IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 6 November 2023, accepted 5 January 2024, date of publication 15 January 2024, date of current version 22 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3353812

RESEARCH ARTICLE

End-to-End Multi-User 360-Degree Video Delivery Using Users' Fixation Points

TSUBASA OKAMOTO[1], TAKUMASA ISHIOKA[1,2], (Member, IEEE), TATSUYA FUKUI[3], RYOUHEI TSUGAMI[3], TOSHIHITO FUJIWARA[3], SATOSHI NARIKAWA[3], TAKUYA FUJIHASHI[1], (Member, IEEE), SHUNSUKE SARUWATARI[1], (Member, IEEE), AND TAKASHI WATANABE[1], (Member, IEEE)

[1]Faculty of Engineering Department of Information and Computer Science, Kyoto Tachibana University, Graduate School of Information Science and Technology, Osaka University, Osaka 565-0871, Japan
[2]Faculty of Engineering Department of Information and Computer Science, Kyoto Tachibana University, Kyoto 607-8175, Japan
[3]Access Network Service Systems Laboratories, Nippon Telegraph and Telephone Corporation, Tokyo 180-8585, Japan

Corresponding author: Tsubasa Okamoto (okamoto.tsubasa@ist.osaka-u.ac.jp)

This work was supported in part by JSPS KAKENHI under Grant 22H03582, and in part by NTT Research.

ABSTRACT Viewport-based 360-degree video delivery is a typical method to reduce video traffic for virtual reality (VR) applications. However, viewport-based solutions cause key issues in multi-user VR applications, including high video traffic due to redundant video transmission across multiple headset users and quality degradation due to viewport transitions. These problems occur in both camera-to-server and server-to-user video transmissions. In this study, we propose a 360-degree video delivery scheme for multi-user VR applications. To overcome the above issues, the proposed approach includes appropriate quality and transmission control for camera-to-server and server-to-user video transmissions. The camera extracts the estimated potential region from the dual fisheye video. The server controls recompression at the server to follow the viewport transition and hybrid unicast and multicast tile delivery to avoid redundant transmissions. Evaluations using 360-degree video and corresponding fixation points from multiple users show that the proposed scheme prevents redundant transmissions across multiple headset users and provides better viewport quality for each user under the same video traffic. For example, the proposed scheme reduces video traffic by up to 36.4% compared to the existing viewport-based 360-degree video delivery scheme for ten headset users.

INDEX TERMS Virtual reality (VR), dual fisheye, hybrid multicast and unicast.

I. INTRODUCTION

With the widespread adoption of virtual reality (VR) headsets and 360-degree cameras, VR services are expected to be used in a variety of fields such as medicine, education, and entertainment. In such services, multiple users can simultaneously view the same 360-degree video in real-time from different perspectives through their headsets.

The simplest way to deliver end-to-end 360-degree video is to encode the full resolution 360-degree video from the 360-degree camera and send it to users over networks. Each user then views a portion of the 360-degree video, called a viewport, on their headset. However, this approach causes video quality degradation under the limited bandwidth both the camera-to-server and the server-to-user networks because the resolution of the 360-degree video is even larger. Existing studies have mainly proposed viewport-based 360-degree video delivery to reduce the required traffic between the server and users. Specifically, each user sends the position of its viewport to the server, and the server encodes and sends back the part of the 360-degree video corresponding to the given viewport position. Typically, the resolution of a viewport is about one-eighth of the full resolution of the 360-degree video.

However, viewport-based 360-degree video streaming schemes suffer from three challenging issues. The first issue is still the large amount of traffic between the camera and server networks. The existing studies assumed that the 360-degree camera sends the full resolution of 360-degree video frames to the server because they assumed a wired

The associate editor coordinating the review of this manuscript and approving it for publication was Andrea Bottino.

connection between the 360-degree camera and the server. However, the networks between the 360-degree camera and the server are not broadband, such as wireless networks and best-effort networks. The bandwidth limitation between the 360-degree camera and the server causes the quality degradation of the 360-degree video.

The second issue is large video traffic due to redundant transmissions between users. When multiple users request the same 360-degree video through their headset, each user's requested viewport from each user overlaps with the viewport of other users. The overlapping areas cause duplicate video transmissions between users, and the traffic increases with the number of users.

The third issue is quality degradation due to time variation in each user's viewport. Each user's viewport moves according to eye movement during video playback. If the end-to-end delay between the 360-degree camera and the headset users is long, a portion of the 360-degree video is encoded and transmitted to the users based on the previous viewport position. Quality degradation and stuttering occur when the gap between the previous and current viewport positions is large.

This study proposes a novel end-to-end 360-degree video delivery scheme to address the above three issues. The main contributions of the proposed scheme are threefold. The first contribution is to design a viewport-based 360-degree video delivery for dual fisheye video to reduce the video traffic between the 360-degree camera and the server. The second contribution is to classify the regions within the 360-degree video frames and the bit allocation algorithm for each region based on the requests from multiple users. The third contribution is to use multicast and unicast to send bit-allocated tiles to multiple users to eliminate redundant transmissions between users. We conducted evaluations using 360-degree video and the corresponding fixation point datasets. The evaluation results show that the proposed scheme reduces the required traffic between the 360-degree camera and the server, and such traffic reduction improves the end-to-end viewport quality under the same required traffic. In addition, the proposed scheme reduces the performance degradation under a long end-to-end delay by estimating the viewing region from the past fixation points.

II. RELATED WORKS

Our study is related to viewport-based 360-degree video delivery and multi-user 360-degree video delivery.

A. VIEWPORT-BASED 360-DEGREE VIDEO DELIVERY

When a full-resolution 360-degree video is transmitted over band-limited networks, the received video quality of the 360-degree video is degraded due to the large amount of traffic. User's viewport-based 360-degree video delivery has been proposed to deliver high-quality 360-degree video over band-limited networks. The key idea of viewport-based delivery is that the server adaptively encodes the 360-degree video according to each user's viewport.

To implement viewport-based 360-degree video delivery, some studies [1], [2], [3], [4], [5], [6] have designed tile-based 360-degree video delivery. Specifically, the server maps the captured dual fisheye video to the equirectangular format and divides the equirectangular video into multiple tiles. Each tile is independently encoded and delivered to the user according to the user's perspective. In [1] and [2], the server encodes and sends only the tiles corresponding to the user's viewport. In [3] and [4], the server obtains the display probability of each tile and adaptively assigns bits to each tile according to the probability. In [7], standardized scalable video coding, i.e., scalable high-efficiency video coding [8], was used for viewport-based 360-degree video delivery. Specifically, a server encodes an entire 360-degree video into a base layer for baseline quality and the tiles corresponding to the viewport into enhancement layers for quality enhancement. Tile size optimization is discussed in [5]. In [6], the authors defined a quality model that considers the speed of the fixation point motion, the pixel luminance fluctuation, and the objects around the fixation point to determine the compression ratio of each tile.

Viewport-based 360-degree video delivery can reduce traffic, while mispredicting the viewport position can degrade video quality. Some studies aim to accurately estimate the viewport position to reduce the quality degradation [1], [9], [10], [11], [12], [13], [14]. The existing studies can be classified into regression-based [1], [9], [10], [11] and learning-based methods [12], [13], [14]. The regression-based methods estimate future viewport positions using linear regression [9], [11] and weighted linear regression [1], [10] based on the user's past head orientation. The learning-based methods estimate the viewport position based on the saliency maps and the user's head orientations using deep learning architectures [12], [15]. Specifically, convolutional neural networks and long short-term memory (LSTM) were combined [15] and recurrent neural networks and LSTM were combined [12] for viewport prediction.

B. MULTI-USER 360-DEGREE VIDEO DELIVERY

Some studies extend the viewport-based 360-degree video delivery schemes to the multi-user environments [7], [16], [17]. The existing studies can be classified into on-demand and live services.

The recent work [18] proposed a multi-user 360-degree video delivery for live services. In the live services, a server unicasts the tiles corresponding to the viewport to each headset user. In this case, redundant video transmission may occur between the headset users when the viewport of some users overlaps with that of other users. The proposed scheme in [18] selectively uses unicast and multicast for each tile to prevent redundant video transmissions.

For the on-demand service, existing studies have introduced edge servers for multi-user 360-degree video delivery. Edge servers in [19], [20], [21], and [22] cache the 360-degree video for low-delay 360-degree video delivery. Specifically, they estimate the popularity of each tile and cache the popular

tiles to the edge servers near the users. An edge server in [23] transcodes the received video from the remote server for bit allocation and multicasts some tiles to multiple users to prevent buffer exhaustion in each user. Another study in [24] uses an image processing technique for smooth playback, i.e. frame interpolation.

C. CONTRIBUTIONS OF OUR STUDY

The proposed scheme is the multi-user 360-degree video delivery scheme for live services. Specifically, the server divides the full-resolution 360-degree video into multiple tiles and determines the bit allocation and transmission methods for each tile based on the fixation points of each user. Unlike existing 360-degree video delivery schemes and our previous work [18], the proposed scheme reduces the end-to-end traffic from the camera to the users.

For this purpose, in the proposed scheme, the server detects potential regions that may be displayed by headset users on the dual fisheye format based on user feedback, and sends the coordinates of the potential regions to the 360-degree camera. The 360-degree camera discards the pixels outside the potential regions and encodes the video frames in the dual fisheye format. By removing the redundant pixels on the 360-degree camera, the quality on the user's headset is improved.

III. PROPOSED SCHEME

A. OVERVIEW

Fig. 1 shows the end-to-end architecture of the proposed scheme. A 360-degree camera captures 360-degree video in the dual fisheye format. A server is connected to the 360-degree camera. Let R (bps) be the available bandwidth between the 360-degree camera and the server. In addition, one or more headset users are simultaneously connected to the server. The available bandwidth between each headset user and the server is $\hat{R}$, and the total available bandwidth between the server and all the headset users is $\bar{R} \geq \hat{R}$.

The 360-degree camera encodes and sends the video frames in the dual fisheye format, and the server maps the video frames to an equirectangular format [25] for encoding and transmission. Each headset user frequently sends his/her perspective information to the server. We consider the fixation point $\boldsymbol{p}$ in the equirectangular format to be sent back by each headset user. The server determines the user's viewport based on the received fixation point and the field of view (FoV) of each user's headset. In this paper, the FoV is assumed to be 90 degrees. The server also calculates the coordinates of the viewport in the dual fisheye format. The coordinates are then sent to the 360-degree camera. The 360-degree camera discards the corresponding pixel values based on the coordinates prior to encoding and transmission.

B. CAMERA-SIDE OPERATIONS

Each headset user sends the coordinates of the fixation point in a 3D Cartesian coordinate (x, y, z). Based on the fixation point, the server can determine which regions in the dual fisheye format are potentially displayed to headset users by referencing the coordinate sets V_{dl} and V_{dr}. These regions are called "potential regions". The 360-degree camera in the proposed scheme can reduce traffic by discarding the pixel values outside the potential regions by receiving the coordinates of the potential regions from the server.

The server must convert the received fixation point to the coordinates in the dual fisheye format (x_d, y_d) to detect the potential regions in the dual fisheye format. The 3D Cartesian coordinates are first transformed to the equirectangular coordinates $(x_{\mathrm{e}}, y_{\mathrm{e}})$ as follows

$$x_{\mathrm{e}} = \frac{2W}{\pi} \arccos \frac{x}{\sqrt{x^2 + y^2}}, \quad y_{\mathrm{e}} = \frac{2W}{\pi} \arccos z \tag{1}$$

where W is the pixel width of the equirectangular 360-degree video. Based on the fixation point in the equirectangular coordinates and the FoV of the headset, the server can define the simple potential regions in the equirectangular 360-degree video.

However, a round-trip delay between the headset and the 360-degree camera may cause quality degradation when the headset users want to display outside the potential regions. In this case, the server expands the potential regions based on the past movement of the fixation points. Let d be the round-trip delay between sending the user's feedback and viewing the corresponding video on the headset. When the instantaneous time of the received fixation point is t, the server needs to estimate how much potential region should be expanded at time $t + d$. For this purpose, the proposed method expands the region based on statistical information obtained from the user's past fixation points as follows:

$$E_{t+d}\ [\text{degrees}] = v + \alpha M_{t,w} + \beta \tag{2}$$

Here, v [degrees] is the FoV of the headset. $M_{t,w}$ is the maximum degree of fixation point movement from the time t to the past w frames. Here we assume that the fixation movement from t to $t+d$ is less than the maximum movement in the past w frames. α and β are hyperparameters. We do not consider the orientation of the fixation point motion, so we consider α of 2.

The server can determine the potential region of each headset user based on the fixation point $(x_{\mathrm{e}}, y_{\mathrm{e}})$ and the degree of the region E_{t+d} for each user. In addition, it considers the union of the potential region over all headset users as the final potential regions. The equirectangular coordinates in the final potential regions are listed in a set $V_{x_{\mathrm{e}}, y_{\mathrm{e}}}$.

All the equirectangular coordinates in the set $V_{x_{\mathrm{e}}, y_{\mathrm{e}}}$ are transformed into the coordinates in the left fisheye image $(x_{\mathrm{dl}}, y_{\mathrm{dl}})$ and in the right fisheye image $(x_{\mathrm{dr}}, y_{\mathrm{dr}})$ as follows:

$$x_{\mathrm{dl}} = \frac{2r}{\pi} \frac{\cos\theta}{\sqrt{1 - \sin^2\theta \sin^2\phi}} \arccos\left(-\sin\theta \cos\phi\right) + x_{\mathrm{cl}} \tag{3}$$

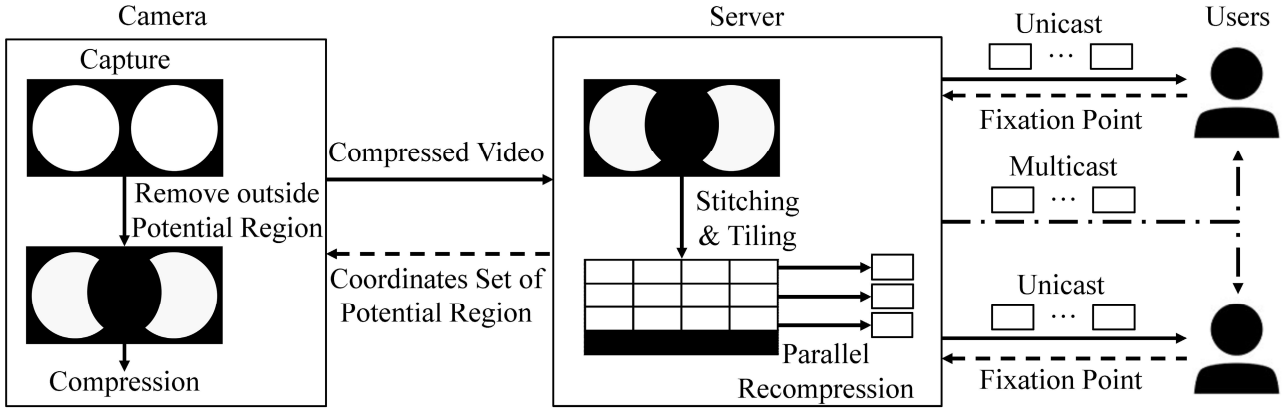


FIGURE 1. End-to-end architecture of proposed 360-degree video delivery scheme.

$$y_{\mathrm{dl}} = \frac{2r}{\pi} \frac{\sin\theta\cos\phi}{\sqrt{1-\sin^2\theta\sin^2\phi}} \arccos\left(-\sin\theta\cos\phi\right) + y_{\mathrm{cl}} \tag{4}$$

$$x_{\mathrm{dr}} = -\frac{2r}{\pi} \frac{\cos\theta}{\sqrt{1-\sin^2\theta\sin^2\phi}} \arccos\left(\sin\theta\cos\phi\right) + x_{\mathrm{cr}} \tag{5}$$

$$y_{\mathrm{dr}} = \frac{2r}{\pi} \frac{\sin\theta\cos\phi}{\sqrt{1-\sin^2\theta\sin^2\phi}} \arccos\left(\sin\theta\cos\phi\right) + y_{\mathrm{cr}} \tag{6}$$

where x_{cl} and y_{cl} are the center coordinates of the left fisheye and x_{cr} and y_{cr} are the center coordinates of the right fisheye images, respectively. The coordinates corresponding to the left and right fisheye images are assigned to the coordinate sets V_{dl} and V_{dr} and sent to the 360-degree camera for traffic reduction.

C. SERVER-SIDE OPERATIONS

1) TILE DIVISION

The server receives the 360-degree video frames in the dual fisheye format. The video frames are mapped to the equirectangular format and the equirectangular video frames are divided into multiple tiles. We consider the video to be divided into M tiles in the vertical direction and N tiles in the horizontal direction. Here, the resolution of the video is $H \times W$ pixels, and the resolution of each tile is $H/M \times W/N$ pixels. The server then encodes each tile in parallel according to the available bandwidth R and transmits the encoded tiles to the servers.

The server detects each user's viewport within the frame based on the feedback information and the FoV. The server considers the tiles that overlap with the viewport as the tiles to be viewed. These are called "viewing tiles" and are then classified into two categories, including 1) the viewing tiles desired by all headset users and 2) the other viewing tiles. The proposed scheme replicates the other viewing tiles for the following procedures. The server also considers the tiles that do not overlap with any user's viewport as non-viewing tiles. Note that some tiles have no pixel values because the 360-degree camera discards the pixels outside the potential regions. Such tiles are called "discarded tiles". The server decompresses and recompresses the viewing and non-viewing tiles based on the classification. The algorithm used to determine the quality of each tile is described in Sec. III-C2.

2) BIT-ASSIGNMENT ALGORITHM

If sufficient bandwidth is available between the server and each headset user, as many bits as possible should be allocated to all tiles to improve the quality of the VR experience. However, the available bandwidth is usually limited to deliver the tiles with sufficient quality. In this case, the server should use recompression to assign enough bits to each tile to satisfy the bandwidth limitation. A brute-force approach provides an easy way to determine the bit allocation for each tile. However, if the degree of bit depth for each tile is L steps and the total number of tiles is NM, the time complexity for each tile is $\mathcal{O}(L^{NM})$.

In this study, we propose a bit allocation algorithm with less computation time, whose time complexity is $\mathcal{O}(NM + \log L)$. Algorithm 1 shows pseudocode of the proposed algorithm. The variables and functions used in the algorithm are shown in Table 1. The proposed method has two steps. The first step determines the bit allocation for viewing and non-viewing tiles. Specifically, the server determines the allocation quality l for viewing tiles according to the available bandwidth $\bar{R}$ and $\hat{R}$ as follows:

$$l = \min(l_1, l_2), \tag{7}$$

$$l_1 = \max \bar{l} \quad \text{s.t. } 1 \le \bar{l} \le L, \quad S_{\bar{l}} \le \frac{\bar{R} - (NM - n_U - n_0)S_1}{n_{\ge 1}}, \tag{8}$$

$$l_2 = \max \hat{l} \quad \text{s.t. } 1 \le \hat{l} \le L, \quad S_{\hat{l}} \le \frac{\hat{R} - (NM - n - n_0)S_1}{n}. \tag{9}$$

Here, the variables used in the algorithm are listed in Table 1. Eq. (8) finds the maximum bit depth for all the

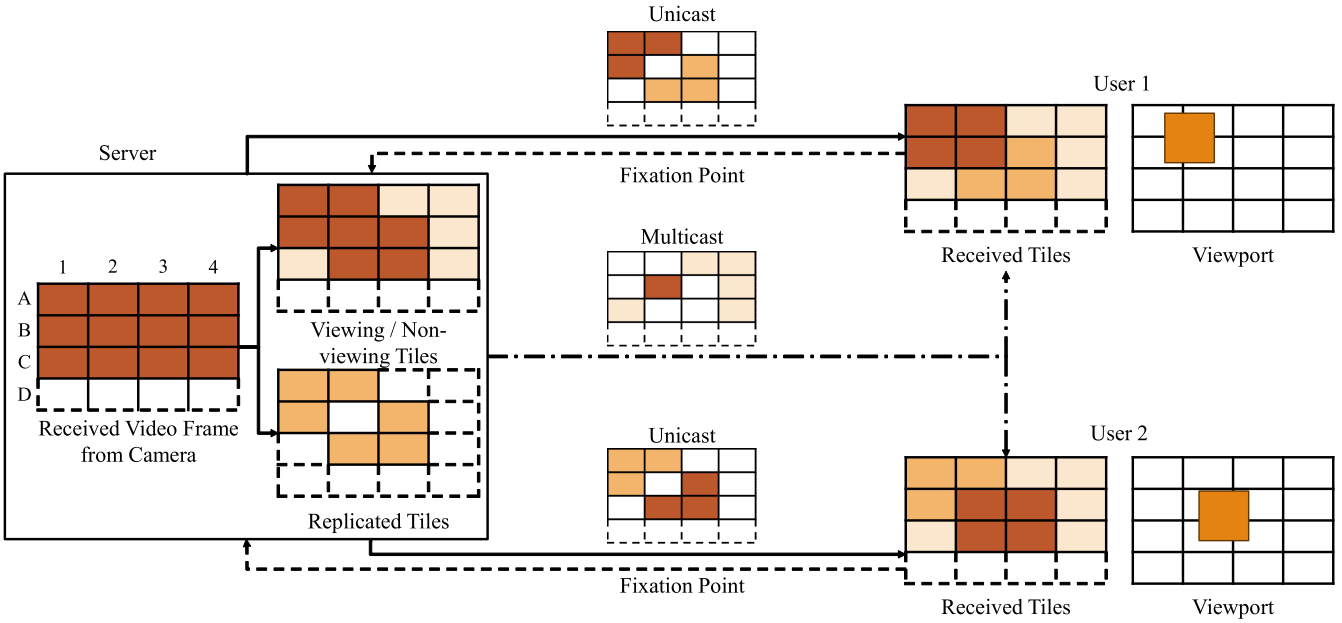


FIGURE 2. Proposed bit-assignment algorithm and hybrid unicast and multicast tile delivery.

TABLE 1. Variables in Bit-Assignment Algorithm.

Variables	Description
U	Number of concurrent users
T_H	Allocation quality for viewing tiles
T_L	Allocation quality for replicated tiles
S_l	Traffic per tile at quality l
n_0	Number of discarded tiles
n	Number of viewing tiles per user
$n_{>1}$	Number of viewing tiles in any user's viewport
n_U	Number of tiles in all users' viewports

users' viewing tiles when the server assigns the lowest bit depth for the non-viewing tiles and replicates viewing tiles under the available bandwidth $\bar{R}$. Eq. (9) also finds the maximum bit depth for each user's viewing tiles when the server assigns the lowest bit depth for the non-viewing tiles under the available bandwidth $\hat{R}$. If l_1 and l_2 are different, one of the bit assignments does not meet the bandwidth requirements, and thus, the server uses a smaller bit depth for the assignment. In the second step, we determine the assigned bits for the replicated viewing tiles. The proposed scheme utilizes the replicated tiles to prevent quality degradation when the estimated viewport at the server differs from the user's actual viewport. Specifically, we find the maximum bit depth for the replicated viewing tiles that can be delivered over the remaining bandwidth. The replicated viewing tiles are then encoded according to the obtained bit depth.

3) HYBRID UNICAST AND MULTICAST TILE DELIVERY

Each viewing and replicated tile is then classified according to the number of users. Specifically, if more than two headset users request the same tile, the server multicasts the tile to the requested users, while if a single user requests the tile, the server unicasts the tile to the user. Note that the server multicasts the non-viewing tiles to all headset users. Fig. 2 shows an example of two users viewing the same 360-degree video. The 360-degree video is divided into tiles of $M = 4$ and $N = 4$. Note that tiles D1, D2, D3, and D4 are not sent to the server because they are not in the potential region. Users 1 and 2 send their fixation points to the server. The server estimates user 1's viewing tiles as A1, A2, B1, and B2, and user 2's viewing tiles as B2, B3, C2, and C3. In this case, user 1's replicated tiles are B3, C2, and C3 and user 2's replicated tiles are A1, A2, and B1. The server decompresses and recompresses the non-viewing tiles A3, A4, B4, C1, and C4 to the lowest quality and multicasts them to the users. The viewing tiles A1, A2, and B1 needed by user 1 and viewing tiles B3, C2, and C3 needed by user 2 are unicast to each user. In addition, the replicated tiles for users 1 and 2 are decompressed and recompressed based on the bit depth determined in the second step of the proposed bit-assignment algorithm and unicast to each user.

IV. EVALUATION

A. EVALUATION SETTINGS

1) TEST SEQUENCE

We used a single 360-degree video and corresponding fixation point datasets [15]. The resolution of the 360-degree video was 1920 × 960 pixels and the frame rate was 30 fps. The fixation point dataset consisted of the eye movement of multiple users who watched a 360-degree video for 30 seconds. We used the 6.6 seconds of eye movement information for comparison.

Algorithm 1 Bit-Assignment Algorithm at Server

function Allocation

Step 1: Bit assignment for viewing and non-viewing tiles

$l_1, l_2 = L$

if $\bar{R} < n_{\geq 1} \cdot S_L + (NM - n_U)S_1$ **then**

Obtain l_1 in Eq. (8)

if $\hat{R} < n \cdot S_L + (NM - n)S_1$ **then**

Obtain l_2 in Eq. (9)

$T_H = \min(l_1, l_2)$

Step 2: Bit assignment for replicated tiles

left $= 0$, right $= T_H + 1$

while (right − left) > 1 **do**

mid $= \lfloor$(right − left)$/2 \rfloor$

if $(NM - n_U - n_0)S_{\text{mid}} \leq (\bar{R} - n_{\geq 1}S_{T_H})$ **then**

left $=$ mid

else

right $=$ mid

$D_1 =$ left

left $= 0$, right $= T_H + 1$

while (right − left) > 1 **do**

mid $= \lfloor$(right − left)$/2 \rfloor$

if $(NM - n_U - n_0)S_{\text{mid}} \leq (\hat{R} - nS_{T_H})$ **then**

left $=$ mid

else

right $=$ mid

$D_2 =$ left

return $T_H, T_L = \min(D_1, D_2)$

2) CAMERA AND SERVER SETTINGS

The network delay between the 360-degree camera and the server and between the server and each headset user was set to 100 ms. Here, the proposed scheme considers the parameters for the potential regions of $\alpha = 2$, $\beta = 10$, and $w = 6$. The effect of the parameters on the video traffic is discussed in Sec. IV-E.

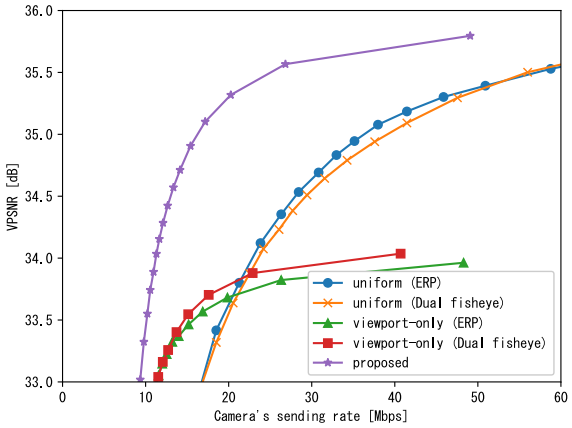
3) QUALITY METRIC

We defined the quality metric of each user's viewport as the viewport peak signal-to-noise ratio (VPSNR) as follows:

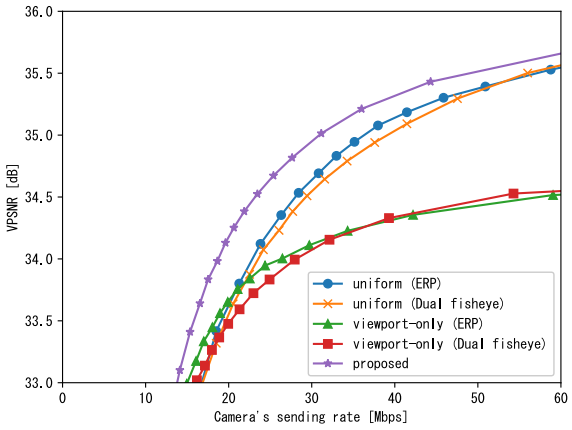
$$\text{VPSNR} = 10 \log_{10} \left(\frac{255^2}{\text{VMSE}} \right), \tag{10}$$

$$\text{VMSE} = \frac{1}{|\mathbf{V}|} \sum_{(h,w) \in \mathbf{V}} [I(h, w) - K(h, w)]^2 \tag{11}$$

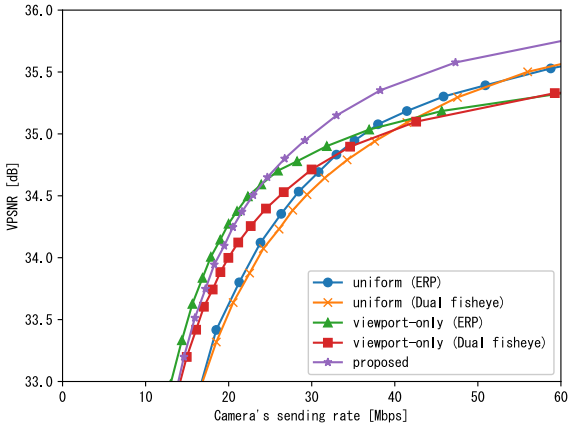
where $\mathbf{V}$ is a set of tiles corresponding to the viewport, $I(h, w)$ is (h, w)-th pixel value of an original 360-degree video frame, and $K(h, w)$ is (h, w)-th pixel value of the reconstructed 360-degree video frame. Here, PSNR above 33 dB achieved the mean opinion score of four and five [26], i.e., sufficient quality of experience, and thus, we discuss the performance of traffic reduction under VPSNR above 33 dB.



(a) Two users



(b) Five users



(c) Ten users

FIGURE 3. Video quality as a function of the sending rate from the camera.

B. CAMERA-SIDE TRAFFIC REDUCTION

We first examine the performance of the camera-side traffic reduction. We consider four baselines: uniform and viewport-only schemes with/without equirectangular projection (ERP). The uniform scheme sends the full resolution of the 360-degree video frames. The viewport-only scheme only sends the video by discarding outside the viewport based on the past fixation point. Figs. 3 (a) to (c) show the viewport

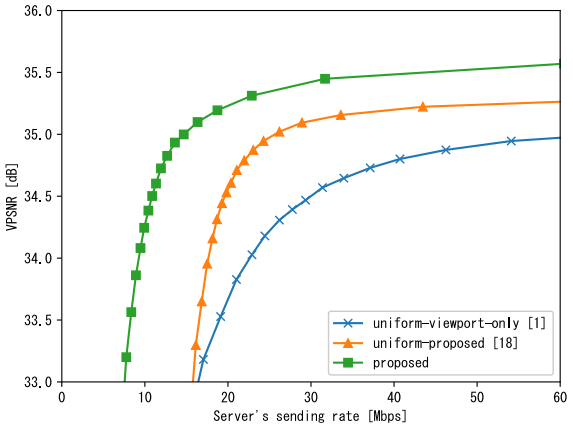
quality as a function of the camera's sending rate for two, five, and ten users, respectively. We can see the following observations:

- The proposed scheme achieves the lowest traffic for the same viewport quality for two and five users by sending only the tiles corresponding to the potential region.
- The performance difference between the proposed and baseline schemes decreases as the number of headset users increases, because all proposed and baseline schemes send almost the entire region of the 360-degree video to the server with a large number of headset users.
- The viewport-only scheme suffers from low viewport quality with the same traffic because the headset user may receive the tiles outside the viewport.
- The performance gap between the baseline schemes with/without the equirectangular projection is small. This means that the 360-degree camera may not be needed to perform the mapping, i.e., an additional operation.

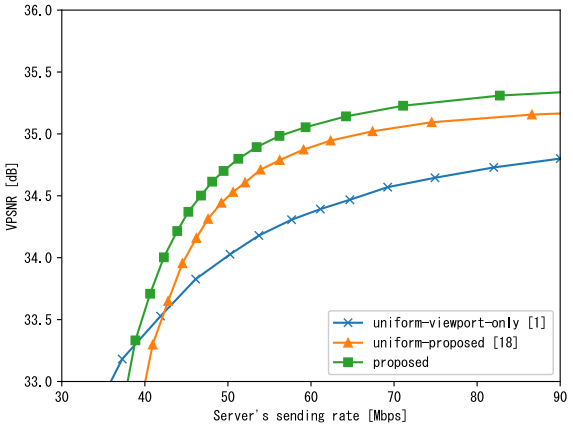
C. END-TO-END PERFORMANCE

The previous section showed that the proposed camera-side operation achieves traffic reduction while maintaining the same video quality. In this section, we evaluate the impact of camera-side traffic reduction on end-to-end performance. We considered three schemes for comparison: uniform-viewport-only, uniform-proposed, and proposed schemes. The uniform-viewport-only scheme [1] transmits the entire dual fisheye video from the camera to the server and unicasts the tiles corresponding to each user's viewport in the equirectangular format. The uniform-proposed scheme is our previous work [18]. It transmits the entire dual fisheye video from the camera to the server, and the server transmits each tile in the equirectangular format as mentioned in Sec. III-C. Here, each scheme divides the equirectangular 360-degree video into 6×6 tiles, and thus the total number of tiles is 36. The effect of the tile division is also discussed in Sec. IV-E.

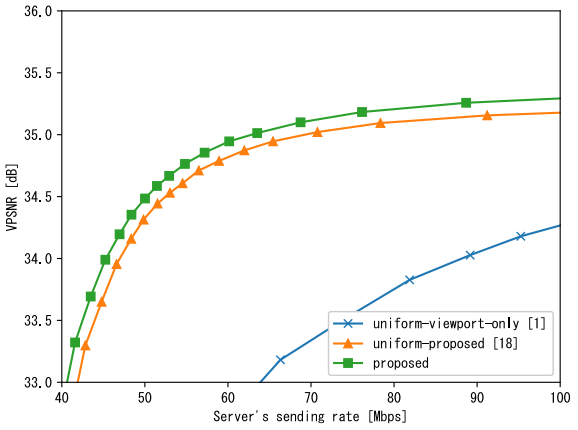
Figs. 4 (a) through (c) show the video quality as a function of the sending rate of the server under two, five, and ten headset users, respectively. The proposed scheme achieves the lowest traffic under the same video quality, especially under two and ten users. This is because the quality improvement in the camera-side operation brings the traffic reduction in the end-to-end performance. On the one hand, the performance gap between the proposed scheme and the uniform-proposed scheme becomes smaller as the number of headset users increases. This is because the 360-degree camera in the proposed scheme transmits almost the entire region of the 360-degree video to the server under ten headset users. On the other hand, the performance gap between the proposed scheme and the uniform-viewport-only scheme becomes larger for a large number of users.



(a) Two users



(b) Five users



(c) Ten users

FIGURE 4. Video quality as a function of the sending rate from the server.

To discuss the effect of the number of headset users, Fig. 5 shows the sending rate of the server as a function of the number of headset users. Here we consider the average VPSNR over the headset users of about 33.0 dB. The evaluation results show the following results:

- The proposed scheme achieves the lowest sending rate regardless of the number of users. The integration of unicast and multicast in the proposed scheme

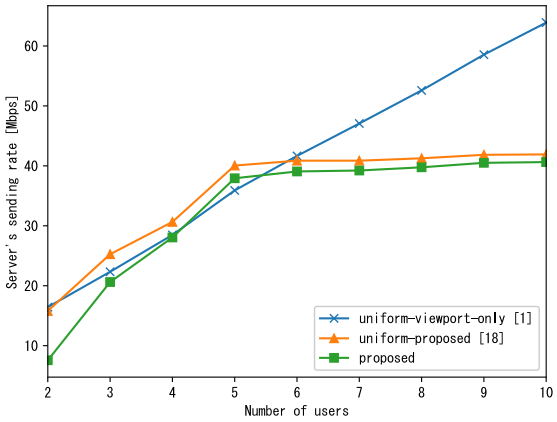


FIGURE 5. Server's sending rate as a function of the number of users.

saturates the traffic at a large number of headset users.

- The viewport-only scheme linearly increases the traffic as the number of users increases because it causes redundant video transmission across the headset users.

Finally, we discuss the visual quality of the proposed and baseline schemes under the same traffic. Figs. 6 (a) to (d) show the viewports of the original 360-degree video frame, uniform-viewport-only, uniform-proposed, and proposed schemes when ten headset users are connected to the server, respectively. Here, the sending rate of the server is 40 Mbps.

The uniform-viewport-only scheme contains a black area at the bottom of the viewport due to the time-variation of the user's viewport. The camera-side traffic reduction in the proposed scheme enhances the viewport quality, and thus, the proposed scheme can reconstruct a clean viewport as shown in Fig. 6 (d).

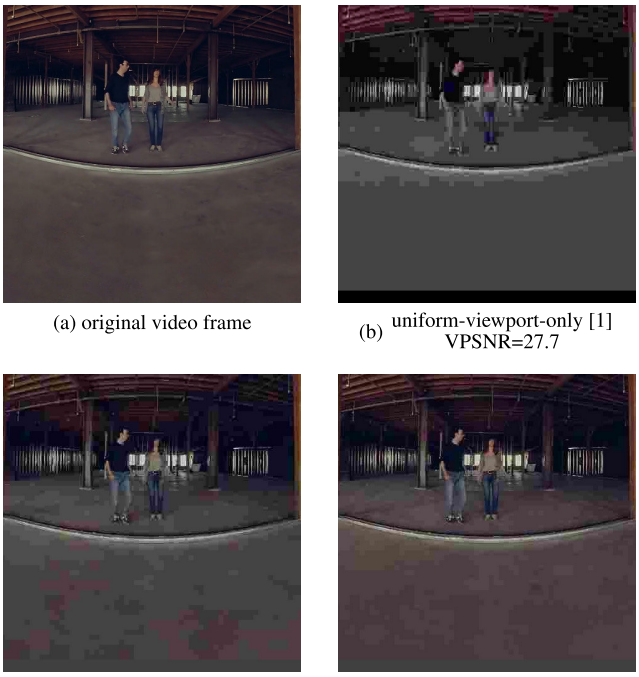


FIGURE 6. Visual quality of the original and reconstructed viewports in baseline and proposed schemes.

D. EFFECT ON CAMERA-SERVER AND SERVER-USER DELAY

The above evaluations showed the traffic reduction under the fixed and identical camera-server and server-user delay. Here, a long/short delay and a biased delay can affect the traffic reduction in the proposed scheme. Fig. 7 (a) shows the server's sending rate as a function of the end-to-end delay with ten users. Here, the camera-server and server-user delays are identical, and the average VPSNR over the headset users is approximately 33.0 dB. The uniform-viewport-only and proposed schemes increase the required traffic as the end-to-end delay increases. In addition, the proposed scheme achieves a smaller ratio of the increase in the server's sending rate to the end-to-end delay than the uniform-viewport-only scheme.

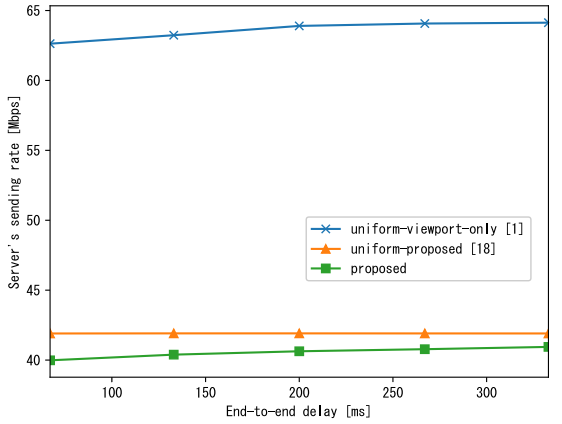
Fig. 7 (b) shows the server's sending rate as a function of the camera-server delay with ten users. Here, the end-to-end delay is fixed at 200 ms. It means when we set the camera-server delay to 0 ms, we consider the server-to-user delay to be 200 ms. The proposed and uniform-proposed schemes achieve almost the same traffic irrespective of the biased delay environments. The uniform-viewport-only slightly increases the traffic as the camera-server delay decreases. A large server-user delay may cause a wrong viewport to be displayed on the user's headset, and thus a large amount of traffic is required to improve the viewport quality.

E. EFFECT OF TILE DIVISION AND CAMERA-SIDE PARAMETERS

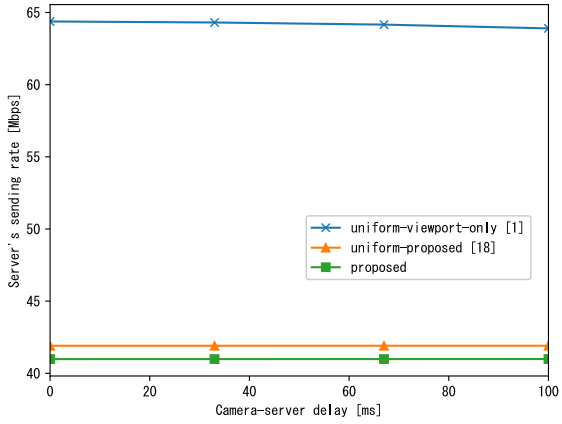
In the above discussion, the baseline and proposed schemes divide the full resolution of 360-degree video frames into 6×6 tiles. Here, the number of divided tiles impacts the traffic reduction performance due to the resolution of each tile and overhead.

We discuss the performance of the proposed scheme under the different numbers of divided tiles. Fig. 8 shows the viewport quality as a function of the sending rate of the server. Here, the server equally divides the 360-degree video into 4×4, 6×6, 8×8, and 10×10, respectively. We can see that the 4×4 or 6×6 tile division is the best performance irrespective of the sending rates. A large number of divided tiles increases the traffic because the overhead of each tile is more significant.

In addition, the proposed camera-side operation determines the potential regions based on the past fixation points with the window size of w and parameter β. From the preliminary discussion, the window size w has a small impact on the traffic reduction. In particular, the proposed scheme



(a) Identical delay



(b) Biased delay

FIGURE 7. Server's sending rate as a function of the identical and biased delay under ten headset users.

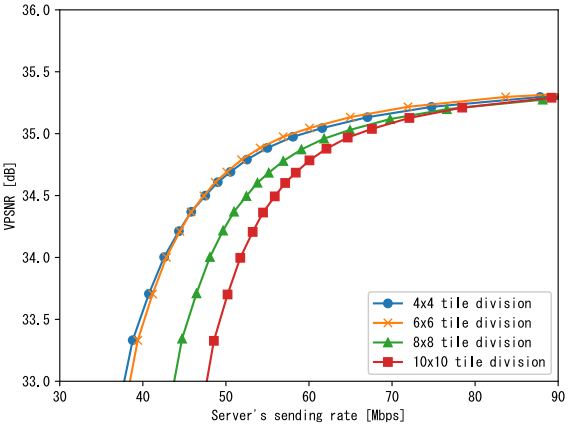


FIGURE 8. Effect of tile division on traffic and viewport quality.

achieves almost the same camera and server sending rate even if we change the window size from 1 to 10. Fig. 9 discusses the viewport quality as a function of the sending rate of the camera. Here we consider β [degrees] of $2, 5, 10, 20, 40$. We can see that $\beta = 20$ gives the best performance. If β is too small, users are more likely to view outside the potential

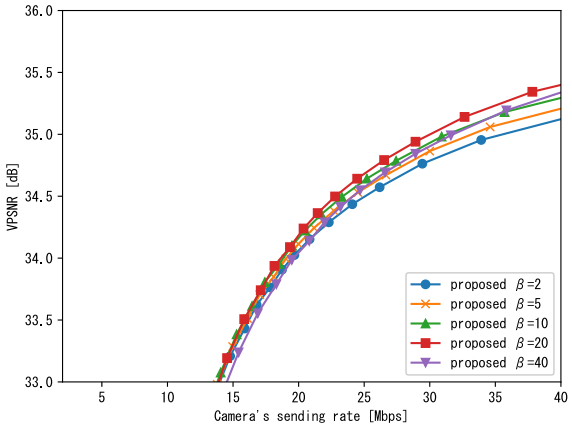


FIGURE 9. Effect of parameter β on traffic and viewport quality.

region. If β is too large, the required traffic increases due to an even larger potential region.

V. CONCLUSION

In this study, we have proposed a novel end-to-end 360-degree video delivery scheme for multiple headset users. The proposed scheme integrates the camera-side and server-side traffic reduction to provide a high-quality viewport for each headset user under the same amount of video traffic. Evaluations using 360-degree video and the corresponding fixation point datasets demonstrated that the proposed camera-side traffic reduction improves each user's viewport quality under the same traffic requirement.

ACKNOWLEDGMENT

This work was partly supported by JSPS KAKENHI Grant Number 22H03582 and NTT Research.

REFERENCES

[1] F. Qian, L. Ji, B. Han, and V. Gopalakrishnan, "Optimizing 360 video delivery over cellular networks," in *Proc. 5th Workshop All Things Cellular, Oper., Appl. Challenges*, Oct. 2016, pp. 1–6.

[2] M. Hosseini, "View-aware tile-based adaptations in 360 virtual reality video streaming," in *Proc. IEEE Virtual Reality (VR)*, Mar. 2017, pp. 423–424.

[3] Y. Im, T. Qiu, L. B. Milstein, and P. C. Cosman, "Tile-based wireless streaming of 360-degree video with rate adaptation using viewport estimation," *IEEE Signal Process. Lett.*, vol. 29, pp. 2707–2711, 2022.

[4] J. Chakareski, R. Aksu, X. Corbillon, G. Simon, and V. Swaminathan, "Viewport-driven rate-distortion optimized 360° video streaming," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–7.

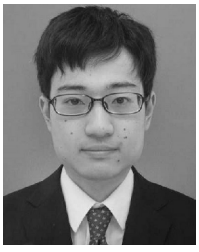
[5] X. Chen, T. Tan, and G. Cao, "Macrotile: Toward QoE-aware and energy-efficient 360-degree video streaming," *IEEE Trans. Mobile Comput.*, vol. 23, no. 2, pp. 1–16, Dec. 2022.

[6] Y. Guan, C. Zheng, X. Zhang, Z. Guo, and J. Jiang, "Pano: Optimizing 360° video streaming with a better understanding of quality perception," in *Proc. ACM Special Interest Group Data Commun.*, Aug. 2019, pp. 394–407.

[7] G. He, J. Hu, H. Jiang, and Y. Li, "Scalable video coding based on user's view for real-time virtual reality applications," *IEEE Commun. Lett.*, vol. 22, no. 1, pp. 25–28, Jan. 2018.

[8] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of SHVC: Scalable extensions of the high efficiency video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 20–34, Jan. 2016.

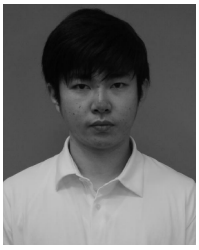
[9] L. Xie, Z. Xu, Y. Ban, X. Zhang, and Z. Guo, "360ProbDASH: Improving QoE of 360 video streaming using tile-based HTTP adaptive streaming," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 315–323.
[10] A. T. Nasrabadi, A. Mahzari, J. D. Beshay, and R. Prakash, "Adaptive 360-degree video streaming using scalable video coding," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 1689–1697.
[11] F. Duanmu, E. Kurdoglu, S. A. Hosseini, Y. Liu, and Y. Wang, "Prioritized buffer control in two-tier 360 video streaming," in *Proc. Workshop Virtual Reality Augmented Reality Netw.*, Aug. 2017, pp. 13–18.
[12] A. Nguyen, Z. Yan, and K. Nahrstedt, "Your attention is unique: Detecting 360-degree video saliency in head-mounted display for head movement prediction," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1190–1198.
[13] B. Han, Y. Liu, and F. Qian, "ViVo: Visibility-aware mobile volumetric video streaming," in *Proc. 26th Annu. Int. Conf. Mobile Comput. Netw.*, Apr. 2020, pp. 1–13.
[14] X. Jiang, S. A. Naas, Y.-H. Chiang, S. Sigg, and Y. Ji, "SVP: Sinusoidal viewport prediction for 360-degree video streaming," *IEEE Access*, vol. 8, pp. 164471–164481, 2020.
[15] Y. Xu, Y. Dong, J. Wu, Z. Sun, Z. Shi, J. Yu, and S. Gao, "Gaze prediction in dynamic 360° immersive videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5333–5342.
[16] C. Guo, Y. Cui, and Z. Liu, "Optimal multicast of tiled 360 VR video," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 145–148, Feb. 2019.
[17] A. Mahzari, A. T. Nasrabadi, A. Samiei, and R. Prakash, "FoV-aware edge caching for adaptive 360° video streaming," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 173–181.
[18] T. Okamoto, T. Ishioka, R. Shiina, T. Fukui, H. Ono, T. Fujiwara, T. Fujihashi, S. Saruwatari, and T. Watanabe, "Edge-assisted multi-user 360-degree video delivery," in *Proc. IEEE 20th Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2023, pp. 194–199.
[19] H. Xiao, C. Xu, Z. Feng, R. Ding, S. Yang, L. Zhong, J. Liang, and G.-M. Muntean, "A transcoding-enabled 360° VR video caching and delivery framework for edge-enhanced next-generation wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 5, pp. 1615–1631, May 2022.
[20] P. Blasco and D. Gündüz, "Learning-based optimization of cache content in a small cell base station," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 1897–1903.
[21] J. Chakareski, "Viewport-adaptive scalable multi-user virtual reality mobile-edge streaming," *IEEE Trans. Image Process.*, vol. 29, pp. 6330–6342, 2020.
[22] Y. Jin, J. Liu, F. Wang, and S. Cui, "Ebublio: Edge assisted multi-user 360-degree video streaming," *IEEE Internet Things J.*, vol. 10, no. 17, pp. 15408–15419, Apr. 2023.
[23] L. Zhong, X. Chen, C. Xu, Y. Ma, M. Wang, Y. Zhao, and G.-M. Muntean, "A multi-user cost-efficient crowd-assisted VR content delivery solution in 5G-and-beyond heterogeneous networks," *IEEE Trans. Mobile Comput.*, vol. 22, no. 8, pp. 4405–4421, Mar. 2022.
[24] S. Yang, P. Yang, J. Chen, Q. Ye, N. Zhang, and X. Shen, "Delay-optimized multi-user VR streaming via end-edge collaborative neural frame interpolation," *IEEE Trans. Netw. Sci. Eng.*, vol. 11, no. 1, pp. 284–298, Jan. 2024.
[25] G.-I. Kweon and Y.-H. Choi, "Image-processing based panoramic camera employing single fisheye lens," *J. Opt. Soc. Korea*, vol. 14, no. 3, pp. 245–259, Sep. 2010.
[26] T. Zinner, O. Abboud, O. Hohlfeld, T. Hossfeld, and P. Tran-Gia, "Towards QoE management for scalable video streaming," in *Proc. 21st ITC Spec. Seminar Multimedia Appl.-Traffic, Perform. QoE*, Miyazaki, Japan, Mar. 2010.



TSUBASA OKAMOTO received the B.S. degree from Osaka University, Osaka, Japan, in 2022, where he is currently the M.S. degree with the Graduate School of Information Science and Technology.

TAKUMASA ISHIOKA (Member, IEEE) received the B.E. and M.E. degrees from Osaka University, Japan, in 2019 and 2021. Since April 2023, he has been an Assistant Professor with the Faculty of Engineering, Kyoto Tachibana University, Japan. His research interest includes wireless networks. He is a member of IPSJ.

TATSUYA FUKUI received the B.E. and M.E. degrees from the Faculty of Science and Engineering, Waseda University, Japan, in 2008 and 2010, respectively. He is currently with Access Network Service Systems Laboratories, Nippon Telegraph and Telephone Corporation. His current research interest includes research and development of carrier networks, such as wide-area Ethernet systems.



RYOUHEI TSUGAMI received the B.E. and M.E. degrees in engineering from Nagasaki University, Japan, in 2018 and 2020, respectively. In 2020, he joined Access Network Service Systems Laboratories, Nippon Telegraph and Telephone Corporation. His current research interest includes research and development of data collection systems for cyber-physical systems (CPS) using remote direct memory access (RDMA).

TOSHIHITO FUJIWARA received the B.E., M.E., and Ph.D. degrees in engineering from the University of Tsukuba, Ibaraki, Japan, in 2002, 2004, and 2011, respectively. In 2004, he joined Access Network Service Systems Laboratories, Nippon Telegraph and Telephone Corporation, Japan, where he has been involved in the research and development of optical video transmission systems, passive optical network systems, content delivery network systems, and ultralow latency video systems.

SATOSHI NARIKAWA received the B.E., M.E., and Ph.D. degrees from the Department of Electrical and Electronics Engineering, Tokyo Institute of Technology, Japan, in 2001, 2003, and 2012, respectively. He is currently with Access Network Service Systems Laboratories, Nippon Telegraph and Telephone Corporation. His current research interests include research and development of optical access systems.

TAKUYA FUJIHASHI (Member, IEEE) received the B.E. and M.S. degrees from Shizuoka University, Japan, in 2012 and 2013, respectively, and the Ph.D. degree from the Graduate School of Information Science and Technology, Osaka University, Japan, in 2016. From 2014 to 2015, he was an Intern with the Electronics and Communications Group, Mitsubishi Electric Research Labs. (MERL). From 2014 to 2016, he was a Research Fellow (DC1) with the Japan Society for the Promotion of Science. In 2016, he was also a Research Fellow (PD) with the Japan Society for the Promotion of Science. Since April 2019, he has been an Assistant Professor with the Graduate School of Information Science and Technology, Osaka University. His research interests include video compression and communications, with a focus on immersive video coding and streaming.

SHUNSUKE SARUWATARI (Member, IEEE) received the Dr.Sci. degree from The University of Tokyo in 2007. From 2007 to 2008, he was a Visiting Researcher with the Illinois Genetic Algorithm Laboratory, University of Illinois at Urbana–Champaign. From 2008 to 2012, he was a Research Associate with the RCAST, The University of Tokyo. From 2012 to 2016, he was an Assistant Professor with Shizuoka University. Since 2016, he has been an Associate Professor with Osaka University. His research interests include wireless networks, sensor networks, and system software.

TAKASHI WATANABE (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Osaka University, Japan, in 1982, 1984, and 1987, respectively. In 1987, he joined the Faculty of Engineering, Tokushima University, as an Assistant Professor, and moved to the Faculty of Engineering, Shizuoka University, in 1990. From 1995 to 1996, he was a Visiting Researcher with the University of California, Irvine. He is currently a Professor with the Graduate School of Information Science and Technology, Osaka University, Japan. His research interests include mobile networking, ad hoc networks, sensor networks, ubiquitous networks, intelligent transport systems, especially MAC and routing. He is a member of the IEEE Communications Society, the IEEE Computer Society, IPSJ, and IEICE. He has served on many program committees for networking conferences, such as IEEE, ACM, IPSJ, and The Institute of Electronics, Information and Communication Engineers (IEICE), Japan.