Collaborative Video Caching Strategy Based on Delay and Energy for Software-Defined Hybrid VLC and mmWave Networks

Vuai A. Kombo *, **Natalia Kryvinska** **, **Mohammed Abdalsalam** ***, **Adnan Ali** ****

* *School of Computer Science and Artificial Intelligence, Wuhan University of Technology, Wuhan 430063, Hubei, PR China, (e-mail: vuaiali15@gmail.com)*
** *Department of Information Management and Business Systems, Head Faculty of Management, Comenius University Bratislava, Odbojárov 10, 82005 Bratislava 25, Slovakia, (e-mail: natalia.kryvinska@uniba.sk)*
*** *Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gudanski, Poland, (e-mail: mohammed.abdalsalam@pg.edu.pl)*
**** *School of Computer Science and Multimedia Studies, The State University of Zanzibar, Tunguu, Zanzibar, PR Tanzania, (e-mail: ali.adnan@suza.ac.tz)*

Abstract: The integration of 5G networks has revolutionized wireless communication with higher data rates and ultra-low delay, enabling new services and applications, especially in mobile edge computing (MEC). MEC aims to minimize delay and enhance user experience by placing computational power and storage capabilities closer to end-users. A key challenge in MEC is efficiently managing network resources, particularly in environments with hybrid Radio Access Technologies (RATs). This paper explores a collaborative video caching strategy focused on delay and energy efficiency for software-defined hybrid Visible Light Communication (VLC) and millimeter Wave (mmWave) networks. These technologies offer high data rates but face challenges like varying channel conditions and blockages. The paper introduces a framework using Software-Defined Networking (SDN) for dynamic resource management and optimized content caching. It integrates a Long Short-Term Memory (LSTM) network to forecast user preferences and use a collaborative communication framework to improve edge caching. Optimization problems are formulated to reduce delay and energy consumption, considering limited storage at small-cell base stations (SBSs). The proposed algorithm significantly improves delay, energy consumption, and cache hit rates.

Keywords: Mobile edge computing, SDN, Edge caching, visible light communication, millimeter Wave.

1. INTRODUCTION

The advent of fifth-generation (5G) networks has introduced transformative advancements in wireless communication, characterized by significantly enhanced data rates, ultra-low delay, and extensive connectivity. These improvements have spurred the development of innovative applications and services, particularly within the domain of MEC (Tomaszewski et al. (2020)). MEC seeks to bring computational and storage capabilities closer to the end users, thereby minimizing delay and improving overall user experience. Despite these advancements, managing network resources efficiently in MEC environments remains a critical challenge, particularly in scenarios where hybrid Radio Access Technologies (RATs) coexist, necessitating sophisticated solutions to ensure seamless operation (Qin et al. (2020)).

Hybrid RAT environments, which integrate multiple wireless technologies such as VLC and mmWave, have emerged as promising solutions for enhancing indoor wireless communications. They offer high data rates and low interference but face deployment challenges like variable channel conditions, blockages (Feng et al. (2018)) and limited storage capacities at SBSs. These challenges underscore the need for an effective strategy that can dynamically manage resources and optimize content delivery to enhance user experience.

This paper addresses these challenges by proposing a collaborative video caching strategy designed to optimize delay and energy efficiency for software-defined hybrid

VLC and mmWave networks. This approach leverages the capabilities of SDN to dynamically manage network resources and optimize content caching. By integrating LSTM network, the user preferences are effectively predicted, allowing for better caching decisions.

The optimization problems are formulated aimed at minimizing content delivery delay and energy consumption, taking into account the limited storage capacities at base stations. The proposed solution includes a collaborative communication framework for content sharing among edge nodes, further enhancing the caching efficiency. Through extensive simulations, significant improvements in delay reduction, energy efficiency, and cache hit rates were demonstrated. While the solution has been validated in a simulated environment, it is designed to be applicable in real-world scenarios, particularly in large-scale 5G deployments.

This comprehensive strategy offers a robust solution for enhancing content delivery efficiency in hybrid VLC and mmWave networks, paving the way for more resilient and user-centric 5G indoor communication systems. By addressing the interplay between various wireless technologies and leveraging advanced prediction and optimization techniques, this research contributes to the ongoing efforts to optimize MEC environments and ensure superior user experiences.

By addressing the integration of hybrid VLC and mmWave technologies with SDN, this work offers a comprehensive solution to the challenges of indoor 5G networks. The proposed collaborative caching strategy not only enhances content delivery efficiency but also ensures that user demands are met promptly and energy-efficiently. This work lays the foundation for more advanced indoor communication systems in the 5G era and beyond.

The key contributions of this paper are:

(1) The hybrid RAT system integrating VLC and mmWave technologies with SDN for enhanced indoor wireless communication is introduced.
(2) An LSTM network is used to capture short-term user dynamics and predict user preferences.
(3) The optimization problem for minimizing content delivery delay and energy consumption, considering the limited storage capacities at SBSs for content placement scenarios, is formulated. Collaborative edge caching algorithm to configure content placement parameters is developed.
(4) The theoretical findings are validated by numerical results, and the performance gains for the proposed algorithms are demonstrated.

The rest of the paper is organized as follows: Section 2 reviews the related works. The hybrid system model is introduced in Section 3, detailing the channel models and deployment scenarios. Section 4 presents the problem formulation, outlining the optimization objectives and constraints and the collaborative caching strategy focusing on delay and energy consumption. Section 5 provides a performance evaluation and a discussion of the proposed methods. The final section, Section 6, presents the conclusions and outlines future research directions.

2. RELATED WORKS

Recent advancements in cooperative edge caching approaches have significantly improved the allocation of resources and content delivery within wireless networks. Numerous studies have been made to lower delay and energy consumption, all while improving cache hit rates.

Wang et al. (2018), introduced a heuristic cooperative cache replacement strategy for zone-based content caching in extensive RATs networks, with edge server storage divided to cache locally and globally popular content separately. A greedy cooperative caching algorithm was developed by Zhang and Mao (2019) that proactively replaces cached layered video files during off-peak hours, resulting in reduced overall transmission delay. Xia et al. (2020) introduced CEDC-O, an online algorithm which is based on Lyapunov optimization, aimed at minimizing costs.

A content collaborative caching mechanism optimizing download delay and energy consumption was introduced by Rui et al. (2022), specifically for communication network field maintenance scenarios. This mechanism proactively caches maintenance-related content at the network edge, establishing models for average download delay from the user's perspective and energy consumption from the caching and distribution perspective. Similarly Liu et al. (2022) addressed the joint problem of minimizing delay and energy in MEC-assisted adaptive video streaming networks by optimizing caching, computing, and power allocation, deriving a closed-form optimal solution for the quasi-convex power allocation problem.

However, these studies assume the advanced knowledge of content popularity, which is often impractical. Some researchers have attempted to predict content popularity using historical data. Zhao et al. (2018) developed a method for predicting user preferences using request history and social information, optimizing cache replacement with a low-complexity heuristic algorithm. To learn popularity profiles and improve content placement using a greedy algorithm, caching content with better channel qualities, a Bayesian learning algorithm was proposed by Nie et al. (2020). Chen et al. (2020) introduced an edge cooperative cache combining neural collaborative filtering and a greedy algorithm to accurately forecast content popularity and significantly reduce transmission delay.

Baccour et al. (2020) introduced the edge framework called CE-D2D, combining collaborative MEC clusters with users' caching capacities allowing various helpers, including user devices and MEC servers, to cache and offload video segments collaboratively. This approach enhances caching efficiency for large multimedia contents and is formulated as an ILP problem to minimize resource allocation costs incorporating a proactive caching strategy along with an online helper selection mechanism for real-time scenarios.

Khanal et al. (2021) studied proactive content caching at the edge of the network and proposed the distributed collaborative learning mechanism named DCoL. This mechanism uses an LSTM model and distributed collaborative filtering to leverage local content popularity information for proactive caching across distributed edge nodes. A non-

parametric algorithm was developed, utilizing regional database feedback to optimize local content caching strategy.

Despite these advancements, previous works have not fully addressed integrating hybrid RAT technologies with SDN for efficient content delivery in MEC environments. This research aims to fill this gap by proposing a novel framework that leverages SDN to dynamically manage resources, content delivery and optimize content caching strategies, thereby reducing delay and energy consumption and improving cache hit rates in hybrid VLC and mmWave networks.

3. SYSTEM MODEL

3.1 Network Model

The architecture depicted in Figure 1 comprises of Macro Base Station (MBS), cache-enabled SBS utilizing SDN, VLC, and mmWave technologies to communicate with User Equipments (UEs) within their coverage area. The edge cache service includes components such as the Open Flow Switch (OFS), SDN controller, cache application, channel selection application, cache directory, and cache server. In this paper, a software-defined hybrid VLC and mmWave access network scenario is considered, wherein numerous users are dispersed within an open indoor office area. The ceiling is equipped with several remote radio light heads (RRLHs) that house mmWave beam and Light Emitting Diode (LED) modules. Each module functions independently as high-speed Access Points (APs), which cover specific confined areas. All APs are connected to the OFS, which manages data coordination with the SDN controller.

When a user requests content, the OFS sends the initial packet to the SDN controller as a packet-in event. The cache application then checks the edge cache for the content. If the content is found, the SDN controller directs the OFS to forward the request to the local cache server. If not, the cache application looks in the cache directory for the content in a neighboring cache server. If the content is found, the SDN controller directs the OFS to forward the flow to the correct cache server. If neither the edge cache nor the neighboring cache has the content, the SDN controller instructs the OFS to send the request to the MBS. Once the content is fetched, it is transferred to the local node if obtained from a neighbouring cache server or MBS. The channel selection application then monitors the traffic on the VLC and mmWave interfaces. Depending on whether the VLC data rate exceeds a set threshold, the system responds via either the VLC AP if the threshold is met or the mmWave AP if it is not. This ensures efficient content delivery by leveraging the strengths of both VLC and mmWave technologies. The flowchart in Figure 2 outlines a process for handling user requests in a hybrid VLC and mmWave network.

3.2 Channel Model

Assuming a user labelled as j positioned at coordinates (x_j, y_j) and an RRLH lamp labelled as i at coordinates

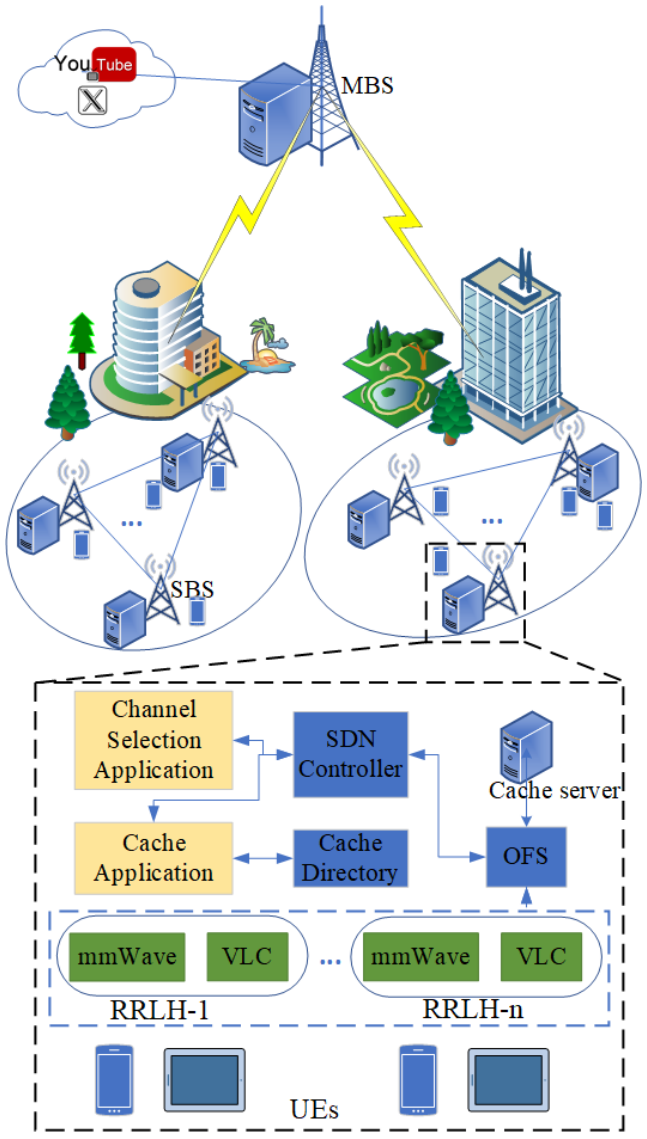


Fig. 1. System model

(x_i, y_i), the Euclidean space depicts the relative distance (d_{ij}) between points i and j as a straight line as:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{1}$$

According to (Kahn and Barry (1997), Fath and Haas (2012)), a LoS channel optical gain for the VLC AP is defined as

$$h_{ij}^{\mathrm{VLC}} = \begin{cases} \frac{(m+1)\zeta}{d_{ij}^2} \cos^m(\phi) \cos(\vartheta) & 0 \le \vartheta \le \vartheta_{\mathrm{FOV}} \\ 0, & \vartheta > \vartheta_{\mathrm{FOV}} \end{cases} \tag{2}$$

where m represents the order of Lambertian emission determined by semi-angle at half-power of RRLH lamp, and is calculated using the following expression:

$$m = \frac{-\ln(2)}{\ln(\cos(\phi_{1/2}))} \tag{3}$$

d_{ij} denotes the distance between the RRLH and the UE, $\zeta = A_d R_d G_c(\vartheta) G_f(\vartheta)/(2\pi)$ is a constant for the VLC system. ϕ represents an angle of radiance between RRLH lamp and UE, ϑ denotes the angle of light incident to the receiving user device surface. $\phi_{1/2}$ represents half-power angle of the lamp, ϑ_{FOV} denotes the receiver Field of

View (FOV) semi-angle. A_d denotes the physical area of receiver's photo-diode, R_d represents the photo-detector's responsivity. $G_c(\vartheta)$ and $G_f(\vartheta)$ are optical filter gain and concentrator gain respectively. $G_c(\vartheta)$ can be calculated as

$$G_c(\vartheta) = \begin{cases} \frac{k^2}{\sin^2 \vartheta_{\text{FOV}}}, & 0 \leq \vartheta \leq \vartheta_{\text{FOV}} \\ 0, & \vartheta > \vartheta_{\text{FOV}} \end{cases} \tag{4}$$

where k denotes the refractive index.

For any user j connected to a VLC AP i, the signal-to-noise ratio (SNR) can be calculated using the following expression:

$$\Gamma_{i,j}^{VLC} = \frac{\left(P_{\text{t}} R_{\text{d}} h_{i,j}^{\text{VLC}}\right)^2}{\gamma^2 N_o^{\text{VLC}} B^{\text{VLC}}} \tag{5}$$

where P_t denotes the transmitted optical power, R_d represents the detector responsivity, and γ represents the optical to electric power conversion coefficient at the receiver. N_o^{VLC} denotes the power spectral density for the shot and thermal noise at the photo-diode, and B^{VLC} denotes the VLC AP system bandwidth.

In reality, line-of-sight (LOS) blocking events occur because of obstructions between APs and UE links. For VLC AP, the received power of light is significantly reduced without LOS, potentially leading to poor reception quality. This paper assumes that non-LOS (NLOS) VLC transmissions often fail and focuses on direct LOS transmissions. Conversely, NLOS transmissions in mmWave systems are still considered capable of achieving adequate reception quality.

The function for the estimated SNR in decibels (dB) for mmWave transmission link between user j and mmWave AP n is expressed as

$$\Gamma_{n,j}^{mmWave} = P_r - P_{noise} \tag{6}$$

where P_r is the received signal power while P_{noise} denotes the noise power. P_r depends on the transmitted power, path loss, antenna gains, and propagation effects such as reflection, diffraction, and scattering. P_r can be estimated as

$$P_r = P_{nj} + G_t + G_j - L_p \tag{7}$$

where P_{nj} is the mmWave AP total transmitted power in dBm, G_t is the antenna gain of the transmitter and G_j is the receiver antenna gain, in dBi and are calculated by $G = 20log_{10}(\frac{\pi l}{\lambda})$ where l is the antenna length and λ is the exploited frequency band. L_p is the path loss in Floating-Intercept(FI) model which has been commonly applied to wireless applications scenarios for characterizing the path loss (Molisch (2012), Rubio et al. (2019)) and is given by:

$$L_p[dB] = \beta + 10\alpha \log_{10}(d) + X_\sigma^{FI} \tag{8}$$

where β is the the floating-intercept parameter (offset term) in dB, α depicts the path loss exponent which is the slope of the path loss line, X_σ^{FI} is the Gaussian random variable(in dB) with shadowing variance σ^2 describing the fluctuations of large-scale signal about the mean path loss over distance. In this paper, 28 GHz and 73 GHz frequency bands are selected since they fit the NLOS and LOS transmission links.

P_{noise} is determined by the thermal noise and noise figure of the receiver. The total noise power can be given by

$$P_{noise} = 10 \log_{10} B^{mmWave} + N_o + N_f \tag{9}$$

where B^{mmWave} is the bandwidth in Hz, N_o and N_f are noise power density and noise figure at the AP respectively.

Substituting equation (7) and (9) into (6), SNR function in dB for (n, j) link between user j and mmWave AP n can be given by

$$\Gamma_{n,j}^{mmWave} = P_{nj} + G_j - L_p - (10 \log_{10} B^{mmWave} + N_o + N_f) \tag{10}$$

After computing the SNR for both channels, the next step is to determine which AP should be used to serve the user request. This will be achieved by determining which AP has the maximum data transfer rate among all VLC and mmWave APs to serve the user as shown in Algorithm 1. Initially, the VLC AP and mmWave AP with the highest communication link data rate among all the APs is identified, as expressed in (11) and (12) respectively. Then based on Shannon's capacity formula, the data transmission rate for every UE j at VLC AP and mmWave AP is calculated as shown in (13) and (15) respectively. This work assumes that each AP serves only one UE at a time.

$$\chi_j^{VLC} = \underset{i \in L^{VLC}}{\arg\max}\, \Gamma_{i,j}^{VLC} \tag{11}$$

where $L^{VLC} = \{i | i \in [1, I], i \in I\}$ represents a set of all VLC APs.

$$\chi_j^{mmWave} = \underset{n \in L^{mmWave}}{\arg\max}\, \Gamma_{n,j}^{mmWave} \tag{12}$$

where $L^{mmWave} = \{n | n \in [1, N], n \in N\}$ denotes a set of all mmWave APs.

The transmission rate for VLC AP according to Shannon's capacity formula, is expressed as:

$$R_j^{VLC} = B^{VLC}.\log_2(1 + \Gamma_{ij}^{VLC}) \tag{13}$$

and that of mmWave APs as

$$R_j^{mmWave} = B^{mmWave}.\log_2(1 + \Gamma_{ij}^{mmWave}) \tag{14}$$

The transmission rate to download the content from the local VLC-gNB node i to the UE j is given by:

$$R_l = \mu R_j^{VLC} + (1 - \mu) R_j^{mmWave} \tag{15}$$

where μ is a binary number representing channel allocation, μ=1 if the user is allocated the VLC channel and μ=0 otherwise.

Using the data rate threshold ϵ, the users achieving the highest communication link data rate are assigned to VLC APs. The optimal AP allocation for each user j can be determined by looking for the highest Γ_j^* for each user among the two RATs given by

$$\chi_j = \begin{cases} \chi_j^{VLC}, \text{if} & R_j^{VLC} \geq \epsilon \\ \chi_j^{mmWave}, & \text{otherwise} \end{cases} \tag{16}$$

While the primary focus of this paper is the video caching strategy, the channel selection algorithm plays an essential role in ensuring efficient content delivery. The system dynamically selects between VLC and mmWave based on real-time SNRs and Shannon's Theoretical Channel Capacity. This ensures that the video content is delivered via the most optimal channel, minimizing delay and improving user experience. By integrating channel selection into the caching strategy, the transmission delays can further be reduced when users request the content.

The channel selection in (16) is implemented by using the Algorithm 1 in the order of $O((I+J)\times J)$ with the big-O notation.

Table 1. Notations

Symbol	Meaning
h_{ij}^{VLC}	Optical channel gain for the VLC AP
m	Order of Lambertian emission
ϑ	Angle of light incident
ϑ_{FOV}	Receiver FOV semi-angle
ϕ	Angle of radiance between lamp and UE
A_d	Physical area of the receiver's photo-diode
R_d	Photo-detector's responsivity
$G_c(\vartheta)$	Optical filter gain
$G_f(\vartheta)$	Concentrator gain
$\phi_{1/2}$	Half-power angle of the lamp
k	Refractive index
B^{VLC}	VLC AP System bandwidth
$\Gamma_{i,j}^{VLC}$	SNR of the VLC channel
$\Gamma_{n,j}^{mmWave}$	SNR of the mmWave channel
γ	optical to electric power conversion coefficient
N_o^{VLC}	Power spectral density

Algorithm 1 Channel selection algorithm

Input: input user coordinates $\{x_j, y_j\}$
input position of VLC and mmWave AP $\{x_i, y_i\}, \{x_n, y_n\}, \forall i \in I, n \in N$
input VLC and mmWave parameters $\phi, \vartheta, \vartheta_{\mathrm{FOV}}, \phi_{1/2}, \zeta$
for all user $j = 1$ to J **do**
  compute d_{ij}, d_{nj} according to (1)
  compute h_{ij}^{VLC} according to (2)
  compute $\Gamma_{ij}^{\mathrm{VLC}}$ and $\Gamma_{nj}^{\mathrm{mmWave}}$ using (5) and (6) respectively
  compute $R_j\left(\Gamma_{ij}^{\mathrm{VLC}}, \Gamma_{ij}^{\mathrm{mmWave}}\right)$ using the results of Step 7
  compute $\chi_j^{VLC}, \chi_j^{mmWave}$ using (11), and (12), respectively
  if $R_j \geq \epsilon$ **then**
    check whether the VLC AP is available then
    allocate user j to VLC AP $\chi_j^* = \chi_j^{VLC}$ according to (16)
  else
    allocate user j to mmWave AP $\chi_j^* = \chi_j^{mmWave}$ according to (16)
  end if
end for

4. THE COOPERATIVE EDGE CACHING

4.1 User content reading behaviour prediction using an LSTM

Predicting user request behavior in the next interval is vital for the effective application of the caching strategy. Due to its effectiveness in processing sequential data and its successful application in several domains like machine translation (Sutskever et al. (2014) and speech recognition (Graves and Jaitly (2014)). The LSTM network is highly suitable for addressing the user behaviour prediction problem. The LSTM network input, train and output are described below:

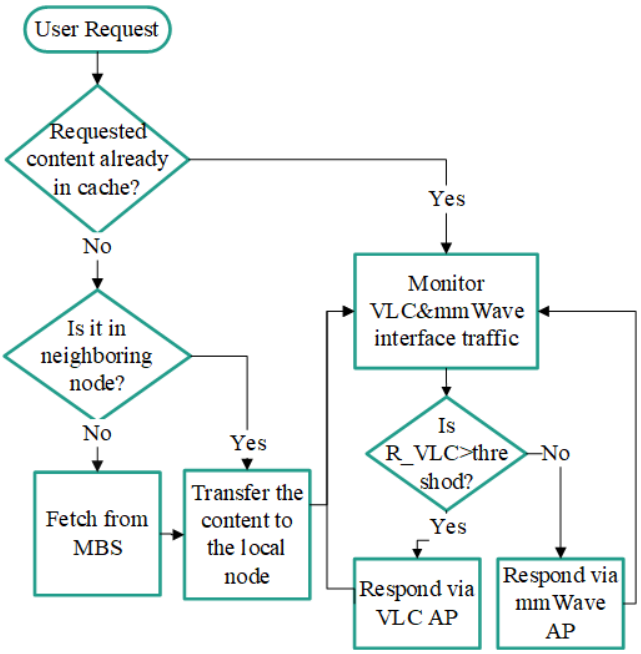


Fig. 2. Edge cache operation flowchart

Input: At each time step t, a user's interaction with the video was captured using the following features:

$$x_t = [\mathrm{rawYaw}_t,\ \mathrm{rawPitch}_t,\ \mathrm{rawRoll}_t,\ \ldots]$$

These features cover head movement features, tile viewing behavior features, temporal features, user engagement features, and video metadata features. The head movement features include raw yaw, pitch, roll values as well as calibrated versions of these angles. The tile viewing behavior features include most viewed tile, tile change frequency, and percentage focus on the most viewed tile. The temporal features include session duration and average time interval. User engagement features include engagement score, rewatch frequency and interaction intensity and video metadata features include genre, title and size. For each user session, a sequence of such feature vectors was generated as $X = \{x_1, x_2, \ldots, x_T\}$ where T is the sequence length.

Train: The input data was loaded as input elements for the LSTM model and processed through LSTM model gates comprising of Forget Gate, Input Gate, Cell State Update and Output Gate as follows:

$$\begin{cases} f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\ C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \\ o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t = o_t \odot \tanh(C_t) \end{cases} \tag{17}$$

The model was trained using Backpropagation Through Time, where the gradients were computed across all time steps in the input sequence to update the LSTM parameters. The model was optimized using the Adam optimizer, which dynamically adjusts the learning rate based on the gradients.

After processing the input sequence, the final hidden state h_T was passed to a softmax layer, which outputs

a probability distribution over the possible video content to be requested:

$$\hat{y} = (W_h \cdot h_T + b_h) \tag{18}$$

Where W_h is the weight matrix connecting the hidden state to the output layer and b_h is the bias term for the output layer.

Output: The predicted video content requested by the user was obtained using the categorical crossentropy loss function as follows.

$$f_j = -\sum_{i=1}^{K} y_i \log(\hat{y}_i) \tag{19}$$

where y_i is the true label for the video requested and $\hat{y}_i$ is the predicted probability for that video obtained from the prediction layer.

The pseudocode for a strategy based on an LSTM model to predict user request behavior is shown in Algorithm 2 where the possible video content to be requested is identified.

Algorithm 2 User preferences prediction algorithm using LSTM

Input: Historical playback information of a user
Output: Predicted video content f_j

1: **for** each user u_j **do**
2: load and process the input data as input elements for the LSTM model
3: split the dataset into training, validation and test sets
4: feed the data to the LSTM model
5: using the model to forecast video content requested for the next time slot
6: **end for**
7: **return** predicted video content

4.2 Cooperative caching scheme

Delay Model When a mobile user requests video content and it is available locally, it will be delivered directly to the user at a transmission rate of R_l. If the content requested is not available in the local edge server, it will be fetched from the neighbouring node at a transmission rate of R_{nl}. If the content is found in the neighbouring node, it will be delivered to the user. Otherwise, the content will be downloaded from the MBS with R_{ml} transmission rate. In this paper, it is assumed that $R_l > R_{nl} > R_{ml}$.

Let e_i denotes the ith edge server, c_{max} represents the maximum storage capacity of the cache server e_i, and $i \in E$, $E = \{1, 2, \ldots, M_e\}$. u_j represents the jth user and $j \in U$, $U = \{1, 2, \ldots, N_u\}$. $F = \{f_1, f_2, \ldots, f_j, \ldots, f_N\}$ denotes the set of video contents requested by all users resulting from the user content preference prediction using LSTM method described in Sectio 4. In this paper, it is assumed that the end-user u_j can request one video content f_j at a time.

When a user within the coverage area of the SBS which caches video content f_j with size S requested by the user u_j, transmission delay D_l is directly proportional to content size S divided by the transmission rate R_l; if the content is not cached locally but is available in a neighbouring node, the transmission delay D_{nl} includes the delay in transmitting the file from the neighbouring node to the local node, expressed as S divided by the transmission rate R_{nl}, added to the local transmission delay D_l. If any neighbouring nodes do not cache the content, the transmission delay D_{ml} involves fetching the content from the MBS, calculated as the file size S divided by the transmission rate R_{ml}, plus the local transmission delay D_l. Given $q_{ij} \in \{0, 1\}$ indicating whether local edge server has the content requested, if $q_{ij} = 1$, the local edge server caches f_j and $q_{ij} = 0$ otherwise. And $q_{nj} \in \{0, 1\}$ indicating whether the neighbouring edge server has the content or not, if $q_{nj} = 1$, then the neighbouring edge server caches the file contents and $q_{nj} = 0$ otherwise. Thus, the time taken for the user u_j to access the requested video content f_j is mathematically expressed as:

$$\begin{cases} D_l = \frac{q_{ij}.S}{R_l}, & \text{if } q_{ij} = 1 \\ D_{nl} = S/R_l + \frac{q_{nj}.S}{R_{nl}}, & \text{if } q_{nj} = 1 \\ D_{ml} = S/R_l + S/R_{ml}, & \text{otherwise} \end{cases} \tag{20}$$

Therefore, the overall transmission delay is given by

$$D_{Total} = \sum_{i=1}^{M}\sum_{j=1}^{N} \frac{S}{R_l} + \sum_{j=1}^{M}(1 - q_{ij}) \left(\frac{q_{nj}.S}{R_{nl}} + \frac{(1 - q_{nj}).S}{R_{ml}} \right) \tag{21}$$

q_{nj} can be expressed as

$$q_{nj} = 1 - \prod_{i \in E}(1 - q_{ij}) \tag{22}$$

Let c_{max} denote the maximum cache capacity of edge server $i \in E$. Then, cache strategy proposed must satisfy the constraint $\sum_{j=1}^{N} q_{ij}.S \leq c_{max}$.

Energy consumption Model The energy consumption of a system when a user places a request to a content mainly comprises of storage energy for content in the local cache and the transmission energy. Transmission energy includes the energy used to deliver content from the local cache server to a user, energy for wired communications between collaborative cache servers, and energy required to transfer content from the MBS to the edge server if the content is not stored locally or in neighboring caches. Each component contributes to the overall energy footprint of operating an edge server with video caching capabilities, emphasizing the need for efficient energy management practices in both storage and transmission processes.

(1) When a user u_j requests the video content f_j of size S, if the requested content is found in the local edge cache, this will add extra storage energy to the cache server, the storage energy consumption for storing the requested video content on the edge cache can be expressed as follows:

$$E_s = \sum_{j=1}^{M} q_{ij}.\kappa \times T_{f_j} \times S \;\; \forall i \in E \tag{23}$$

Where $q_{ij} = \{0, 1\}$ determines whether the requested video is available in the local cache server, and κ represents the energy consumption for unit bit of stored content, T_{f_j} represents the average storage time of content f_j in the edge server j and S is the size of the requested video content to be transferred.

(2) The energy used in transmitting the requested content f_j from the edge cache to the user u_j can expressed as:

$$E_l = \frac{q_{ij}.P_{ij}.S}{R_l} \quad \forall i \in E, \forall j \in U \tag{24}$$

where P_{ij} represents the transmission power between cache server e_i and user u_j, S denotes the size of the requested content and R_l represents the transmission rate from the edge cache to the user.

(3) When the requested content is not available in the edge caches, it has to be retrieved from the MBS cache. This retrieval process involves transferring the new content from the MBS to the edge cache server. The following equation can be used to calculate the energy consumption involved in the transmission of content items.

$$E_{ml} = (E_{bng} + E_{sw} + N_c E_c + N_e E_e) \times S \tag{25}$$

Where E_c, E_e, E_{bng} and E_{sw} denote the energy consumption per bit for the core router, the edge router, the broadband network gateway, and the Ethernet switch, respectively. N_c represents the number of core routers and N_e is the number of edge routers. S denotes the size of the content that needs to be transmitted (Yan et al. (2019).

Assuming that E_{nl} is the total energy consumed in transferring the video content between the neighbouring edge servers. The total energy consumption E_{Total} can be expressed as:

$$E_{Total} = \sum_{i=1}^{M} (1 - q_{ij}) \left(q_{nj} E_{nl} + (1 - q_{nj}) E_{ml}\right) + \sum_{i=1}^{M} \sum_{j=1}^{N} E_l + E_s \tag{26}$$

where $q_n j$ is as shown in equation (22)

4.3 Optimization of delay and energy consumption problem

In MEC environments, the overall system access delay and energy consumption are the primary metrics to assess the performance of the caching strategy. The approach proposed in this paper optimizes access delay and energy consumption, ensuring uninterrupted video streaming for users by strategically caching content on edge servers. By employing the efficient edge caching strategy to choose the most suitable servers for storing video content, the objective is to reduce user access delay and energy consumption. This problem can be formally represented through mathematical expression as follows.

$$min : \omega D_{Total} + (1 - \omega) E_{Total} \tag{27}$$

$$s.t$$

$$c1 : \sum_{j=1}^{N} q_{ij}.S \leq c_{max} \tag{27a}$$

$$c2 : \sum_{j=1}^{N} q_{ij} = 1 \tag{27b}$$

$$c3 : q_{ij} \in \{0, 1\} \tag{27c}$$

$$c4 : \omega \in [0, 1] \tag{27d}$$

The constraint in equation 27(a) specifies that the video content size must not exceed the edge server's maximum cache capacity. The constraints in equation 27(b) and 27(c) show the binary variable that denotes an access strategy for every user, ensuring that each one of them is connected to only one base station. Additionally, the constraint 27(d) shows a weight parameter used to balance delay and energy consumption.

Problem solution The edge cache algorithm proposed addresses optimization challenges by first analyzing video popularity trends and user preferences. The LSTM model predicts the future video requests of mobile users, forecasting which videos will be needed in the next time slot and caching them locally in advance. During video transmission, delay and energy consumption are optimized using a branch and bound algorithm. The branches are evaluated for their objective function values, with those having larger values pruned until the optimal solution is obtained, and the location for content caching is determined as shown in Algorithm 3.

5. EXPERIMENT AND ANALYSIS

5.1 Experimental environment

Experimental setup: The simulation environment replicates an indoor office setting with multiple RRLHs installed on the ceiling. These RRLHs are equipped with LED photodiodes for VLC and mmWave beam modules for mmWave communication, functioning as high-speed APs to provide seamless connectivity. The OFS manages data traffic between the RRLHs and the SDN controller, which dynamically manages network resources. The SDN controller makes real-time decisions for data routing and content caching.

MATLAB combined with Simulink was chosen for the simulation due to its robust capabilities for modeling and analyzing dynamic systems. The simulation settings included using the Communication System Toolbox for modeling VLC and mmWave protocols and the Simulink Modeling Environment for building and simulating the network architecture. This setup facilitated the integration of different network components and the implementation of the caching strategy. Additionally, MATLAB's Deep Learning Toolbox was used to implement the LSTM model, which predicts user content requests based on historical playback data, enabling proactive caching decisions and improving overall network efficiency.

Experimental dataset: The 360° Video Viewing Dataset (Lo et al. (2017)), published in 2017, was used as the benchmark test data to evaluate the proposed edge caching algorithm. It consists of ten 360° YouTube videos, encoded in H.264 with 4K resolution at 30 fps, chosen for their relevance to head-mounted virtual reality. The dataset size is 4293MB, with 1 to 2-minute clips extracted for this study. These videos are well-suited for reflecting the characteristics of 5G networks, such as high bandwidth and low latency, making the dataset ideal for caching experiments. Its high resolution and variety of content types provide a comprehensive resource for testing caching strategies.

Algorithm 3 Cooperative Cache Algorithm Based on Energy and Delay Using Branch and Bound

Input:
  Feature vector $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iK})$ of user's historical playback information
  M and N: Number of cache servers and video content
  $B_0 = Q_0$: Cache list size
Output: Content cache location collection $loc = \{l_{ij}\}$
for each user $j = 1, 2, \ldots, N$ **do**
  Predict content reading behavior using algorithm 2
end for
Obtain the predicted video content collection $F = \{f_1, f_2, \ldots, f_j, \ldots, f_N\}$
Initialize global upper and lower bounds L_i and u_j for the i-th branch
while $i \leq N$ and $L_i \neq u_j$ **do**
  Select the best branch problem B_{i-1} and split into M sub-branches
  for each sub-branch $j = 1, 2, \ldots, M$ **do**
    Define the new branch question Q_{ij}
    switch (Q_{ij})
      case $Q_{ij} = 1$:
        $l_{ij} = 1$
        Calculate total delay D_{total} and energy E_{total}
        Solve Q_{ij} using Lagrangian relaxation, set as lower bound $LB(Q_{ij})$
        Add solution to loc
        break
      case $Q_{ij} = 0$:
        Scale undetermined variable l_{ij} to $[0, 1]$
        Calculate total delay D_{total} and energy E_{total}
        Solve Q_{ij} using Lagrangian relaxation, set as lower bound $LB(Q_{ij})$
        Add solution to loc
        break
      end switch
  end for
  if all l_{ij} are integers **then**
    Set upper bound $UB(Q_{ij}) = LB(Q_{ij})$
    Prune branch
    $loc \leftarrow loc - Q_{ij}$
  else
    Choose random feasible solutions for upper bound $UB(Q_{ij})$
  end if
  if $LB(Q_{ij}) > u_j$ **then**
    Prune branch
  else
    Add Q_{ij} to active branch list B_i
  end if
  Update L_i with minimum $LB(Q_{ij})$ in B_i
  $i \leftarrow i + 1$
end while
return Content cache location collection $loc = \{l_{ij}\}$

Evaluation metrics: To provide a thorough analysis, selecting the right measures that accurately depict the strategy's efficiency is essential. This study focuses on cache hit rate and data transmission time as the main criteria for evaluating the strategy's effectiveness.

- The hit ratio (Saputra et al. (2019)) is defined by the frequency at which video content is successfully found in the edge cache relative to the total number of requests. A higher hit ratio indicates a more effective caching strategy.
- Access delay (Hoang et al. (2018)) refers to the total time taken from the moment a request is sent until a response is received. A strategy that minimizes this delay is deemed superior, especially since mobile video service users have a lower delay tolerance.
- Energy consumption (Hoang et al. (2018)) becomes a key performance metric with rising data traffic. In various communication environments, caching strategies are adjusted to enhance energy efficiency in mobile edge computing.

Experimental parameters: A simulation experiment platform was established to assess of the edge cooperative caching algorithm performance proposed. The parameters for the simulation are detailed in Table 2. The mmWave path loss parameters are based on (Bai and Heath (2014)) model.

Table 2. System Simulation Parameters

Parameter					**Value**	
VLC Parameters						
Physical area of the photo-diode, A_d					$1cm^2$	
Photo-detector's responsivity, R_d					0.53A/W	
Optical filter gain, $G_c(\vartheta)$					1.0	
Concentrator gain, $G_f(\vartheta)$					0.7	
Half-power angle of the lamp, $\phi_{1/2}$					30°	
Power conversion coefficient, γ					0.9	
System bandwidth of an VLC AP, B^{VLC}					20MHz	
Transmitted optical power, P_t					35dBm	
Angle of radiance, ϕ					40°	
Power spectral density of noise, N_o^{VLC}					10-21A^2/Hz	
FOV semi-angle, ϑ_{FOV}					60°/65°	
mmWave Parameters						
Allocated bandwidth per beam, B^{mmWave}/N					20 MHz	
mmWave AP Transmission Power, P_{nj}					40dBm	
Noise power density, N_o					−174dBm/Hz	
Noise figure, N_f					6 dB	
Transmission distance, d					0 m to 40 m	
Transmitter antenna length					0.1 m	
Receiver antenna length					0.01 m	
Path Loss Parameters		β	α	λ	σ	X_σ^{FI}
28 GHz	NLOS	72	2.92	10.7	8.7	1.0
	LOS	61.4	2	10.7	5.8	5.0
73 GHz	NLOS	86.6	2.45	4.1	8.0	1.0
	LOS	69.8	2	4.1	5.8	8.0
Caching Parameters						
Capacity of edge cache node					[32,256]GB	
Number of UEs					[5,25]	
Relative cache size					[5%,100%]	
Content Size					[50,160]MB	

Benchmark algorithms: To evaluate the performance of the algorithm proposed, the following existing baseline approaches were selected

- Collaborative Edge Caching with Personalized Modeling of Content Popularity (PCEC) (Shen et al. (2022)) combines personalized content popularity modeling with cooperative learning to enhance edge caching in indoor mobile social networks. This approach improves content delivery efficiency and accuracy by tailoring caching to users' specific needs in different indoor environments.
- Q-Learning-based Collaborative Cache Algorithm (Q-LCCA) (Chien et al. (2020)) proposes a collaborative cache mechanism using Q-learning to improve cache performance in Baseband Units and multiple Remote Radio Heads within SDN environment by solving cache allocation problems in a collaboratively. This approach reduces data transmission delay from remote clouds by finding the optimal cache state through reinforcement learning.
- The Least Frequently Used (LFU) algorithm is a caching policy that prioritizes the eviction of cache items based on their access frequency rather than their recency of use.
- The Least Recently Used (LRU) algorithm is a cache replacement policy that evicts items that have not been accessed for the longest time. It operates on the assumption that items not recently used are less likely to be needed soon.

5.2 Results and analysis

In this section, the effect of relative cache size on cache hit rate, access delay, and energy consumption is examined followed by the influence of the number of edge servers on access delay, energy consumption, and cache hit rate. Finally, the impact of relative cache size on access delay, energy consumption, and cache hit rate metrics is analyzed.

(1) Performance evaluation for content reading behaviour prediction

The model was trained on a dataset of user interaction data, including head movements and tile viewing behavior. The model achieved a 93.6% accuracy on the test set, validating its robustness in predicting future video requests based on historical user interactions. Additionally, the precision and recall metrics were calculated to assess the model's ability to correctly identify requested videos. In addition to the LSTM model, other models were trained for comparison, including a GRU model, a Simple RNN, and a 1D CNN. The accuracy of the models are as shown in Figure 3 and the precision, recall and F1-score metrics are as shown in Table 3.

Table 3. User behaviour prediction performance metrics

Model	Precision	Recall	F1-Score
LSTM	93.7%	94.3%	93.6%
GRU	92.2%	92.2%	92.2%
RNN	86.2%	86.2%	86.1%
1D CNN	77.0%	76.4%	76.2%

(2) The Effect of relative cache size on performance

Figure 4 shows how relative cache size affects cache hits, delay and the energy consumption in content delivery by varying the percentage from 5% to 100%.

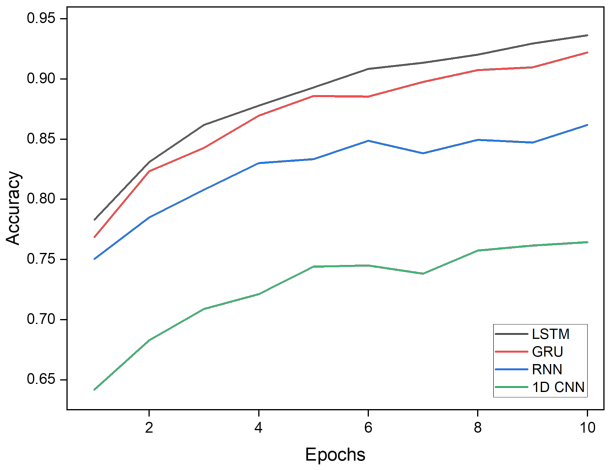


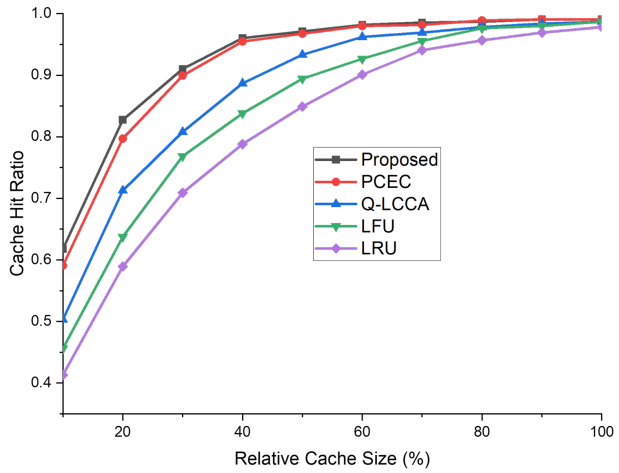
Fig. 3. Comparison of the accuracy of the proposed and other models

As shown in Figure 4(a), increasing the storage capacity of cache servers allows more video content to be stored at the edge nodes. Consequently, the probability of requested video content being cached locally rises, resulting in higher cache hit rates for all five algorithms. With a relative cache size of 40, the proposed algorithm enhances cache hit rate by up to 21.85%, 14.60%, 8.29%, and 5.8% compared to LRU, LFU, Q-LCCA, and PCEC schemes, respectively. This is due to the proposed algorithm's use of SDN and hybrid access technology.

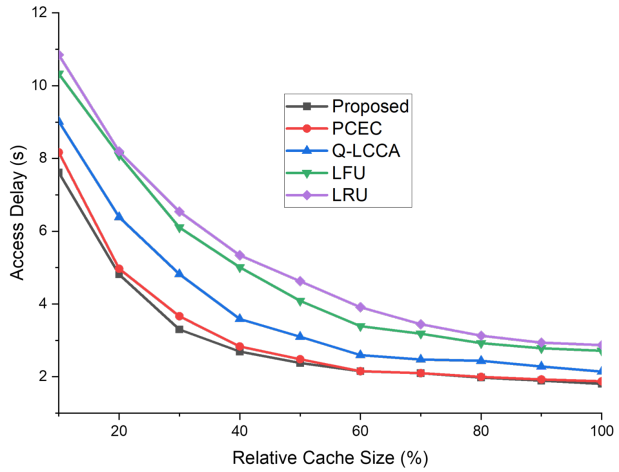
In Figure 4(b), as relative cache size increases, edge nodes in closer proximity to the user can store more video content based on user preferences. This allows end users to retrieve desired video content directly from the local edge cache server and avoids the need for data transfer from the neighbouring edge servers or MBS to the edge server. Consequently, there is a downward trend in the access delay for all algorithms. The proposed algorithm predicts user video requests, ensuring more videos are stored at the edge, thereby reducing access delay. With a relative cache size of 40, the proposed algorithm performs better than PCEC by 4.96%, the Q-LCCA algorithm by 24.98%, the LFU algorithm by 46.29%, and the LRU algorithm by 49.59%.

As shown in Figure 4(c), it is clear that increasing the storage capacity of the edge server leads to a reduction in energy consumption costs for all five algorithms due to the fact taht a larger cache capacity enables the edge node to store more user-preferred videos, minimizing the need for video content transmission from remote servers and thus reducing energy consumption. With a relative cache size of 40, the proposed algorithm minimizes the energy consumption as compared to PCEC algorithm by 5.94%, the Q-LCCA algorithm by 8.77%, the LFU algorithm by 11.77%, and the LRU algorithm by 12.62%.

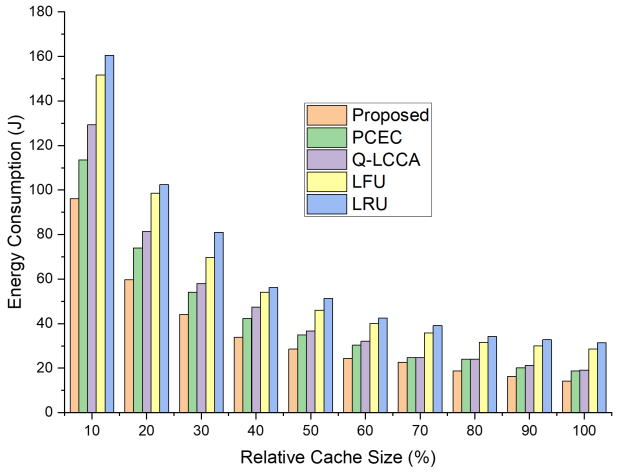
(3) The influence of the number of edge cache nodes on performance The Figure 5 illustrates the effects of varying the number of edge cache nodes on cache hit rate, access delay, and energy consumption. The



(a) Cache hit ratio



(b) Access delay



(c) Energy consumption

Fig. 4. The Effect of relative cache size on performance

study examines cache nodes numbering from 1 to 10, with the number of users remaining consistent.

In Figure 5(a) the cache hit rate is seen to increase as the number of edge cache nodes rises. This improvement is due to the additional storage space provided by the increased number of nodes, allowing more content to be stored proactively. As a result, the probability that a requested video is already cached increases, leading to a higher cache hit rate. The proposed algorithm enhances the cache hit rate by 1.94%, 0.97%, 0.97%, and 0.48% compared to the LRU algorithm, LFU algorithm, Q-LCCA algorithm, and PCEC algorithm, respectively.

In Figure 5(b), it is shown that the access delay decreases as the number of cache servers increases. This reduction occurs because more edge nodes provide additional storage space for proactive content caching, reducing the time required to fetch content from the MBS and consequently lowering system access delay. The proposed algorithm consistently demonstrates the lowest delay, outperforming other algorithms. With 10 edge cache servers, the proposed algorithm minimizes the access delay by 14.44%, 31.14%, 55.27%, and 77.89%, as compared to the PCEC, Q-LCCA, LFU, and LRU algorithms, respectively.

Figure 5(c) illustrates the impact of the number of edge cache servers on energy consumption. As more cache nodes are added, more content is cached, which increases cache energy consumption. However, because cache energy consumption is a small fraction of the total energy consumption, the overall energy consumption decreases due to lower energy requirements for access requests. The proposed algorithm shows the lowest energy consumption, consistently outperforming the other algorithms. With 10 edge cache servers, the proposed algorithm minimizes the energy consumption of PCEC, Q-LCCA, LFU, and LRU algorithms by 20.99%, 20.99%, 48.77%, and 52.46%, respectively.

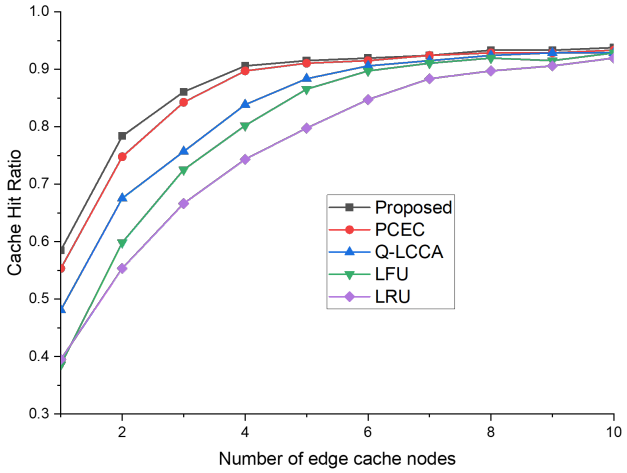
(4) The influence of the number of users on performance

The Figure 6 shows how the access delay, energy, and cache hit rate, consumption are affected by varying the number of users. The study explores a range of 5 to 25 end users with other factors, like the relative cache size, kept constant.

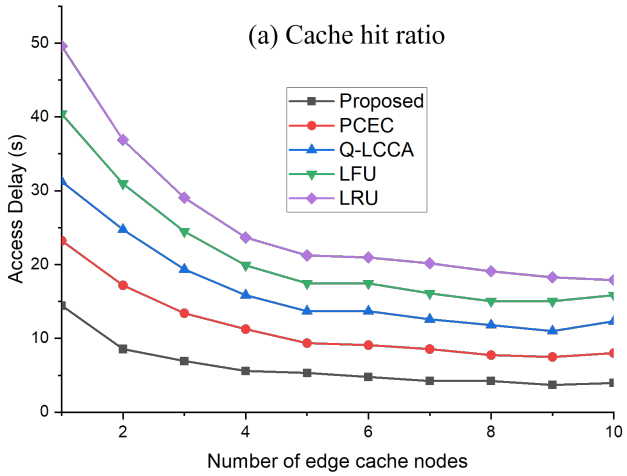
The Figure 6(a), illustrates that as the number of mobile users grows, the cache hit rate decreases. This decline is due to the edge server's limited storage capacity, which can't handle the increasing volume of video content requests, resulting in a lower cache hit rate. Compared to the LRU algorithm, LFU algorithm, Q-LCCA algorithm, and PCEC algorithm, the proposed algorithm improves the cache hit rate by 27.78%, 22.09%, 13.93%, and 2.98%, respectively.

Figure 6(b) shows that the delay in caching algorithms increases with with the increases in number of users. This is because more users lead to more video content requests, and the edge server's limited capacity can't store all the content, resulting in the need to fetch the missed video content from the neighboring edge servers or MBS, thereby increasing access delay. When the number of users is set to 25, the proposed algorithm outperforms the PCEC algorithm by 0.83%, the Q-LCCA algorithm by 7.15%, the LFU algorithm by 18.01% and the LRU algorithm by 19.23%.

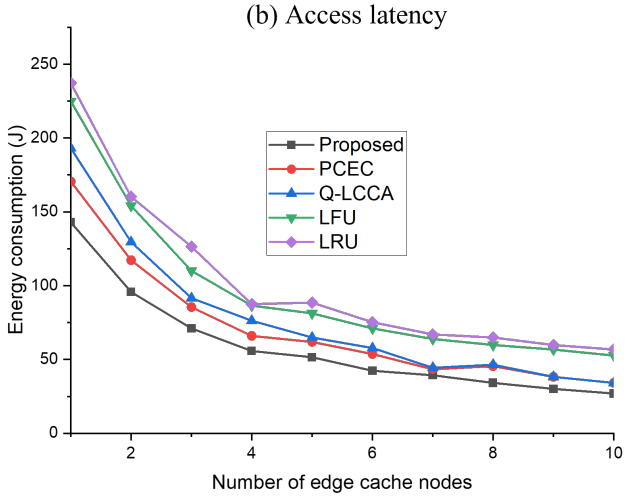
In Figure 6(c), it is observed that as the number of users increases, the energy consumption of all



(a) Cache hit ratio

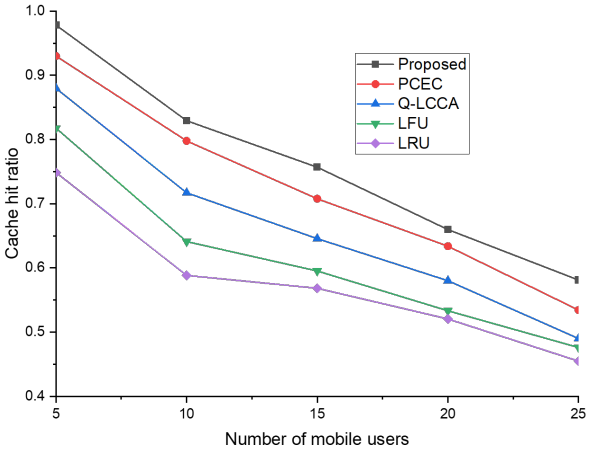


(b) Access latency

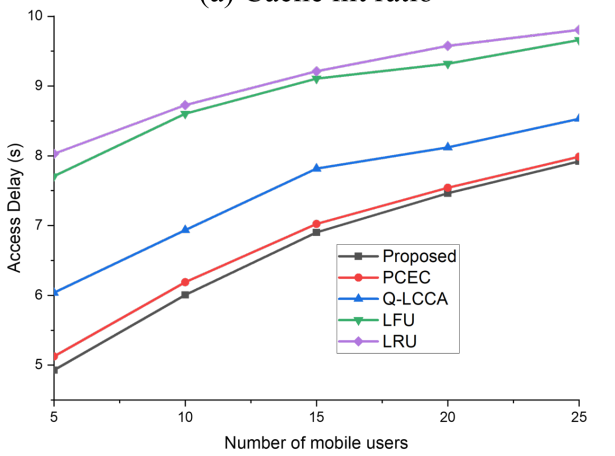


(c) Energy consumption

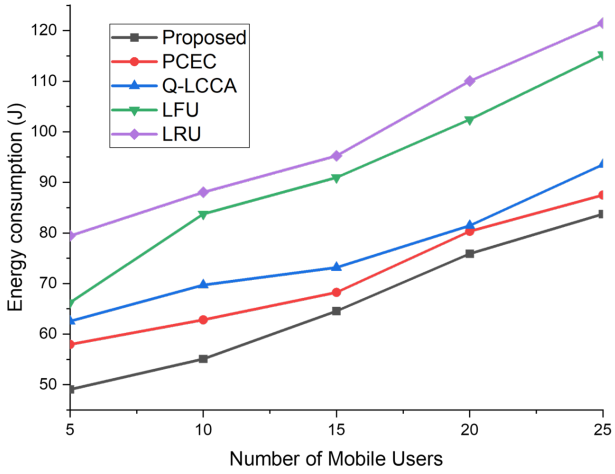
Fig. 5. The influence of the number of edge cache nodes on performance



(a) Cache hit ratio



(b) Access delay



(c) Energy consumption

Fig. 6. The influence of the number of users on performance

five caching strategies also rises. With more users, the variety of requested videos increases, leading to more frequent cache misses. Downloading these videos from remote servers incurs higher transmission energy costs, thereby increasing overall energy consumption. The proposed algorithm significantly lowers energy consumption, reducing it by 44.97%, 37.70%, 12.11%, and 4.61% compared to the LRU algorithm, LFU algorithm, Q-LCCA algorithm, and PCEC algorithm, respectively.

6. CONCLUSION AND FUTURE WORKS

A collaborative video caching strategy to enhance delay and energy efficiency for software-defined hybrid VLC and mmWave networks in 5G edge computing is proposed. By

integrating LSTM network, user preferences were effectively predicted. To address limited storage at the base station, an optimization problem was formulated to reduce delay, energy consumption, and improve cache hit rates. The algorithm demonstrated significant improvements in simulations and is designed for real-world deployment on edge servers connected to various video sources like YouTube. Future work will focus on addressing user mobility, and testing the system in a real-world 5G testbed to evaluate scalability and performance under high-load conditions with diverse content.

REFERENCES

Baccour, E., Erbad, A., Mohamed, A., Guizani, M., and Hamdi, M. (2020). Ce-d2d: Collaborative and popularity-aware proactive chunks caching in edge networks. In *2020 International Wireless Communications and Mobile Computing (IWCMC)*, 1770–1776. IEEE.

Bai, T. and Heath, R.W. (2014). Coverage and rate analysis for millimeter-wave cellular networks. *IEEE Transactions on Wireless Communications*, 14(2), 1100–1114.

Chen, Y., Liu, Y., Zhao, J., and Zhu, Q. (2020). Mobile edge cache strategy based on neural collaborative filtering. *IEEE Access*, 8, 18475–18482.

Chien, W.C., Weng, H.Y., and Lai, C.F. (2020). Q-learning based collaborative cache allocation in mobile edge computing. *Future generation computer systems*, 102, 603–610.

Fath, T. and Haas, H. (2012). Performance comparison of mimo techniques for optical wireless communications in indoor environments. *IEEE Transactions on Communications*, 61(2), 733–742.

Feng, L., Yang, H., Hu, R.Q., and Wang, J. (2018). Mmwave and vlc-based indoor channel models in 5g wireless networks. *IEEE Wireless Communications*, 25(5), 70–77.

Graves, A. and Jaitly, N. (2014). Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning*, 1764–1772. PMLR.

Hoang, D.T., Niyato, D., Nguyen, D.N., Dutkiewicz, E., Wang, P., and Han, Z. (2018). A dynamic edge caching framework for mobile 5g networks. *IEEE Wireless Communications*, 25(5), 95–103.

Kahn, J.M. and Barry, J.R. (1997). Wireless infrared communications. *Proceedings of the IEEE*, 85(2), 265–298.

Khanal, S., Thar, K., and Huh, E.N. (2021). Dcol: Distributed collaborative learning for proactive content caching at edge networks. *IEEE Access*, 9, 73495–73505.

Liu, W., Zhang, H., Ding, H., and Yuan, D. (2022). Delay and energy minimization for adaptive video streaming: A joint edge caching, computing and power allocation approach. *IEEE Transactions on Vehicular Technology*, 71(9), 9602–9612.

Lo, W.C., Fan, C.L., Lee, J., Huang, C.Y., Chen, K.T., and Hsu, C.H. (2017). 360 video viewing dataset in head-mounted virtual reality. In *Proceedings of the 8th ACM on Multimedia Systems Conference*, 211–216.

Molisch, A.F. (2012). *Wireless communications*, volume 34. John Wiley & Sons.

Nie, T., Luo, J., Gao, L., Zheng, F.C., and Yu, L. (2020). Cooperative edge caching in small cell networks with heterogeneous channel qualities. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 1–6. IEEE.

Qin, M., Cheng, N., Jing, Z., Yang, T., Xu, W., Yang, Q., and Rao, R.R. (2020). Service-oriented energy-latency tradeoff for iot task partial offloading in mec-enhanced multi-rat networks. *IEEE Internet of Things Journal*, 8(3), 1896–1907.

Rubio, L., Peñarrocha, V.M.R., Molina-García-Pardo, J.M., Juan-Llácer, L., Pascual-García, J., Reig, J., and Sanchis-Borras, C. (2019). Millimeter wave channel measurements in an intra-wagon environment. *IEEE Transactions on Vehicular Technology*, 68(12), 12427–12431.

Rui, L., Song, D., Chen, S., Yang, Y., Yang, Y., and Gao, Z. (2022). Content collaborative caching strategy in the edge maintenance of communication network: A joint download delay and energy consumption method. *IEEE Transactions on Parallel and Distributed Systems*, 33(12), 4148–4163.

Saputra, Y.M., Hoang, D.T., Nguyen, D.N., and Dutkiewicz, E. (2019). A novel mobile edge network architecture with joint caching-delivering and horizontal cooperation. *IEEE Transactions on Mobile Computing*, 20(1), 19–31.

Shen, S., Yu, C., Zhang, K., and Ci, S. (2022). Collaborative edge caching with personalized modeling of content popularity over indoor mobile social networks. In *ICC 2022-IEEE International Conference on Communications*, 4114–4119. IEEE.

Sutskever, I., Vinyals, O., and Le, Q.V. (2014). Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

Tomaszewski, L., Kukliński, S., and Kołakowski, R. (2020). A new approach to 5g and mec integration. In *Artificial Intelligence Applications and Innovations. AIAI 2020 IFIP WG 12.5 International Workshops: MHDW 2020 and 5G-PINE 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings 16*, 15–24. Springer.

Wang, N., Shen, G., Bose, S.K., and Shao, W. (2018). Zone-based cooperative content caching and delivery for radio access network with mobile edge computing. *IEEE Access*, 7, 4031–4044.

Xia, X., Chen, F., He, Q., Grundy, J., Abdelrazek, M., and Jin, H. (2020). Online collaborative data caching in edge computing. *IEEE Transactions on Parallel and Distributed Systems*, 32(2), 281–294.

Yan, M., Chan, C.A., Li, W., Lei, L., Gygax, A.F., and Chih-Lin, I. (2019). Assessing the energy consumption of proactive mobile edge caching in wireless networks. *IEEE Access*, 7, 104394–104404.

Zhang, T. and Mao, S. (2019). Cooperative caching for scalable video transmissions over heterogeneous networks. *IEEE Networking Letters*, 1(2), 63–67.

Zhao, X., Yuan, P., Tang, S., et al. (2018). Collaborative edge caching in context-aware device-to-device networks. *IEEE Transactions on Vehicular Technology*, 67(10), 9583–9596.