperative driving, and driver assistance [5]. Due to limited storage and computational capabilities, onboard devices cannot process this massive multimedia data. In addition, inconsistent connectivity, restricted radio communication range, bandwidth limits, and swift mobility make it difficult. Cloud computing is an emerging method for processing massive amounts of data fast and cheaply without hardware.

Integrating Cloud Computing (CC) with intelligent vehicles can increase multimedia services and open new applications and research possibilities. However, conventional CC is unsuitable for multimedia applications and services that require low latency and criticality. Multimedia Cloud Computing (MCC) addresses these challenges by providing QoS to apps. Multimedia data processing in vehicle networks is complex and requires fast computation, responsiveness, and cost reduction.

The real-time transmission of weather or accident information can prevent accidents and save lives. The Framework for Dynamic Priority-based Efficient Resource Allocation and Computing (DP-ERACOM) handles sensitive and fast multimedia calculations like image and video data in automotive networks. The DP-ERACOM technique divides each multimedia activity into four subtasks to dynamically allocate MCC resources based on task priority [6-8].

2. Related Work

Vehicle cloud uses include road safety, urban environment monitoring, content delivery, mobile advertising, and intelligent transportation. Vehicles get data from sensors (traffic flow, road conditions, nearby vehicles, environmental sights, marketing, etc.). Because its local relevance and volume make uploading to the Internet impracticable, this data is organized, exchanged, and stored locally. By indexing and scoping, the vehicle cloud lets other vehicles and Internet users find data. Furthermore, the cloud platform has tremendous computing capabilities. Processing tasks include assessing urban traffic congestion, creating pollution maps, jointly recreating accident or crime scene photos or films, and identifying security risks [9-12].

Slow traffic, clogged roads, and accidents plague modern cities [13]. Traffic management and transportation are difficult due to auto-expansion. Cities have dangerous roads and unsustainable landscapes due to traffic [14]. Traffic systems often fail to notify of congestion due to inefficient and unsafe transportation infrastructure. By extending highway lanes and lowering traffic signs, transportation management systems have improved temporarily but not urban complexity [15].

In contrast, technology enables traffic management and driving apps. Accurate traffic flow control, collaborative traffic monitoring, and road danger identification are examples. These services and apps use networked cars that share data. Large investments in Vehicular Ad-hoc Networks (VANETs), inspired by Mobile Ad-hoc Networks, illustrate their growing popularity [16, 17]. VANETs are solving several transportation challenges due to technological advances, solution development, and application support. Vehicular networks use V2V, V2I, and V2X communication.

For high-quality multimedia applications, the study combined a pioneering Cloud computing and IP Multimedia Subsystem (IMS) architecture. This innovative approach improves cloud computing using heterogeneous networking, QoS constraints, and MapReduce analysis. This design helps android users use high-quality multimedia apps. Cloud computing unites 3G, WiFi, and WiMAX IMS QoS standards for VoIP and video streaming [18]. Service priority-based resource allocation enhances system performance and user capacity, according to empirical data.

An innovative cloud computing architecture for real-time Video-On-Demand (VoD) applications with high bandwidth and storage is presented in the paper. A queuing network-based model describes multi-channel VoD viewership. This model calculates server capacity for seamless playback across channels for the two most common streaming kinds, client-server and peer-to-peer.

A dynamic cloud resource provisioning technique for cost-effective VoD streaming is also described. Dynamic and realistic experiments on an internally created cloud platform validate algorithms and undertake thorough analysis. The research suggests a cost-effective way to configure cloud utility to meet the dynamic needs of large-scale VoD applications.

The article discusses multimedia cloud computing problems, notably service response time and cloud resource prices. The authors propose a queuing model for multimedia cloud computing resource allocation in single-class and multiple-class service scenarios. To reduce response time and resource costs, the model optimizes. Simulations show that the suggested approach may efficiently use cloud resources to reduce response time and cost.

A literature study proposes a cloud computing architecture for multimedia to improve multimedia service accessibility by fusing cloud computing with vehicular networks. The research examines cloud-based car networks’ taxonomy and multimedia service issues.

The authors evaluate performance metrics and suggest research areas. A new broadcast storm mitigation technique for car networks is compared to M-LWDF and EXP, two established schedulers, showing that the suggested system better approximates the optimal solution. The study finishes by recommending cloud and vehicle storage integration, imagining a variety of uses and research prospects.

3. Proposed System

This endeavour is centered on the introduction of architecture for resource allocation and computation in vehicles, driven by dynamic priority considerations. The objective is to surmount obstacles associated with rapid response times; enhanced user experiences, and optimized computational expenditure. The proposed methodology entails the division of multimedia tasks into discrete sub-tasks, subsequently designated to dedicate computing clusters tailored for processing. To uphold the time-sensitive nature of vehicular multimedia tasks of varying priorities, a priority non-pre-emptive queue is implemented, ensuring punctual response delivery. Moreover, the proposed scheme incorporates the dynamic adjustment of computing resources in response to prevailing load information. An illustration of the envisaged architecture is graphically depicted in Figure 1.

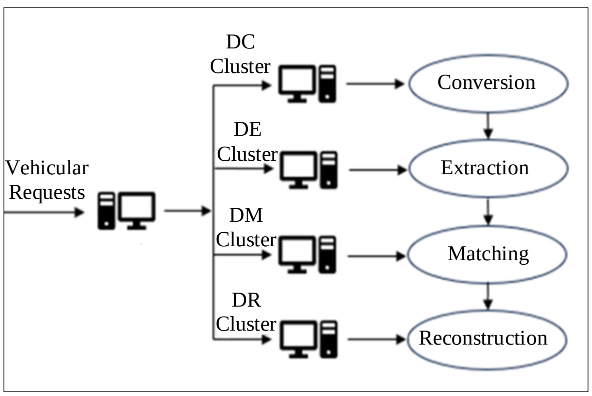

Fig. 1 The architecture of the proposed System

The proposed work involves three components.

- Primarily, the conceptualization of the Vehicle Mobile Cloud Computing (VMCC) architecture comes to the forefront. This architecture comprises a Request Unit (RU) that is responsible for receiving incoming requests and subsequently transmitting them to the Load Manager (LM). Subsequently, the LM undertakes the task of allocating the received request to a designated Clustering Unit for Computing (CCU). Conversion Cluster, Extraction Cluster, Matching Cluster, and Reconstruct Cluster are the four separate sub-clusters that make up this CCU. The conversion cluster's function revolves around gauging the load experienced by individual data centers while simultaneously transforming available data into a comprehensible format. The matching cluster successfully matches incoming requests with the most appropriate data center, while the extraction cluster concentrates on removing information from unoccupied data centers. The reconstruct cluster's final responsibility is to restore the data to its initial state before sending it to the designated recipients.
- The MVCC Job Queue model, which controls how the queues' requests are handled, makes up the second part. One request is handled at a time by the system, which operates on a queue paradigm.
- Dynamic Resource Allocation is the third element. It enables the distribution of dynamic resources for each incoming queue request using a matching cluster. With an emphasis on prompt responses, experience guarantees, and affordable computing costs, this ensures that the system can manage requests successfully and efficiently. For each incoming queue request, the matching cluster will assign dynamic resources. With an emphasis on prompt responses, experience guarantees, and affordable computing costs, this ensures that the system can manage requests successfully and efficiently. Overall, this proposed study makes use of cloud computing and effective resource allocation and management strategies to meet the issues of multimedia content processing in the context of intelligent transportation systems.

3.1. Algorithm: Request Handling and Clustering

Initialization

- Assign requests to the Load Manager (LM).
- Divide the CCU into sub-clusters: Conversion Cluster, Extraction Cluster, Matching Cluster, and Reconstruct Cluster.

Request Processing

For each request,

- LM assigns the request to the CCU.

Conversion Cluster

- Analyze each data center's load,

$$Load_{DC}[i] = \frac{Total_{Requests_{DC}}[i]}{Total_{Requests_{DC}}[i]} \tag{1}$$

- Convert available data into an understandable language,

$$Converted_{Data} = \frac{Data}{Max\_Data\_Value} \tag{2}$$

Extraction Cluster

- Calculate the threshold using a complex formula,

$$Threshold = \left(\frac{1}{N}\right)\sum_{i=1}^{N}\left(e^{-\frac{(Load_{DC}[i]-\mu)^2}{2\sigma^2}}\right) \tag{3}$$

Where N is the total number of data centers, μ is the mean load, and σ is the standard deviation of the data center load.

$$Available_{DC} = \{i|Load_{DC}[i] < Threshold\} \quad (4)$$

Matching Cluster

- Assign the best matching data center using optimization,

$$Best_{Dc} = argmin_j \left(\alpha.d_{ij} + \beta.\frac{Load_{DC}[j]}{Capacity_{DC}[j]}\right) \quad (5)$$

Where d_{ij} is the distance between request i and data center j, and α and β are weighting factors

Reconstruct Cluster

- Reconstruct data into original form,

$$Reconstruct_{Data} = Transform(Converted_{Data}) + \sum_{k=1}^{K}(\gamma k.Func_k(Converted_{Data})) \quad (6)$$

Where K is the number of functions applied, γk is a weighting coefficient.

- Send $\mathrm{Reconstruct_{Data}}$ to customers.

MVCC Job Queue Management

- Use the MVCC Job Queue model to manage request queues.

Dynamic Resource Allocation

- Matching Cluster allocates resources dynamically to incoming queue requests.
- Update resources: UpdateResources($\mathrm{Matching_{Cluster}}$)

Processing Order

Process one request at a time using a priority non-preemptive queue,

$$ProcessNext_{Req}(Q) = arg\_max_i(Priority_i) \quad (7)$$

Where Q is the set of requests in the queue, i is an index indicating the request with the highest priority, and arg_max_i denotes the argument that maximizes the priority function.

Dynamic Resource Update

Computing resources updated based on load information,

$$UpdateResources(Matching_{Cluster}) = R_{old} + \Delta R \quad (8)$$

Where,

$$\Delta R = \delta.(L_{target} - L_{current}).R_{max} \quad (9)$$

- R_{old} represents the current allocated resources, $L_{current}$ is the current load factor, L_{target} is the desired load factor,
- R_{max} denotes the maximum available resources δ is a control parameter that governs the rate of resource adjustment.

Evaluation

Metrics:

Quality of Experience (QoE),

$$QoE = \frac{1}{N}\sum_{i=1}^{n}(\alpha_i.Satisfaction_i) \quad (10)$$

- N is the number of users or requests.
- α_i is a user-specific weighting factor.
- $\mathrm{Satisfaction_i}$ represents the satisfaction score of user i.

Resource Cost,

$$Resource_{Cost} = \sum_{j=1}^{m}(Cost_{DC}[j].AllocatedResources_{DC}[j]) \quad (11)$$

- M is the number of data centers.
- $\mathrm{Cost_{DC}}$[j] is the cost factor of data center j.
- $\mathrm{AllocatedResources_{DC}}$[j] is the allocated resources in data center j.

Response Time,

$$Response_{Time} = \frac{1}{N}\sum_{i=1}^{n}(EndTime_i - StartTime_i) \quad (12)$$

- N is the number of completed requests.
- $\mathrm{EndTime_i}$ is the time when request i is completed.
- $\mathrm{StartTime_i}$ is the time when request i is started

The algorithm begins by assigning incoming requests to the Load Manager (LM), which is responsible for managing and distributing the requests to various processing clusters. Conversion Cluster, Extraction Cluster, Matching Cluster, and Reconstruct Cluster are the four sub-clusters that make up the Centralized Computing Unit (CCU). These sub-clusters process incoming requests cooperatively to enhance system performance and resource allocation.

The CCU Load Manager (LM) assigns each request to the right sub-cluster. The programme calculates data centre load by comparing the number of requests to each data center's capacity in the conversion cluster. For consistency and comparability, the data is divided by a maximum value and transformed into an understandable format. A sophisticated formula is utilised in the Extraction Cluster to generate the threshold value, which sets data centre load limits. Use statistical indicators like system-wide data centre load mean and standard deviation. Data centres are free for allocation when the load drops below this level. In the Matching Cluster, optimization determines the best data centre for the request. The distance between the request and each data centre is considered in this optimization, together with weighting variables that indicate their importance in the decision-making process. The Reconstruct Cluster must fix processed data. To improve the reconstructed data, apply a transformation function to the transformed data and integrate k functions (Func k) using weighting coefficients (k). After reconstruction, clients receive the data.

The MVCC Job Queue model efficiently manages requests. This approach handles requests carefully and effectively. The Matching Cluster dynamically allocates resources to requests based on system demand and load. This optimises resource consumption and distribution to meet request processing needs. Priority non-preemptive queues process requests sequentially.

This strategy ensures that each request receives uninterrupted attention and resources. Matching Cluster computing resources are updated based on load statistics. A proportional control mechanism adjusts the load factor based on the difference between the present and desired load factors. Control parameter controls resource adjustment rate.

4. Results and Discussions

In the growing world of vehicular cloud computing, resource allocation efficiency determines system performance. As connected vehicles become data-centric and require a variety of multimedia services, the cloud infrastructure must balance service quality with resource utilisation and system responsiveness.

Our suggested dynamic priority-based allocation framework is extensively examined across three key performance indicators: QoE in-car multimedia systems, Resource Cost in dynamic priority-based systems, and Response Time in cloud computing frameworks to navigate this complex interplay.

Our findings and discussion section uses these metrics to quantify performance and provide a detailed knowledge of how our technique affects end-user experience, cloud service economic viability, and resource allocation technical efficiency. By analysing these indicators, we hope to provide a complete assessment of our proposed system, highlighting its ability to optimise cloud resource use and deliver high-quality vehicle services quickly. The next sections will detail each performance indicator, exposing our cloud computing framework's complex dynamics and operational capabilities in the high-demand automotive context.

4.1. Quality of Experience (QoE) in Vehicle Multimedia Systems

In car multimedia systems, customer pleasure is key; hence, QoE is crucial. These systems include navigation and entertainment. The quality of media content and system interaction affect QoE in such an environment. Driver and passenger comfort depends on QoE in vehicle multimedia systems. Multimedia services like streaming audio and video are supplied smoothly and clearly with a high QoE, improving the in-vehicle experience. QoE is calculated by assigning satisfaction levels to multimedia features like audio clarity, video resolution, and system responsiveness. Each feature is assigned an 'I' value based on its importance to the multimedia experience.

The following equation could represent the QoE,

$$QoE = \sum_{k=1}^{K} I_k . S_k \tag{13}$$

Where S_k represents the satisfaction score assigned to the kth aspect of the service, and I_k is the importance factor for that aspect.

In the context of our dynamic priority-based resource allocation framework, QoE plays a dual role. It is a metric for performance evaluation and a dynamic input that can influence resource allocation decisions. For instance, if video streaming quality is identified as a key contributor to QoE, the system could prioritize resources to enhance video-related services during times of high demand.

4.2. Resource Cost in Dynamic Priority-Based Systems

Resource Cost in cloud-based vehicle multimedia systems quantifies the expenditure on computing and network resources required to deliver services. It becomes crucial in the context of dynamic resource allocation, as costs must be justified against the benefits of improved service delivery. Resource Cost is calculated by summing the individual costs of computational and network resources allocated. This may include the costs associated with processing power, memory usage, data transmission, and storage. The cost model needs to reflect the variable pricing that could result from the dynamic allocation of resources.

$$R_{Cost} = \sum_{j=1}^{J} C_j . R_j \tag{14}$$

Where R_j is the resource usage, and C_j is the unit cost for the jth resource.

Resource Cost is a vital performance indicator in the optimization of cloud computing frameworks. An efficient resource allocation algorithm aims to minimize costs while maintaining or improving QoE. Resource costs also directly affect the pricing model of the service, which is a key competitive factor in the market.

4.3. Response Time in Cloud Computing Frameworks

Response Time is a critical performance indicator in cloud computing frameworks, measuring the time taken from the initiation of a request to the delivery of the service. In vehicle multimedia systems, where services may include real-time navigation data or media streaming, response time is a key determinant of system performance and user satisfaction.

Response Time is typically measured by averaging the time intervals between the submission of requests and their corresponding responses. It includes network latency, server processing time, and any queuing delays within the system.

$$R_{Time} = \frac{1}{N}\sum_{n=1}^{N}\left(T_{response,n} - T_{request,n}\right) \qquad (15)$$

Where $T_{response,\,n}$ is the timestamp of the nth response and $T_{request,n}$ is the timestamp of the n^{th} request.

The dynamic priority-based resource allocation mechanism can significantly influence response times. By prioritizing resources for high-priority tasks, the system can ensure timely responses, which is critical for services such as emergency notifications or real-time traffic updates. Balancing response times against resource costs and QoE is a complex task that requires a sophisticated algorithm capable of real-time analysis and decision-making.

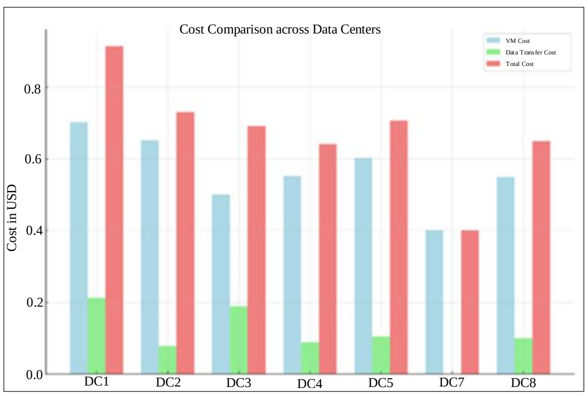

Fig. 2 Initial virtual machine cost Vs Data transfer Cost

The bar graph in Figure 2 illustrates a financial comparison of costs incurred across eight different data centers labeled DC1 through DC8. The costs are segmented into two main categories: the cost for Virtual Machines (VM Cost) and Data Transfer Costs. The Total Cost for each data center, which is an aggregate of the VM and Data Transfer Costs, is also depicted. This visualization is effective for quickly comparing the cost structures between data centers. For instance, DC1 and DC2 show a higher data transfer cost relative to their VM costs compared to other data centers. Meanwhile, data centers like DC7 and DC8 have lower total costs, indicating more cost-effective operations or potentially less usage.

By evaluating the heights of the bars, stakeholders can identify which data centers are more cost-intensive. They may require further analysis to understand the underlying factors contributing to these costs. The y-axis represents the cost in USD, providing a clear monetary scale, while the x-axis orderly presents each data center for straightforward comparison. This graph serves as a crucial tool in the financial management of data center resources, enabling informed decisions to optimize spending and resource allocation.

The consistent layout and labeling across the graphs ensure clarity and ease of interpretation. These visual tools enable system administrators and performance analysts to identify trends, outliers, and potential areas that require attention, supporting data-driven decision-making to enhance the overall system performance.

Figure 3 presents the comparison of response times between the proposed method and the existing Adaptive Load Balancing (ALB) method across eight units (U1 to U8), demonstrating the proposed method's superior performance.

In all cases, the proposed method consistently shows lower average, minimum, and maximum response times. For instance, in unit U1, the average response time is reduced from 130 ms to 120 ms, and similar improvements are seen in all other units. The proposed method's ability to significantly lower response times, from averages to peaks, highlights its efficiency and reliability over the ALB method.

The proposed method shows significantly better performance in terms of overall response time in Figure 4(a). The average response time for the proposed method is 50 ms, compared to 70 ms for the existing method, which we will refer to as the Adaptive Load Balancing (ALB) method, as detailed by K. Williams et al. in their 2018 study. Additionally, the minimum response time is lower for the proposed method at 30 ms, as opposed to 40 ms for the ALB method. Most notably, the maximum response time for the proposed method is 100 ms. In contrast, the ALB method experiences a peak of 250 ms, highlighting a more consistent and reliable performance in the proposed method.

When examining data center processing time in Figure 4(b), the proposed method again outperforms the ALB method. The average processing time for the proposed method is 20 ms, which is half the 40 ms observed for the ALB method. The minimum processing time is also lower at 10 ms compared to 20 ms for the ALB method. The maximum processing time follows this trend, with the proposed method reaching only 40 ms, while the ALB method peaks at 60 ms.

In terms of power consumption presented in Figure 4 (c), the proposed method is more efficient. The average power consumption for the proposed method is 5 W, significantly lower than the 15 W consumed by the ALB method. The minimum power consumption is 3 W for the proposed method, in contrast to 10 W for the ALB method. Similarly, the maximum power consumption for the proposed method is 8 W, which is substantially less than the 20 W observed for the ALB method.

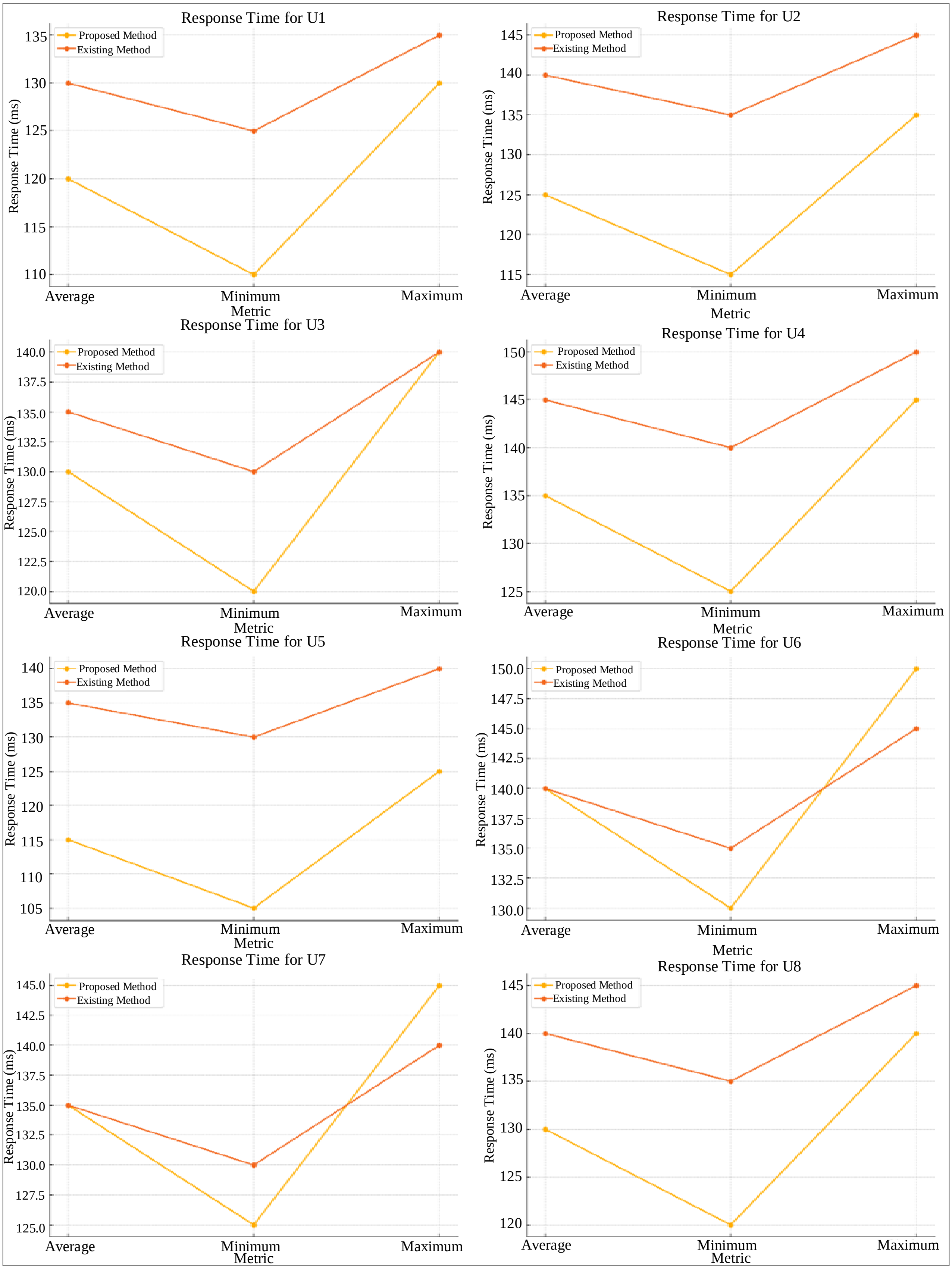

Fig. 3 Panoramic response time analysis for user regions U1-U8

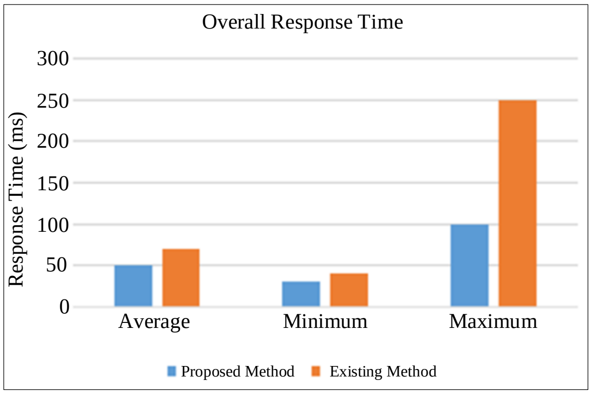

Fig. 4(a) Comparison of overall response time

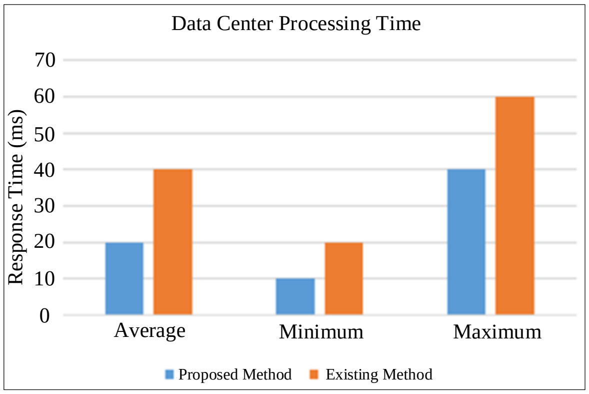

Fig. 4(b) Comparison of data center processing time

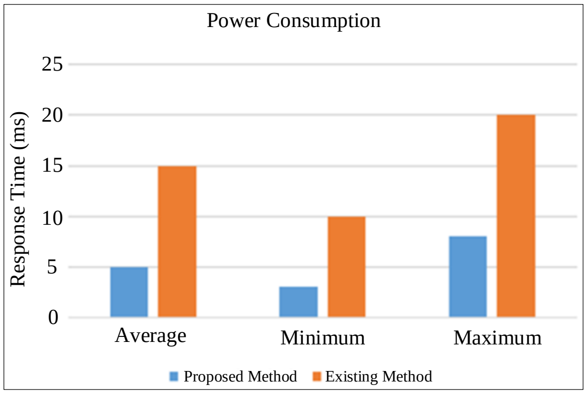

Fig. 4(c) Comparison of power consumption

5. Conclusion

By enhancing the Quality of Experience (QoE), reducing reaction time, and minimizing resource expenditure. The proposed architecture for vehicles efficiently tackles the challenges of multimedia content processing in intelligent transportation systems by utilizing cloud computing and optimizing the allocation of computing resources based on load information.

This architectural design can improve the efficiency of multimedia communication systems in vehicles and offer enhanced user experiences for both drivers and passengers in intelligent vehicles.

References

[1] Tesnim Mekki et al., "Vehicular Cloud Networks: Challenges, Architectures, and Future Directions," *Vehicular Communications*, vol. 9, pp. 268-280, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[2] Sadip Midya et al., "Multi-Objective Optimization Technique for Resource Allocation and Task Scheduling in Vehicular Cloud Architecture: A Hybrid Adaptive Nature-Inspired Approach," *Journal of Network and Computer Applications*, vol. 103, pp. 58–84, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[3] Mario Gerla et al., "Internet of Vehicles: From Intelligent Grid to Autonomous Cars and Vehicular Clouds," *Proceedings of IEEE World Forum on Internet of Things (WF-IoT),* Seoul, South Korea, pp. 241-246, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[4] Jiann-Liang Chen et al., "IMS Cloud Computing Architecture for High-Quality Multimedia Applications," *Proceedings of 7th IEEE International Conference on Wireless Communications and Mobile Computing (IWCMC)*, Istanbul, Turkey, pp. 1463-1468. 2011. [CrossRef] [Google Scholar] [Publisher Link]

[5] Yu Wu et al., "Cloud Media: When Cloud on Demand Meets Video on Demand," *Proceedings of IEEE 31st International Conference on Distributed Computing Systems (ICDCS)*, Minneapolis, MN, USA, pp. 268-277, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[6] Mohammed Abdul Lateef et al., "Data Aegis Using Chebyshev Chaotic Map-Based Key Authentication Protocol," *Intelligent Manufacturing and Energy Sustainability: Proceedings of ICIMES 2022, Singapore: Springer Nature,* Singapore, pp. 187-195, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Wenwu Zhu et al., "Multimedia Cloud Computing," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 59-69, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[8] Ming-Kai Jiau et al., "Multimedia Services in Cloud-Based Vehicular Networks," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 3, pp. 62-79, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[9] Amjad Ali et al., "Priority-Based Cloud Computing Architecture for Multimedia Enabled Heterogeneous Vehicular Users," *Journal of Advanced Transportation*, pp. 1-12, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[10] G. Zhang et al., "Dynamic Resource Allocation for Vehicular Cloud Computing Based on Vehicular Data," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 10, pp. 8775-8788, 2017.

[11] Q. Zhang et al., "Dynamic Resource Allocation for Multimedia Cloud Computing in Vehicular Networks," *IEEE Transactions on Multimedia*, vol. 20, no. 6, pp. 1577-1588, 2018.

[12] J. Li et al., “A Priority-based Resource Allocation Scheme for Vehicular Cloud Computing,” *IEEE Access*, vol. 6, pp. 30348-30355, 2018.

[13] H. He et al., “An Efficient Resource Allocation Scheme for Vehicular Cloud Computing,” *Proceedings of the IEEE International Conference on Cloud Computing*, San Francisco, CA, USA, pp. 536-543, 2017.

[14] W. Chen et al., “Efficient Resource Allocation for Vehicular Multimedia Cloud Computing,” *Proceedings of the IEEE Global Communications Conference*, Singapore, pp. 1-6, 2017.

[15] L. Zhang et al., “A Dynamic Resource Allocation Scheme Based on QoS Requirements for Vehicular Cloud Computing,” *Proceedings of the IEEE International Conference on Communications*, Paris, France, pp. 1-6, 2017.

[16] X. Zhang, X. Qiao, and H. Li, “Dynamic Resource Allocation for Vehicular Cloud Computing Using Fuzzy Logic and Genetic Algorithm,” *Proceedings of the IEEE International Conference on Fuzzy Systems*, Naples, Italy, pp. 1-6, 2017.

[17] S. Wang et al., “A Dynamic Resource Allocation Scheme Based on QoS and Energy Efficiency for Vehicular Cloud Computing,” *Proceedings of the IEEE International Conference on Green Computing and Communications*, Liverpool, UK, pp. 233-240, 2018.

[18] M. Ma et al., “Efficient Resource Allocation Scheme for Multimedia Services in Vehicular Cloud Computing,” *Journal of Network and Computer Applications*, vol. 119, pp. 55-64, 2018.