Viewpoint Leakage in Proactive VR Streaming: Modeling and Tradeoff

Xing Wei, Chenyang Yang, and Chengjian Sun

Abstract

Proactive tile-based virtual reality (VR) video streaming employs the trace of viewpoint of a user to predict future requested tiles, then renders and delivers the predicted tiles before playback. Recently, it has been found that the identity and preference of the user can be inferred from the viewpoint data uploaded in proactive streaming procedure. When considering viewpoint leakage, several fundamental questions arise. When is the viewpoint leaked? Can privacy-preserving approaches, e.g., federated or individual training, or using local predicting and predictors with no need for training avoid viewpoint leakage? In this paper, we investigate viewpoint leakage during proactive streaming. We find that if the prediction error or the quality of experience (QoE) metric is uploaded for adaptive streaming, the real viewpoint can be inferred even with the above privacy-preserving approaches. Then, we define viewpoint leakage probability to characterize the inference accuracy of the real viewpoint, and respectively derive the probability when uploading prediction error and the QoE. We find there is a conditional tradeoff between viewpoint leakage probability and prediction performance, QoE, or the resources. Simulation with the state-of-the-art predictor over a real dataset shows that the tradeoff between viewpoint privacy and prediction performance, resources, and QoE exist for most cases.

Index Terms

Privacy-aware proactive VR, viewpoint leakage, VR privacy, 360 video privacy

I. INTRODUCTION

As the main type of wireless virtual reality (VR) services, 360° video streaming consumes large amount of computing and communication resources to avoid playback stalls and black holes that degrade the quality of experience (QoE). However, humans can only see a small
spherical cap of the full panoramic sphere (e.g., 18% of the sphere [1]) at arbitrary time, which is called the field of view (FoV), and the viewpoint (i.e., center of the FoV) can be predicted in advance [2]. To improve QoE with limited resources, proactive VR video streaming is proposed [3], which contains viewpoint predictor training and online streaming stages.

Predictor training with collected viewpoint trace can be operated at the multi-access edge computing (MEC) server or the head mounted display (HMD), and can be omitted for some simple predictors with no need for training as those used in [2], [3]. Online streaming stage contains viewpoint prediction, proactive rendering and transmitting. The QoE depends on the combined effect of viewpoint prediction and configured resources for rendering and transmitting. To keep the QoE stable during the online streaming, the real-time prediction error or the QoE should be uploaded for adaptive streaming [4]–[7].

A. Motivation and Major Contributions

While proactive VR video streaming has been intensively investigated, most existing works neglect that privacy may be leaked during the proactive streaming procedure, where the request of a VR video and the trace of viewpoint should be uploaded. Although the request of a VR video can be anonymous to avoid privacy leakage, as the key information to improve QoE and save resources, the trace of viewpoint is inevitably to be uploaded. Then, will the privacy be leaked from the viewpoint trace?

Recent works reveal the risk. With less than five minutes’ viewpoint trace, the identity of 95% users among all the 511 users can be correctly identified in [8]. This indicates that the viewpoint trace may contain biometric information. Furthermore, with the trace of viewpoint, the content of FoV that the user chooses to see is also leaked. This can be used to reveal the intent and preference of the user [9]. Therefore, viewpoint leakage incurs privacy leakage.

When considering viewpoint leakage in proactive VR video streaming procedure, several fundamental questions arise:
• Can existing privacy-preserve approaches, e.g., federated learning in predictor training and local predicting in online streaming, avoid viewpoint leakage?

• If the leakage is unavoidable, when does the most serious viewpoint leakage happens? When the viewpoint can be protected to the largest extent?

• Is there any relation between the viewpoint leakage, prediction performance, configured resources, and QoE?

In this paper, we strive to answer these questions. Our main contributions are as follows.

• We find that although the viewpoint leakage can be avoided with privacy-preserve approaches in predictor training, the real viewpoint can be inferred in online streaming.

• To characterize the leakage, we define viewpoint leakage probability and derive the probability when uploading the prediction error and the QoE, from which we find the conditions that the probability achieves the maximum and minimum.

• We find that prediction performance, QoE, or resources can be traded for the reduction of the viewpoint leakage probability. Simulations with the state-of-the-art predictor shows that the tradeoff exists in most cases.

B. Related Works

As far as the authors known, there are no prior works to investigate viewpoint leakage in VR video streaming.

Existing works consider protecting real gaze position (another useful data for viewpoint prediction) by adding noise in spatial domain [9]–[12] and reducing the samples of uploaded gaze position in temporal domain [9], [13].

However, protecting real gaze position cannot be transferred into viewpoint protecting, but the insight gained from our work can be transferred into gaze leakage in VR video streaming, owing to the following reasons.
• Gaze position may not be always useful for viewpoint prediction. In fact, the state-of-the-art accuracy on a real dataset [1] can be achieved only with the trace of viewpoint [2].

• With privacy-preserving approaches, real gaze position can be stored at the HMD and the protecting becomes unnecessary. The same is true for viewpoint protecting.

• More important issues are that with such privacy-preserving approaches, whether the viewpoint can be still leaked out, and when does the most serious viewpoint leakage happen.

• Existing works consider the gaze position on two dimensional (2D) plane, while the actual gaze position or viewpoint is on the three-dimensional sphere, the impact of which on the privacy leakage has not been investigated.

The rest of this paper is organized as follows. Section II describes the system model. Section III defines the viewpoint leakage probability. Section IV and V derive and analyze the probability when uploading the prediction error and the QoE metric, respectively. Trace-drive simulation results are provided in Section VI to show the tradeoff between viewpoint privacy and prediction performance, resources, or the QoE exists for most cases. Section VII concludes the paper.

II. System Model

Consider a proactive VR video streaming system with a MEC server, which accesses a VR video library by local caching or high-speed backhaul, thus the delay from the Internet to the server can be ignored. The server also equips with powerful computing units for training, predicting, and rendering.

Each of $K$ users wears an HMD, which can measure and upload the trace of viewpoint,\(^1\) calculate and upload the real-time QoE and prediction error, and pre-buffer segments. The HMD can also equip with a light-weighted computing unit for training a predictor and predicting the viewpoint. During the playback, each user can turn around freely to view FoVs.

\(^1\)Since tile requests can be transformed into the viewpoints [14], we do not consider the case that uploading the tile requests.
Each 360° video consists of $L$ segments, each segment consists of $N_f$ panoramic video frames in temporal domain, and each frame consists of $M$ tiles in spatial domain. For ease of analysis, we assume that the areas of all the tiles within a FoV and a streamed field of view (SFoV) in a panoramic frame can be respectively approximated as the area of the FoV and SFoV [15]. This can be achieved by sufficient fine grained tiling or adaptive tiling [4].

![Diagram](image.png)

Fig. 1: Streaming the first four segments of a VR video. $t_b$ is the start time of the observation window, $t_e$ is the start time of playback of the $l_0$th segment, $l_0 = 3$.

A. **Proactive VR Video Streaming Procedure**

Proactive VR video streaming requires to predict viewpoints. If a predictor needs to be trained in advance, the whole proactive procedure contains two stages: (1) offline predictor training, and (2) online streaming. Otherwise, stage (1) can be omitted.

Predictor training can be operated at the MEC server or the HMD. When training at the MEC server (i.e., centralized learning), the real viewpoint traces of each users should be uploaded. When training at each HMD (i.e., federated learning), the real viewpoint traces can be stored...
locally and only the model parameters of the predictor are uploaded to the MEC server. When training local predictors individually at each HMD, both the real viewpoint traces and the model parameters of predictor can be stored at the HMD.

The procedure in the online streaming stage is shown in Fig. 1. When a user requests a VR video, the radius of the FoV, denoted as $r_{\text{fov}}$, is uploaded. The MEC server first renders and transmits the initial $(l_0 - 1)$ segments in a passive manner [16]. After an initial delay, the first segment begins to play at the time instant $t_b$, i.e., the start time of an observation window. Then, proactive streaming begins. In the sequel, we take the $l_0$th segment as an example for elaboration.

After the viewpoint trace in the observation window with duration $T_{\text{obw}}$ is collected, the viewpoint sequence in a prediction window with duration $T_{\text{pdw}} = T_{\text{seg}}$ can be predicted at the MEC server or a HMD. According to the distance between the predicted viewpoint and center of each tile, the MEC server can determine the tiles for the $N_f$ frames of the segment to be streamed [14], [17]. Then, the server renders these tiles with duration $t_{\text{cpt}}$ to generate the $N_f$ images of SfOVs, which are then transmitted with duration $t_{\text{com}}$. To avoid stalling, $N_f$ SfOVs in the $l_0$th segment should be delivered before the playback start time of the $l_0$th segment, i.e., the time instant $t_e$. The duration starting from $t_b$ to $t_e$ is the online proactive streaming time for a segment, denoted as $T_{\text{ps}}$. We can observe from the figure that $T_{\text{ps}} = (l_0 - 1)T_{\text{seg}}$, where in the example $l_0 = 3$ and hence $T_{\text{ps}} = 2T_{\text{seg}}$. The durations for observation, computing, and transmission should satisfy $T_{\text{obw}} + T_{\text{cc}} = T_{\text{ps}}$, where $T_{\text{cc}} \triangleq t_{\text{com}} + t_{\text{cpt}}$ is the total duration for communication and computing. A predictor can be more accurate with a smaller value of $T_{\text{cc}}$. This is because the viewpoints to be predicted are closer to and hence are more correlated with the viewpoint sequence in the observation window [17]. Given a predictor and required viewpoint prediction accuracy, the value of $T_{\text{cc}}$ can be pre-determined [14], [17], [18].

During the playback of the segment, the real-time QoE metric or the prediction error can be calculated at the HMD and uploaded to the MEC server for adaptive streaming, i.e., when the prediction error is large or the QoE is small, the MEC server can stream more tiles [4]–[7].
B. Field of View on Sphere

As illustrated in Fig. 2a, when watching a panoramic video, the user wearing an HMD is at the center of the unit sphere, i.e., \(O\). The FoV of the HMD can be considered as a spherical cap of the sphere [1], [14]. For any given HMD, the size of FoV is determined. Denote viewpoint as \(O_v\). The spherical distance from \(O_v\) to the base of the cap is \(r_{fov}\) (in radian), which we refer to as the “cap radius”. The half apex angle of the cone corresponding the cap is \(\alpha\). When measures in radian, \(\alpha = \frac{r_{fov}}{r} = r_{fov}\). Then, the area of the FoV is \(A_{fov} = 2\pi (1 - \cos(\alpha)) = 2\pi (1 - \cos(r_{fov}))\).

Fig. 2: FoV, SFoV, and QoE on the unit sphere, \(O_v\) and \(O_p\) are respectively the real and predicted viewpoint, \(r_{fov}\) and \(r_{sv}\) are respectively the radius of FoV and SFoV, \(e\) is the prediction error.

C. Computing and Transmission Model

1) Computing Model: For each user, the number of bits that can be rendered per second, referred to as the computing rate [17], is \(c_{cpt,k} \triangleq \frac{F_{cpt}}{K\mu_r}\) (in bit/s), where \(F_{cpt}\) is the configured resource at the server for rendering (in floating-point operations per second, FLOPs/s), \(\mu_r\) is the required floating-point operations (FLOPs) for rendering one bit of FoV (in FLOPs/bit) [17].

2) Transmission Model: The base station co-located with the server equipped with \(N_t\) antennas serves \(K\) single-antenna users using zero-forcing beamforming. The instantaneous data rate at
the $i$th time slot in $t_{\text{com}}$ for the $k$th user is $c_{\text{com},k}^i = B \log_2 \left( 1 + \frac{p_k d_k^{-\beta} |(h_k^i)^H w_k^i|^2}{\sigma^2}\right)$, where $B$ is the bandwidth, $h_k^i$ and $w_k^i$ are respectively the channel vector and beamforming vector, $d_k$ is the distance from the BS to the user, $p_k$ is the transmit power, $\beta$ is the path-loss exponent, $\sigma^2$ is the noise power, and $(\cdot)^H$ denotes conjugate transpose. Denote the time-average data rate over $t_{\text{com}}$ as $\tau_{\text{com},k}$. We use ensemble-average rate $\mathbb{E}_h\{c_{\text{com},k}^i\}$ to approximate $\tau_{\text{com},k}$ [17], where $\mathbb{E}_h\{\cdot\}$ is the expectation over $h$.

In the sequel, to gain useful insight, we consider one user for analysis.

3) **Capability of Resources:** We use the ratio of tiles in a segment that can be rendered and transmitted with configured resources (reflect by the data rate, computing rate, and duration $T_{cc}$) to measure the capability of the system for streaming tiles, which is [17]

$$C = \min \left\{ \frac{T_{cc}}{N_f M \left( \frac{s_{\text{com}}}{E_h\{c_{\text{com},k}^i\}} + \frac{s_{\text{cpt}}}{E_c\{c_{\text{cpt},k}\}} \right) + 1} \right\} \in [0, 1] \tag{1}$$

where $s_{\text{com}} = px_w \cdot px_h \cdot b / \gamma_c$ [19] is the number of bits in each tile for transmission, $s_{\text{cpt}} = px_w \cdot px_h \cdot b$ is the number of bits in a tile for rendering, $px_w$ and $px_h$ are the pixels in wide and high of a tile, $b$ is the number of bits per pixel relevant to color depth, and $\gamma_c$ is the compression ratio of video.

The number of tiles within SFOV is identical among $N_f$ frames in a segment, hence the ratio of the SFOV to the panoramic frame is also $C$, i.e., $C = \frac{A_{sv}}{4\pi}$, where $A_{sv}$ denotes the area of the SFOV, $4\pi$ is the surface area of the unit sphere.

**D. Streamed Field of View on Sphere**

SFOV contains all the tiles whose spherical distances to from their centers to $O_p$ are no more than a given value, then we can assume that the SFOV is also a spherical cap with center $O_p$, as shown in Fig. 2b. Then, the area of the SFOV is

$$A_{sv} = 2\pi \left( 1 - \cos \left( r_{sv} \right) \right) \tag{2}$$

where $r_{sv}$ is the cap radius of the SFOV.
By substituting (1) and \( C = \frac{A_{sv}}{4\pi} \) into (2), we obtain the radius of SFOV as
\[
r_{sv} = \arccos (1 - 2C) = \arccos \left( 1 - 2 \min \left\{ \frac{T_{cc}}{N_j M \left[ \frac{1}{\frac{1}{\sum_{c_{com}^{j,k}}}}} \right]}, 1 \right) \right) \in [0, \pi]
\] (3)

**Remark 1:** From (3), \( r_{sv} \) monotonically increases with the configured resources, \( E_{h} \{ c_{com}^{j,k} \}, \)
\( c_{cpt,k} \), and \( T_{cc} \). Then, \( r_{sv} \) can be used to reflect the amount of resources.

When \( r_{sv} = \pi \), according to (3) and (2), the configured resources achieve \( C = 1 \) and \( A_{sv} = 4\pi \), which indicates that the complete panoramic sphere are streamed. In this case, proactive VR streaming degrades into passive streaming and prediction is unnecessary [16]. When \( r_{sv} = 0 \), similarly, \( C = 0 \) and \( A_{sv} = 0 \), which indicates that no resources are configured and the panoramic frame is not streamed at all.

### E. Prediction Error: Spherical Distance

Spherical distance, known as orthodromic distance [2], [20], has been considered as the most appropriate metric to measure the viewpoint prediction error of FoV on the unit sphere [2]. As shown in Fig. 2c, the spherical distance from the real viewpoint \( O_v = (\theta_v, \varphi_v) \) to the predicted viewpoint \( O_p = (\theta_p, \varphi_p) \) can be expressed as
\[
e \triangleq d(O_v, O_p) = \arccos (\cos(\varphi_v) \cos(\varphi_p) \cos(|\theta_v - \theta_p|) + \sin(\varphi_v) \sin(\varphi_p)) \in [0, \pi]
\] (4)
where \( d(x, y) \) denotes the spherical distance between two points \( x \) and \( y \), \( \theta \) and \( \varphi \) are respectively the longitude and latitude of a point on a unit sphere [2], [21]. When \( e = 0 \), the predicted and real viewpoints coincide. When \( e = \pi \), the prediction error reaches the maximum.

After the user watches a video frame of a segment, the value of \( e \) can be calculated from (4) at the HMD.

### F. Quality of Experience

For proactive VR video streaming, there will be no latency if the SFOVs in a segment can be delivered before the playback of the segment. However, due to the prediction error and
insufficient resources, black holes will appear [22]. We use the correctly streamed portion in a FoV as the QoE metric [17], which can be expressed as the ratio of the overlapped area of FoV and SFOV to the area of FoV, as shown in Fig. 2c. Denote the overlapped area of FoV and SFOV as \( A_{ol} \in [0, \min\{A_{fov}, A_{sv}\}] \), then the QoE metric can be expressed as

\[
QoE \triangleq \frac{A_{ol}}{A_{fov}} \in [0, 100\%]
\]  

(5)

After the user watches a video frame of a segment, the QoE can be calculated at the HMD.

III. VIEWPOINT LEAKAGE PROBABILITY IN PROACTIVE VR VIDEO STREAMING

In this section, we analyze the viewpoint leakage issue in proactive VR video streaming. We find that although real viewpoint is unnecessary to be uploaded, it can still be inferred with some probability from the uploaded viewpoint prediction and prediction error or the QoE metric. Then, we define the viewpoint leakage probability to characterize the degree of inference accuracy.

A. Viewpoint Leakage in Proactive Streaming Procedure

In the proactive VR Video streaming procedure, the viewpoint may be leaked out during predictor training and online streaming. According to whether a predictor requires training and where the training and predicting are respectively executed, we provide seven cases in Table I.

TABLE I: Uploaded data from HMD in proactive VR video streaming procedure

| No. | Where to train | Where to predict | Predictor training | Online streaming | Adaptive streaming |
|-----|----------------|------------------|-------------------|-----------------|-------------------|
|     |                |                  |                   | Viewpoint observation | Viewpoint prediction | |
| 1   | MEC            | MEC              | Real viewpoint    | Real viewpoint   |                   | |
| 2   | MEC            | HMD              | Real viewpoint    | Predicted viewpoint |                   | |
| 3   | HMD \*         | HMD              |                | Predicted viewpoint |                   | Prediction error or QoE |
| 4   | HMD            | HMD              | Model parameters | Predicted viewpoint |                   | |
| 5   | HMD            | MEC              | Model parameters | Real viewpoint    |                   | |
| 6   | No need        | MEC              |                  | Real viewpoint    |                   | |
| 7   | No need        | HMD              |                  | Predicted viewpoint |                   | |

* This is training individually at each HMD for local predictors. The other cases that training at the HMD are federated.
We can find that although privacy-preserving approaches (i.e., federated training, training individually at each HMD, selecting training-free predictor, or local predicting) can avoid the real viewpoint leakage during the predictor training and viewpoint observation, the predicted viewpoint is still uploaded during viewpoint prediction in online streaming. With the uploaded prediction error or QoE, the real viewpoint may still be inferred. For example, when (1) \( e = 0 \) or (2) \( QoE = 100\% \) and the area of SFoV equals the area of FoV (i.e., \( r_{sv} = r_{fov} \)), \( e = 0 \). In both cases, one can infer that the predicted viewpoint is the real viewpoint. That is to say, the prediction error or QoE can be regarded as a signal to indicate how accurate the predicted viewpoint is. Then, a natural questions is, with predicted viewpoint at hand, with what probability the real viewpoint can be inferred from the prediction error or the QoE?

**B. \( \varepsilon \)-Viewpoint Leakage Probability**

We first define the viewpoint-sensitive neighborhood, viewpoint leakage event, and the possible viewpoint zone. Based on the above definitions, we define the \( \varepsilon \)-viewpoint leakage probability. Denote \( \mathcal{V} \) as the set of all points on a unit sphere.

**Definition 1:** \( \varepsilon \)-viewpoint-sensitive neighborhood \( \mathcal{N}(O_v, \varepsilon) \): A subset of \( \mathcal{V} \) where the spherical distance between every point in the subset and the real viewpoint is no larger than \( \varepsilon \), i.e.,

\[
\mathcal{N}(O_v, \varepsilon) \triangleq \{ x | d(x, O_v) \leq \varepsilon \text{ and } x \in \mathcal{V} \}, \quad \varepsilon \in [0, r_{fov}].
\]

\( \mathcal{N}(O_v, \varepsilon) \) is a region within FoV that a user does not willing to be leaked. The value of \( \varepsilon \) reflects the privacy requirement of the user. When \( \varepsilon = 0 \), the user has no privacy requirement. When \( \varepsilon = r_{fov} \), the whole FoV is required not to be leaked. We refer to \( \varepsilon \) as “the radius of the neighborhood”.

**Definition 2:** Possible viewpoint zone \( \mathcal{Z}_v \): A subset of \( \mathcal{V} \) that consists of all possible inferred viewpoints, given the predicted viewpoint \( O_p \) and the prediction error or the QoE.

If the prediction error or the QoE can be used for inferring real viewpoint accurately, the only one inferred viewpoint is the real viewpoint and \( \mathcal{Z}_v = \{ O_v \} \). If the prediction error or the
QoE cannot provide any useful information, the possibly inferred viewpoint can be arbitrary one viewpoint on the unit sphere and $\mathcal{Z}_v = \mathcal{V}$.

**Definition 3:** $\varepsilon$-viewpoint leakage event: When inferring the real viewpoint, the inferred viewpoint falls in the $\varepsilon$-viewpoint-sensitive neighborhood, i.e., $\hat{O}_v \in \mathcal{N}(O_v, \varepsilon)$, where $\hat{O}_v$ is the inferred viewpoint.

**Definition 4:** $\varepsilon$-viewpoint leakage probability: The probability that $\varepsilon$-viewpoint leakage event happens, which is

$$
\Pr \{ \hat{O}_v \in \mathcal{N}(O_v, \varepsilon) \} = \min \left\{ \frac{\lambda[\mathcal{N}(O_v, \varepsilon)]}{\lambda[\mathcal{Z}_v]}, 1 \right\}
$$

where $\lambda[\mathcal{X}]$ is the Lebesgue measure of $\mathcal{X}$. When $\mathcal{X}$ is a set of all viewpoints in a curve or a surface, $\lambda[\mathcal{X}]$ is the length of the curve or the area of the surface.

When the prediction error or the QoE can provide sufficient information such that the possible viewpoint zone is limited in the $\varepsilon$-viewpoint sensitive neighborhood, the real viewpoint is $\varepsilon$-leaked, the measure of the neighborhood of $O_v$ is no more than the measure of the possible viewpoint zone, i.e., $\lambda[\mathcal{N}(O_v, \varepsilon)] \geq \lambda[\mathcal{Z}_v]$, thus $\Pr \{ \hat{O}_v \in \mathcal{N}(O_v, \varepsilon) \} = 1$. When the prediction error or the QoE cannot provide any useful information such that the possible viewpoint zone is the whole sphere, the real viewpoint can be fully protected. In this case, the measure of the possible viewpoint zone is the surface area of the sphere, i.e., $\lambda[\mathcal{Z}_v] = 4\pi$, the measure of viewpoint sensitive neighborhood is the area of a spherical cap with center $O_v$ and radius $\varepsilon$, i.e., $\lambda[\mathcal{N}(O_v, \varepsilon)] = 2\pi(1-\cos(\varepsilon))$, then the minimal viewpoint leakage probability is $\Pr^{\min} = \frac{1-\cos(\varepsilon)}{2}$.

**Remark 2:** $\Pr \{ \hat{O}_v \in \mathcal{N}(O_v, \varepsilon) \}$ is a non-decreasing function of $\varepsilon$.

This is because $\Pr \{ \hat{O}_v \in \mathcal{N}(O_v, \varepsilon) \}$ is a non-decreasing function of $\lambda[\mathcal{N}(O_v, \varepsilon)]$, and $\lambda[\mathcal{N}(O_v, \varepsilon)]$ monotonically increases with $\varepsilon$. Remark 2 indicates that if a user demands larger viewpoint-sensitive neighborhood, then the viewpoint leakage probability increases.

In the sequel, we derive and analyze the $\varepsilon$-viewpoint leakage probability when uploading prediction error and QoE, respectively.
IV. VIEWPOINT LEAKAGE WHEN UPLOADING PREDICTION ERROR

In this section, we derive the probability when uploading prediction error, from which we find a conditional tradeoff between prediction performance and viewpoint privacy. Moreover, the targets of satisfying the viewpoint privacy requirement and maximizing the QoE are contradictory.

\( r = e \cdot \sin(\theta) \cdot r = \sin(e) \)

Possible viewpoint zone and \( \varepsilon \)-viewpoint-sensitive neighborhood.

(a) Possible viewpoint zone and \( \varepsilon \)-viewpoint-sensitive neighborhood.

(b) \( P_{e} \) vs. \( e \) with \( e = 0.4r_{vw} \).

(c) \( P_{e} \) vs. \( e \) and \( \varepsilon \)

Fig. 3: Viewpoint leakage when uploading \( e \).

A. \( \varepsilon \)-viewpoint Leakage Probability

When the server obtains the predicted viewpoint \( O_{p} \) and prediction error \( e \), the real viewpoint \( O_{v} \) can be inferred. As shown in Fig. 3a, when given \( O_{p} \) and \( e \), the possible viewpoint zone becomes a circle with center \( O_{e} \). The \( \varepsilon \)-viewpoint sensitive neighborhood becomes an arc of the circle with center \( O_{e} \). One can only infer that \( O_{v} \) is on a circle. In the sequel, we first derive the measures of the zone and neighborhood, respectively, based on which we obtain the probability.

The measure of \( Z_{v} \) is the circumference of the circle, i.e., \( \lambda[Z_{v}] = 2\pi r_{e} \), where \( r_{e} \) is the radius of the circle and is a function of \( e \) as derived in the following. The straight-line distance from arbitrary one viewpoint on the circle \( O_{a} \) to the center of the sphere \( O \) is the radius of the sphere \( r = 1 \). The angle between the line segment \( O_{p}O \) and the line segment \( O_{a}O \) is \( \theta \). When measured in radian, \( \theta = e \). The radius of the circle \( r_{e} = \sin(\theta) \cdot r = \sin(e) \). By substituting \( r_{e} = \sin(e) \) into \( \lambda[Z_{v}] = 2\pi r_{e} \), the measure of \( Z_{v} \) is \( \lambda[Z_{v}] = 2\pi \sin(e) \).
The measure of $\mathcal{N}(O_v, \varepsilon)$ is the arc length $2\varepsilon$. Since the length of the neighborhood is no more than the circumference of the circle, $\lambda[\mathcal{N}(O_v, \varepsilon)] = \min\{2\varepsilon, 2\pi \sin(e)\}$.

According to Definition 4, the $\varepsilon$-viewpoint leakage probability when prediction error is

$$\mathrm{Pr}_e \triangleq \min \left\{ \frac{\min\{2\varepsilon, 2\pi \sin(e)\}}{2\pi \sin(e)}, 1 \right\} = \min \left\{ \frac{\varepsilon}{\pi \sin(e)}, 1 \right\}$$

(7)

which increases as $\sin(e)$ decreases. Then, when $\sin(e)$ achieves the maximum, i.e., $e = 0.5\pi$, the minimal viewpoint leakage probability achieves as $\mathrm{Pr}_e^{\min} = \frac{\varepsilon}{\pi}$.

From (7), $\mathrm{Pr}_e = 1$ when $\pi \sin(e) \leq \varepsilon$, from which we obtain the range of $e$ as

$$e \in [0, \arcsin(\frac{\varepsilon}{\pi})] \cup [\pi - \arcsin(\frac{\varepsilon}{\pi}), \pi]$$

(8)

When $e$ falls in this range, taking arbitrary viewpoint in the possible viewpoint zone as an inferred viewpoint, $\varepsilon$-viewpoint leakage event happens.

**B. Relation between Prediction Performance and Viewpoint Privacy**

From (8), when $e = \arcsin(\frac{\varepsilon}{\pi})$ or $\pi - \arcsin(\frac{\varepsilon}{\pi})$, $\mathrm{Pr}_e = 1$. From (7), as $e$ increases from $\arcsin(\frac{\varepsilon}{\pi})$ to $\pi - \arcsin(\frac{\varepsilon}{\pi})$, $\mathrm{Pr}_e$ first decreases and then increases, and is mirror symmetric with respect to (w.r.t.) $e = 0.5\pi$.

When $e \in [\arcsin(\frac{\varepsilon}{\pi}), 0.5\pi]$, as $e$ increases, the viewpoint leakage probability $\mathrm{Pr}_e$ decreases. This indicates that one can “trade prediction performance for privacy”. When $e \in [0.5\pi, \pi - \arcsin(\frac{\varepsilon}{\pi})]$, $\mathrm{Pr}_e$ decreases as $e$ decreases. This shows the consistency between prediction performance and viewpoint privacy.

In Fig. 3b and 3c, we provide the value of $\mathrm{Pr}_e$ obtained from (7).

The following corollary provides the relation of prediction error with the viewpoint privacy requirement.
**Corollary 1:** If the required maximal viewpoint leakage probability is $Pr_u$, then the required range of prediction error for satisfying $Pr_e \leq Pr_u$ is

$$e \in \begin{cases} [0, \pi], & Pr_u = 1, \\ [e_{\text{min}}^u, e_{\text{max}}^u], & Pr_u \in [Pr_{e_{\text{min}}}^u, 1), \\ \text{Infeasible}, & Pr_u \in [0, Pr_{e_{\text{min}}}^u). \end{cases} \quad (9)$$

where $e_{\text{min}}^u \triangleq \arcsin\left(\frac{e}{Pr_e \pi}\right)$ and $e_{\text{max}}^u \triangleq \pi - \arcsin\left(\frac{e}{Pr_e \pi}\right)$. When $Pr_u \in [Pr_{e_{\text{min}}}^u, 1)$, the range of prediction error for satisfying $Pr_e > Pr_u$ is

$$e \in \left[0, e_{\text{min}}^u\right) \cup \left(e_{\text{max}}^u, \pi\right] \quad (10)$$

**Proof:** See Appendix A. \qed

From (9), we can find that if a user has no privacy viewpoint requirement, i.e., $Pr_u = 1$, then the prediction error should be minimized for maximizing the QoE. When $Pr_u < 1$, there is a contradiction between the target of maximizing the QoE and the target of satisfying the viewpoint privacy requirement. In Fig. 3b, we provide an example when $Pr_u = 0.2$, in this case, $e_{\text{min}}^u = 0.19\pi$ and $e_{\text{max}}^u = 0.81\pi$.

**V. Viewpoint Leakage Probability when Uploading the QoE**

In this section, we derive the $\varepsilon$-viewpoint leakage probability when uploading the value of QoE. Although QoE cannot be used directly for inferring $O_v$, the prediction error may be inferred from the QoE. We first derive QoE as a function of configured resources and prediction error, from which the prediction error can be inferred. Then, we derive the viewpoint leakage probability.
A. QoE as a Function of Resources and Prediction Error

**Proposition 1:** The expressions of QoE with respect to $r_{sv}$ and $e$ are as follows.

\begin{align}
QoE &= 0, \text{ if } r_{sv} = 0 \text{ (i.e., no streaming).} \quad (11a) \\
QoE &= 100\%, \text{ if } r_{sv} = \pi \text{ (i.e., streaming the sphere).} \quad (11b) \\
QoE &= 100\%, \text{ if } r_{sv} \geq r_{fov} + e \text{ and } 0 < r_{sv} < \pi \text{ (i.e., FoV} \subset S\text{FoV).} \quad (11c) \\
QoE &= \frac{1 - \cos(r_{sv})}{1 - \cos(r_{fov})}, \text{ if } r_{fov} \geq r_{sv} + e \text{ and } 0 < r_{sv} < \pi \text{ (i.e., } S\text{FoV} \subset \text{FoV).} \quad (11d) \\
QoE &= 0, \text{ if } e \geq r_{fov} + r_{sv} \text{ and } 0 < r_{sv} < \pi \text{ (i.e., } \text{FoV} \cap S\text{FoV} = \emptyset). \quad (11e) \\
QoE &= \frac{-\cos(r_{sv}) - \cos(r_{fov})}{1 - \cos(r_{fov})}, \text{ if } r_{fov} + r_{sv} + e \geq 2\pi \text{ and } 0 < r_{sv} < \pi \text{ (i.e., } S\text{FoV-C} \subset \text{FoV).} \quad (11f) \\
QoE &= A_{ol}^{rm}(r_{sv}, e) \frac{2\pi}{2\pi(1 - \cos(r_{fov}))}, \text{ if } r_{sv} \in [r_{sv,min}^{rm}(e), r_{sv,max}^{rm}(e)] \text{ and } 0 < r_{sv} < \pi \text{ (i.e., remaining case),} (11g)
\end{align}

where

\begin{align}
A_{ol}^{rm}(r_{sv}, e) &\triangleq 2\pi - 2\pi \cos(r_{sv}) - 2\pi \cos(r_{fov}) - 2\arccos\left(\frac{\cos(e) - \cos(r_{sv}) \cos(r_{fov})}{\sin(r_{sv}) \sin(r_{fov})}\right) \\
&+ 2\cos(r_{sv}) \arccos\left(\frac{-\cos(r_{fov}) + \cos(e) \cos(r_{sv})}{\sin(e) \sin(r_{sv})}\right) \\
&+ 2\cos(r_{fov}) \arccos\left(\frac{-\cos(r_{sv}) + \cos(e) \cos(r_{fov})}{\sin(e) \sin(r_{fov})}\right),
\end{align}

$r_{sv,min}^{rm}(e) \triangleq |r_{fov} - e|$, and $r_{sv,max}^{rm}(e) \triangleq \min\{r_{fov} + e, 2\pi - (r_{fov} + e)\}$.

Proof: See Appendix B. \hfill \square

In the proposition, $\text{FoV} \subset S\text{FoV}$, $S\text{FoV} \subset \text{FoV}$, $\text{FoV} \cap S\text{FoV} = \emptyset$, and $S\text{FoV-C} \subset \text{FoV}$ respectively denote the cases that FoV being included in SFoV, SFoV being included in the FoV, no intersection between FoV and SFoV, and the complement set of SFoV being included in FoV. These four cases are illustrated in Fig. (4a)-(4d).

**Remark 3:** In the four cases that $\text{FoV} \subset S\text{FoV}$, $S\text{FoV} \subset \text{FoV}$, $\text{FoV} \cap S\text{FoV} = \emptyset$, or $S\text{FoV-C} \subset \text{FoV}$, QoE has no relation with prediction error $e$. \hfill \square
Fig. 4: Four cases when $r_{sv} \in (0, \pi)$, and QoE v.s. $r_{sv}$ and $e$.

**Remark 4:** In the remaining case, $\text{QoE} = \frac{A_{mf}(r_{sv}, e)}{2\pi(1-\cos(r_{fov}))}$ is a strictly monotonically decreasing and increasing function of $e$ and $r_{sv}$, respectively.

To visualize the impact of resources $r_{sv}$ and prediction error $e$ on the QoE under different cases and the conditions of these cases, we provide the QoE obtained from (11) given all possible values of $r_{sv}$ and $e$ in Fig. 4e. We can observe that in the first four cases, QoE remains constant as the increase of $e$, as indicated in Remark 3. In the remaining case QoE monotonically increases as the decrease of $e$ or the increase of $r_{sv}$, as indicated in Remark 4.
B. Infer Prediction Error from QoE

From subsection II-A, we can find that except the QoE, the server also knows the predicted viewpoint $O_p$, the radius of the FoV $r_{fov}$, and the radius of the SFoV $r_{sv}$.

The following proposition provides the prediction error inferred from these information.

**Proposition 2:** When $r_{sv} \in (0, \pi)$, given the value of QoE, the range or value of $e$ is

\begin{align}
    e &\in [0, r_{sv} - r_{fov}], \text{ if } \text{QoE} = 100\% \text{ (i.e., } \text{FoV} \subset \text{SFoV}) \quad (13a) \\
    e &\in [0, r_{fov} - r_{sv}], \text{ if } \text{QoE} = \frac{1 - \cos(r_{sv})}{1 - \cos(r_{fov})} \text{ (i.e., } \text{SFoV} \subset \text{FoV}) \quad (13b) \\
    e &\in [r_{fov} + r_{sv}, \pi], \text{ if } \text{QoE} = 0 \text{ (i.e., } \text{FoV} \cap \text{SFoV} = \emptyset) \quad (13c) \\
    e &\in [2\pi - r_{fov} - r_{sv}, \pi], \text{ if } \text{QoE} = -\frac{\cos(r_{sv}) - \cos(r_{fov})}{1 - \cos(r_{fov})} \text{ (i.e., } \text{SFoV-C} \subset \text{FoV}) \quad (13d) \\
    e &= e^{bi}, \text{ otherwise (i.e., remaining case)} \quad (13e)
\end{align}

where $e^{bi}$ is the value of $e$ determined by (11g) with bisection search.

Proof: See Appendix C.

\[\square\]

C. $\varepsilon$-Viewpoint Leakage Probability

Based on Propositions 1 and 2 as well as Remark 4, we can derive the $\varepsilon$-viewpoint leakage probability, denoted as $Pr_{q}$. When $r_{sv} = 0$ or $\pi$, neither the range or the value of $e$ can be inferred. Then, the $\varepsilon$-viewpoint leakage probability achieves its minimum as $Pr_{q}^{\text{min}} = Pr_{q}^{\text{min}} = \frac{1 - \cos(\varepsilon)}{2}$.

In the following, we derive the probability in the other cases.

1) $\text{FoV} \subset \text{SFoV}$: From (13a), $e \in [0, r_{sv} - r_{fov}]$. As shown in Fig. 5a, given the predicted viewpoint $O_p$, the possible viewpoint zone $Z_v$ is a spherical cap with radius $r_z = r_{sv} - r_{fov} \geq 0$ and area $A_{Z_v} = 2\pi(1 - \cos(r_z))$. The $\varepsilon$-viewpoint-sensitive neighborhood is a spherical cap with center $O_v$ and radius $\varepsilon$, whose area is $A_{N(O_v,\varepsilon)} = 2\pi(1 - \cos(\varepsilon))$. Then, according to Definition 4, the $\varepsilon$-viewpoint leakage probability is

\[Pr_{q}^{\text{VF} \subset \text{SV}}(\varepsilon, r_{sv}) = \min \left\{ \frac{2\pi(1 - \cos(\varepsilon))}{2\pi(1 - \cos(r_z))}, 1 \right\} = \min \left\{ \frac{1 - \cos(\varepsilon)}{1 - \cos(r_{sv} - r_{fov})}, 1 \right\}\]
2) \( S\text{FoV}\subset\text{FoV}: \) From (13b), \( e \in [0, r_{\text{fov}} - r_{sv}] \). As shown in Fig. 5a, the only difference from the case where \( \text{FoV}\subset S\text{FoV} \) is the radius of the possible viewpoint zone becomes \( r_z = r_{\text{fov}} - r_{sv} \geq 0 \). Then, the \( \varepsilon \)-viewpoint leakage probability is

\[
\Pr_{q}^{SV\subset V}(\varepsilon, r_{sv}) = \min \left\{ \frac{1 - \cos(\varepsilon)}{1 - \cos(r_{\text{fov}} - r_{sv})}, 1 \right\}
\]

3) \( \text{FoV}\cap S\text{FoV}=\emptyset \): From (13c), \( e \in [r_{\text{fov}} + r_{sv}, \pi] \). As shown in Fig. 5b, \( Z_v \) is a spherical cap with radius \( r_z = \pi - (r_{sv} + r_{\text{fov}}) \geq 0 \). Then, the \( \varepsilon \)-viewpoint leakage probability is

\[
\Pr_{q}^{V\cap SV=\emptyset}(\varepsilon, r_{sv}) = \min \left\{ \frac{1 - \cos(\varepsilon)}{1 - \cos(\pi - (r_{sv} + r_{\text{fov}}))}, 1 \right\} = \min \left\{ \frac{1 - \cos(\varepsilon)}{1 + \cos(r_{sv} + r_{\text{fov}})}, 1 \right\}
\]

4) \( S\text{FoV-C}\subset\text{FoV} \): From (13d), \( e \in [2\pi - r_{\text{fov}} - r_{sv}, \pi] \). As shown in Fig. 5b, the only difference from the case that \( \text{FoV}\cap S\text{FoV}=\emptyset \) is that the radius of the possible viewpoint zone becomes \( r_z = \pi - (2\pi - r_{\text{fov}} - r_{sv}) = r_{\text{fov}} + r_{sv} - \pi \geq 0 \). The \( \varepsilon \)-viewpoint leakage probability is

\[
\Pr_{q}^{SV-C\subset V}(\varepsilon, r_{sv}) = \min \left\{ \frac{1 - \cos(\varepsilon)}{1 - \cos(r_{\text{fov}} + r_{sv} - \pi)}, 1 \right\} = \min \left\{ \frac{1 - \cos(\varepsilon)}{1 + \cos(r_{sv} + r_{\text{fov}})}, 1 \right\}
\]

The viewpoint leakage probability in the four cases can be unified as

\[
\Pr_{q}(\varepsilon, r_{sv}) = \min \left\{ \frac{1 - \cos(\varepsilon)}{1 - \cos(r_z)}, 1 \right\}
\]
where $r_z = |r_{sv} - r_{fov}|$ for $\text{FoV} \subset \text{SFoV}$ and $\text{SFoV} \subset \text{FoV}$, $r_z = |\pi - r_{sv} - r_{fov}|$ for $\text{FoV} \cap \text{SFoV} = \emptyset$ and $\text{SFoV-C} \subset \text{FoV}$. With $r_z$, the range of $e$ can be expressed as follows.

If $\text{FoV} \subset \text{SFoV}$ or $\text{SFoV} \subset \text{FoV}$, $e \leq r_z$. If $\text{FoV} \cap \text{SFoV} = \emptyset$ or $\text{SFoV-C} \subset \text{FoV}$, $e \geq \pi - r_z$. (15)

5) Remaining case: Since the value of $e$ can be obtained from (11g) with bisection search according to Remark 4, the $\varepsilon$-viewpoint leakage probability degrades into the probability when uploading the prediction error. As in (7), the $\varepsilon$-viewpoint leakage probability in this case is $\Pr_{q_{RM}} = \Pr_{e} = \min \left\{ \frac{\varepsilon}{\pi \sin(e)}, 1 \right\}$. As in (11g), the condition of the case is $r_{sv} \in \left[ r_{rm_{sv}}(e), r_{rm_{sv}}(e) \right]$. 

Remark 5: For the four cases, it is shown from (14) and (15) that when the resources are fixed and prediction error $e \leq r_z$ or $e \geq \pi - r_z$, the viewpoint leakage probability does not depend on $e$. For the remaining case, when the prediction error is fixed and the resources $r_{sv} \in \left[ r_{sv_{min}}(e), r_{sv_{max}}(e) \right]$, the viewpoint leakage probability does not depends on $r_{sv}$. □

The viewpoint leakage probability are illustrated in Fig. 6.

![Fig. 6](image)

(a) Frontal view. (b) Dorsal view.

Fig. 6: $\varepsilon$-viewpoint leakage probability v.s. $r_{sv}$ and $e$, $\varepsilon = 0.4r_{fov}$.

D. When the $\varepsilon$-Viewpoint Leakage Probability Achieves the Maximum and Minimum?

Since for the cases where $r_{sv} = 0$ or $\pi$, the global minimal value of $\Pr_{q_{1}}$ can be achieved, we consider the cases where $r_{sv} \in (0, \pi)$. 

DRAFT March 8, 2022
For the four cases, the conditions of $r_{sv}$ and $e$ that achieve maximal and infimum of viewpoint leakage probabilities can be obtained from (14) and (15), the conditions are listed in Table II.

To show the relation between viewpoint leakage and configured resources, we also provide the monotonicity of $Pr_q$ w.r.t. $r_{sv}$.

**TABLE II: Maximal and infimum of $Pr_q$ in the four cases when $r_{sv} \in (0, \pi)$**

| Case                  | $Pr_q = 1$ | Monotonicity of $Pr_q$ w.r.t. $r_{sv}$ when $Pr_q < 1$ | $Pr_q = Pr_q^{inf}$ | Condition of $r_{sv}$ |
|-----------------------|------------|--------------------------------------------------------|----------------------|------------------------|
| $FoV \subset SFoV$   | $e \leq \varepsilon$ | Increasing $\frac{1 - \cos(e)}{1 + \cos(\delta_{fov})} > Pr_{min}$ | $\pi - r_{sv} \leq \delta^{**}$ |                        |
| $SFoV \subset FoV$   | $|r_{sv} - r_{fov}| \leq \varepsilon$ | Decreasing $\frac{1 - \cos(e)}{1 + \cos(\delta_{fov})} > Pr_{min}$ | $r_{sv} \leq \delta$ |                        |
| $FoV \cap SFoV = \emptyset$ | $e \geq \pi - \varepsilon$ | Decreasing $\frac{1 - \cos(e)}{1 + \cos(\delta_{fov})} > Pr_{min}$ | $r_{sv} \leq \delta$ |                        |
| $SFoV-C \subset FoV$ | $|\pi - r_{sv} - r_{fov}| \leq \varepsilon$ | Increasing $\frac{1 - \cos(e)}{1 + \cos(\delta_{fov})} > Pr_{min}$ | $\pi - r_{sv} \leq \delta$ |                        |

$^*$ $Pr_q^{inf}$ is the infimum of $Pr_q$. The minimum of $Pr_q$ does not exist. $^{**}\delta$ is an arbitrarily small positive number.

One condition for viewpoint leakage probability achieving one, $e \leq \varepsilon$ or $e \geq \pi - \varepsilon$, indicates that either the predicted viewpoint or its symmetry point is in the $\varepsilon$-viewpoint sensitive neighborhood. The other condition, $r_z \leq \varepsilon$, where $r_z = |r_{sv} - r_{fov}|$ or $|\pi - r_{sv} - r_{fov}|$, indicates that the configured resources make the difference between the radius of SFoV or SFoV-C and the radius of FoV no larger than $\varepsilon$. Specifically, for $FoV \subset SFoV$ and $SFoV \subset FoV$, $e = d(O_p, O_v) \leq \varepsilon$. This indicates that the predicted viewpoint lies in the viewpoint sensitive neighborhood. For $FoV \cap SFoV = \emptyset$ and $SFoV-C \subset FoV$, the arc length $d(O_v, \tilde{O}_p) = d(O_p, \tilde{O}_p) - d(O_p, O_v) = \pi - e \leq \varepsilon$, where $\tilde{O}_p$ is the symmetry point of $O_p$ w.r.t. the center of the unit sphere. This indicates that the symmetry point of $O_p$ is in the viewpoint sensitive neighborhood. However, low or high prediction error (i.e., $e \leq \varepsilon$ or $e \geq \pi - \varepsilon$) may not lead to $Pr_q = 1$. Only if the configured resources satisfy $|r_{sv} - r_{fov}| \leq \varepsilon$ or $|\pi - r_{sv} - r_{fov}| \leq \varepsilon$, $Pr_q = 1$.

When $|r_{sv} - r_{fov}| > \varepsilon$ or $|\pi - r_{sv} - r_{fov}| > \varepsilon$, i.e., $Pr_q < 1$, the viewpoint leakage probability decreases as $r_{sv} \rightarrow 0$ or $r_{sv} \rightarrow \pi$. Then, viewpoint leakage probability achieves the minimum when $r_{sv} = 0$ or $\pi$, which however is not defined in the four cases. Hence, the infimum rather than the minimum exist. Furthermore, the infima of viewpoint leakage probability in the four
cases are larger than the global minimum, $Pr_{\text{min}}$. This is because the radius of possible viewpoint zone, $r_z$, is not continuous when $r_{sv} = 0$ or $\pi$. For example, when $\text{FoV} \subset \text{SFoV}$, the limit of $r_z$ from the left is $\lim_{r_{sv} \to \pi^-} r_z = \pi - r_{fov}$. However, when $r_{sv} = \pi$ that is not belongs to the case $\text{FoV} \subset \text{SFoV}$, the real viewpoint is arbitrary on the sphere and the radius of possible viewpoint zone becomes $r_z = \pi$.

For the remaining case, since $e$ can be determined from QoE and $r_{sv}$, the viewpoint leakage probability achieves the maximum when $e \in [0, \arcsin(\frac{\xi}{\pi})] \cup [\pi - \arcsin(\frac{\xi}{\pi}), \pi]$, as shown in (8). As analyzed in subsection IV-A, $Pr_{\text{min}} = \frac{\xi}{\pi}$, which is achieved when $e = 0.5\pi$. With $\xi \in [0, r_{fov}]$ and $r_{fov} \in [0, 0.5\pi]$, after some regular derivations, we obtain that $\frac{\xi}{\pi} > \frac{1 - \cos(e)}{2}$, i.e., the minimal viewpoint leakage probability when uploading prediction error is larger than the minimal viewpoint leakage probability when uploading the QoE.

**E. Relation between QoE, Resources and $\varepsilon$-viewpoint Leakage Probability**

For the four cases, when $|r_{sv} - r_{fov}| \leq \varepsilon$ or $|\pi - r_{sv} - r_{fov}| \leq \varepsilon$, i.e., $Pr_q = 1$, the variation of resources does not affect $Pr_q$, otherwise $Pr_q$ is a monotonic function of $r_{sv}$. When $Pr_q$ decreases as $r_{sv}$ increases, the resources are sacrificed to protect the viewpoint privacy and improve the QoE. When $Pr_q$ decreases as $r_{sv}$ decreases, unless that the QoE is already zero, the QoE is sacrificed to save resources and protect the viewpoint privacy. That is to say, unless $Pr_q = 1$ or QoE = 0, there is a tradeoff between QoE, resources, and viewpoint privacy, because the viewpoint leakage probability is a monotonic function of resources. To visualize the relation, we provide values of QoE and $Pr_q$ in the four cases in Fig. 7.

**Remark 6:** The tradeoff does not exist when (a) $r_{sv} \leq r_{fov} + \varepsilon$ in $\text{FoV} \subset \text{SFoV}$, (b) $r_{sv} \geq r_{fov} - \varepsilon$ in $\text{SFoV} \subset \text{FoV}$, (c) $r_{sv} \leq \pi - r_{fov} + \varepsilon$ in $\text{SFoV-C} \subset \text{FoV}$, (d) in $\text{FoV} \cap \text{SFoV}=\emptyset$ for the four case, as shown in Fig. 7, or (e) in the remaining case according to Remark 5. We refer to (a)-(e) as the “exceptional cases”. From (11), we can find that the conditions of $\text{FoV} \subset \text{SFoV}$, $\text{SFoV} \subset \text{FoV}$, $\text{SFoV-C} \subset \text{FoV}$, $\text{FoV} \cap \text{SFoV}=\emptyset$, and the remaining case are all related to $r_{sv}$ and $e$, hence the conditions of exceptional cases are also related to $r_{sv}$ and $e$. $\square$
Fig. 7: $\varepsilon$-viewpoint leakage probability and QoE in the four cases when $r_{sv} \in (0, \pi)$. For Fig. 7a and 7b, $e = 0.1\pi$, for Fig. 7c and 7d, $e = 0.9\pi$.

VI. TRACE-DRIVEN SIMULATION RESULTS

In this section, we show that although the tradeoff between viewpoint privacy and prediction performance, resources, or QoE is conditional theoretically, with state-of-the-art predictor, the tradeoff exists in most case.

First, we consider the viewpoint prediction on a real dataset [1], where 300 traces of viewpoints
from 30 users watching 10 VR videos are used for training and testing predictors. $r_{fov} = 50^\circ$ [1], [2], the total traces are randomly split into training and test sets with ratio 8:2 [2]. The number of traces in the test set is $n_{\text{trace}} = 30 \times 10 \times 0.2 = 60$. For each trace with playback duration 60 s, the viewpoint is sampled five times per second [2]. The total proactive streaming time is set as $T_{ps} = 2$ s, the playback duration of a segment is $T_{seg} = 1$ s [17]. The initial two segments are streamed in passive manner. Then, the number of viewpoint samples in each trace to be predicted is $n_{sp} = 60 \times 5 - 2 \times 5 = 290$. The total number of viewpoint samples to be predicted is $N_{sp} = n_{\text{trace}} \times n_{sp} = 17400$.

We use the deep-position-only predictor, which achieve the state-of-the-art accuracy for the dataset according to evaluation in [2]. The predictor employs a sequence-to-sequence long short term memory (LSTM)-based architecture, which uses the past viewpoint positions as input to predict the future positions [2]. In the initial setting of the predictor, the prediction window is right after the observation window. To reserve time for computing and communication, we tailor the predictor by setting the duration between the end of the observation window and the beginning of the prediction windows as $T_{cc} = 1$ s, and set the durations of observation and prediction windows as $T_{obw} = 1$ s and $T_{pdw} = T_{seg} = 1$ s, respectively. To protect the viewpoint, we consider training and predicting at the HMD, which is case 4 in Table I. When training predictor, we consider a classical federated learning algorithm, FederatedAveraging [23]. The settings of the federated learning are as follows. In each round, all of $K = 30$ users update the model parameters of the predictor. The number of local epochs for each user is $E_l = 50$, the number of communication rounds is $R = 10$. The weighting coefficient of the $k$th user on the model parameter is $c_k = \frac{n_k}{N_{\text{train}}}$, where $N_{\text{train}} = 300 \times 0.8 = 240$ is the total number of traces in the training set, and $n_k$ is the number of video traces of the $k$th user in the training set. Due to

\footnote{According to the analysis in [2], [20], the traces of the first 20 users in the dataset have mistakes, thus we only use the traces of the other 30 users.}
the random division of training and test sets, $n_k$ varies from 6 to 10. We refer to the predictor as **tailored federated position-only** predictor. Other details and hyper-parameters of the tailored predictor are the same as the **deep-position-only** predictor [2]. To gain useful insight, we assume that all users have identical privacy requirement $(\varepsilon, \text{Pr}_{u_e})$ for all videos. After making prediction, the prediction error of each viewpoint sample $e_i, i=1, \ldots, N_{sp}$ on the test set can be obtained.

A. Tradeoff between Prediction Performance and Viewpoint Privacy When Uploading Prediction Error

To evaluate the tradeoff, we first provide the ratio of privacy requirement violation. Given arbitrary privacy requirement $(\varepsilon, \text{Pr}_{u_e})$, the ratio on the test set can be obtained from (9) as

$$
\gamma = \begin{cases} 
0, & \text{Pr}_{u_e} = 1, \\
\frac{\sum_{i=1}^{N_{sp}} \mathbb{1}(e_i \notin [e_{\text{min}}^u, e_{\text{max}}^u])}{N_{sp}}, & \text{Pr}_{u_e} \in [\text{Pr}_{e_{\text{min}}^u}, 1), \\
1, & \text{Pr}_{u_e} \in [0, \text{Pr}_{e_{\text{min}}^u}).
\end{cases}
$$

where $\mathbb{1}(\cdot)$ is the indicator function. In Fig. 8a, we provide the results of $\gamma$. We can observe that when the required viewpoint leakage probability is small, e.g., $\text{Pr}_{u_e} = 20\%$, as the radius of viewpoint-sensitive neighborhood $\varepsilon$ increases, the ratio of requirement violation eventually approach to one. Even with a more relaxed requirement, e.g., $\text{Pr}_{u_e} = 50\%$ and $\varepsilon = r_{fov}$, the ratio can still achieve $\gamma = 0.721$. That is to say, with the state-of-the-art viewpoint predictor, the privacy requirement cannot be satisfied in high ratio.

When $\text{Pr}_{u_e} \in [\text{Pr}_{e_{\text{min}}^u}, 1)$, we can find from (10) that both small and large prediction errors cause the violation of privacy requirement. To show the dominant factor, we provide the values of $\gamma_{e \leq e_{\text{min}}^u} \triangleq \frac{\sum_{i=1}^{N_{sp}} \mathbb{1}(e_i \in [0, e_{\text{min}}^u])}{N_{sp}}$ and $\gamma_{e \geq e_{\text{max}}^u} \triangleq \frac{\sum_{i=1}^{N_{sp}} \mathbb{1}(e_i \in [e_{\text{max}}^u, \pi])}{N_{sp}}$ in Fig. 8b. We can observe that $\gamma_{e \leq e_{\text{min}}^u} \gg \gamma_{e \geq e_{\text{max}}^u}$. For example, when $\varepsilon = 0.4r_{fov}$ and $\text{Pr}_{u_e} = 20\%$, $\gamma_{e \leq e_{\text{min}}^u} = 0.717$ and $\gamma_{e \geq e_{\text{max}}^u} = 0.004$. This indicates that with the state-of-the-art predictor, the primary reason that privacy requirement cannot be satisfied is the small prediction error rather than the large prediction error. Then, to satisfy the privacy requirement (i.e., to reduce the viewpoint leakage
Fig. 8: Ratios of privacy requirement violation, $e \in [0, e_{\text{min}}^u)$ and $e \in (0, e_{\text{max}}^u, \pi)$ on the test set when uploading $e$.

probability), the prediction performance should be sacrificed rather than improved. That is to say, the tradeoff exists in most cases.

B. Tradeoff between Average Viewpoint Leakage Probability, Resources, and Average QoE When Uploading the QoE

For the viewpoint leakage probability when uploading QoE, the tradeoff does not exist in exceptional cases and the conditions of these cases are all related to prediction error and resources, according to Remark 6. Since the prediction error rather than the resources is hard to be controlled, to investigate the overall tradeoff, we evaluate the relation between average viewpoint leakage probability, resources, and average QoE taken over the prediction error. To this end, we first analyze the relation between average viewpoint leakage probability and resources, based on which we show that the overall tradeoff exist in most cases.

Given $r_{sv}$, the average viewpoint leakage probability over $e$ can be obtained as

\[
\overline{Pr_q} = Pr_q^{V \subset SV} \cdot \gamma_{V \subset SV} + Pr_q^{SV \subset V} \cdot \gamma_{SV \subset V} + Pr_q^{V \subset SV = \emptyset} \cdot \gamma_{V \subset SV = \emptyset} + Pr_q^{SV \subset V = \emptyset} \cdot \gamma_{SV \subset V = \emptyset}
\]
\[
\frac{1}{N_{sp}} \sum_{i=1}^{N_{sp}} \Pr_{q}^{RM}(e_i) \cdot 1 (r_{sv} \in [r_{sv,min}^{RM}(e_i), r_{sv,max}^{RM}(e_i)])
\]

where \(\gamma^X\) is the ratio of case \(X\) on the test set. For example, from (11), \(\gamma^{VCSV} = \frac{1}{N_{sp}} \sum_{i=1}^{N_{sp}} 1 (r_{sv} \geq r_{fov} + e_i)\). Except the ratios of the four cases, the ratio of the remaining case can be obtained as

\[
\gamma^{RM} = \frac{1}{N_{sp}} \sum_{i=1}^{N_{sp}} 1 (r_{sv} \in [r_{sv,min}^{RM}(e_i), r_{sv,max}^{RM}(e_i)])
\]

We provide the value of \(\Pr_{q}^{RM}\) in Fig. 9a. We can observe that given arbitrary \(\varepsilon\), the range of \(r_{sv}\) can be divided in two increasing regions (\(I_1\) and \(I_2\)), two decreasing regions (\(D_1\) and \(D_2\)), and a constant region (\(C\)). Since except \(r_{sv} \in C\), \(\Pr_{q}^{RM}\) is a monotonic function of \(r_{sv}\), we can infer that overall tradeoff exists in most cases. We verify the finding in Fig. 9b where \(\varepsilon = 0.4r_{fov}\). Since the relation of \(\Pr_{q}^{RM}\) with \(r_{sv}\) is similar for other values of \(\varepsilon\) as shown in Fig. 9a, the tradeoff exists except \(r_{sv} \in C\) for arbitrary \(\varepsilon\).

To explain why the exceptional cases are reduced from five cases in Remark 6 to \(r_{sv} \in C\), we provide the values of average viewpoint leakage probabilities in Fig. 9c and the ratio of each case in Fig. 9d, from which we observe the following results.

(1) In Fig. 9d, the value of \(\gamma^{SV \subset C \backslash V}\) and \(\gamma^{V \backslash SV = \emptyset}\) is small. This is because the two cases happens when the prediction error is large according to (15) and the ratio of large prediction error is small for the state-of-the art predictor, as shown in Fig. (8b). That is to say, the exceptional cases (c) and (d) in Remark 6 can be omitted because most of prediction errors are small.

(2) In Fig. 9c, when in the exceptional cases (a), (b), and (e) in Remark 6, although the variation of \(r_{sv}\) does not affect \(\Pr_{q}^{RM}\), \(\Pr_{q}^{V \subset SV}\), \(\Pr_{q}^{SV \subset V}\), and \(\Pr_{q}^{RM}\) monotonically increases, decreases, first increases then decreases as \(r_{sv}\), respectively. From Fig. 9d we can observe that this is because the monotonicity of the ratios \(\gamma^{VCSV}\), \(\gamma^{SV \subset V}\), and \(\gamma^{RM}\) respectively, which lead to the tradeoff. That is to say, the exceptional cases (a), (b), and (e) in Remark 6 can be omitted because the relation of \(\gamma^{VCSV}\), \(\gamma^{SV \subset V}\), and \(\gamma^{RM}\) with \(r_{sv}\).

(3) When \(r_{sv} \in C\), the increase and decrease of \(\Pr_{q}^{RM}\) are respectively counteracted by the
decrease of $P_{r_{q,SV}}^{SV<V}$ and the increase of $P_{r_{q,V}}^{V<SV}$. This lead the new exception.

Since the overall tradeoff exist except $r_{sv} \in C$, we can find from Fig. 9a and 9b that to reduce the average viewpoint leakage probability, either the resources or the average QoE should be sacrificed for most cases.

Fig. 9: Average viewpoint leakage probabilities, QoE, and ratio of each cases when uploading QoE, $\varepsilon = 0.4r_{fov}$ in Fig. (b), (c), and (d).
VII. Conclusion

In this paper, we studied the viewpoint leakage in proactive VR streaming and found that existing privacy-preserving approaches cannot avoid leakage. We defined the viewpoint leakage probability and derived the probability when uploading prediction error and QoE. When uploading prediction error, if the prediction error is no larger than half of the maximum, there is a tradeoff between the prediction performance and viewpoint leakage probability, otherwise the tradeoff does not exist. The targets of maximizing the QoE and satisfying viewpoint privacy requirement are contradictory. When uploading the QoE, the value or the range of prediction error can be inferred, the probability in the former case degrades into the probability when uploading the prediction error. For the latter case, the probability achieves one when two conditions hold simultaneously. (1) Either the predicted viewpoint or its symmetry point is in the viewpoint sensitive neighborhood. (2) The radius of FoV no larger than the radius of viewpoint sensitive neighborhood. Besides, unless the probability remains one or the QoE remains zero, there is a tradeoff between QoE, resources, and viewpoint privacy probability. To reduce the probability, either the QoE or resources should be sacrificed.

Simulation with the state-of-the-art predictor over a real dataset shows that when uploading the prediction error, the privacy requirement cannot be satisfied in high ratio. To reduce the viewpoint leakage probability, the prediction performance should be sacrificed rather than improved. When uploading the QoE, the tradeoff between average QoE, resources, and average viewpoint leakage probability exists in most cases. To reduce the average viewpoint leakage probability, either the resources or the average QoE should be sacrificed.

References

[1] W.-C. Lo, C.-L. Fan, J. Lee, C.-Y. Huang, K.-T. Chen, and C.-H. Hsu, “360° video viewing dataset in head-mounted virtual reality,” ACM MMSys, 2017.

[2] M. F. Romero Rondon, L. Sassatelli, R. Aparicio-Pardo, and F. Precioso, “Track: A new method from a re-examination of deep architectures for head motion prediction in 360-degree videos,” IEEE Trans. Pattern Anal. Machine Intell., early access, 2021. [Online]. Available: https://gitlab.com/miguelfromeror/head-motion-prediction/tree/master/
[3] F. Qian, L. Ji, B. Han, and V. Gopalakrishnan, “Optimizing 360 video delivery over cellular networks,” ACM SIGCOMM Workshop, 2015.

[4] N. Kan, J. Zou, C. Li, W. Dai, and H. Xiong, “Rapt360: Reinforcement learning-based rate adaptation for 360-degree video streaming with adaptive prediction and tiling,” IEEE Trans. Circuits Syst. Video Technol., early access, 2021.

[5] A. Yaqoob, T. Bi, and G.-M. Muntean, “A survey on adaptive 360° video streaming: Solutions, challenges and opportunities,” IEEE Communications Surveys Tutorials, vol. 22, no. 4, pp. 2801–2838, 2020.

[6] G. Xiao, M. Wu, Q. Shi, Z. Zhou, and X. Chen, “DeepVR: Deep reinforcement learning for predictive panoramic video streaming,” IEEE Trans. Cogn. Commun. Netw., vol. 5, no. 4, pp. 1167–1177, 2019.

[7] S. Li, C. She, Y. Li, and B. Vucetic, “Constrained deep reinforcement learning for low-latency wireless VR video streaming,” IEEE GLOBECOM, 2021.

[8] M. R. Miller, F. Herrera, H. Jun, J. A. Landay, and J. N. Bailenson, “Personal identifiability of user tracking data during observation of 360-degree VR video,” Scientific Reports, vol. 10, no. 1, pp. 1–10, 2020.

[9] B. David-John, D. Hosfelt, K. Butler, and E. Jain, “A privacy-preserving approach to streaming eye-tracking data,” IEEE Trans. Vis. Comput. Graphics, vol. 27, no. 5, pp. 2555–2565, 2021.

[10] J. Li, A. R. Chowdhury, K. Gawaz, and Y. Kim, “Kaleido: Real-time privacy control for eye-tracking systems,” 30th USENIX Security Symposium (USENIX Security 21), Aug. 2021.

[11] E. Bozkir, A. B. Ünal, M. Akgün, E. Kasneci, and N. Pfeifer, “Privacy preserving gaze estimation using synthetic images via a randomized encoding based framework,” ACM Symposium on Eye Tracking Research and Applications, 2020.

[12] A. Liu, L. Xia, A. Duchowski, R. Bailey, K. Holmqvist, and E. Jain, “Differential privacy for eye-tracking data,” ACM Symposium on Eye Tracking Research and Applications, 2019.

[13] J. Steil, M. Koelle, W. Heuten, S. Boll, and A. Bulling, “Privaceye: Privacy-preserving head-mounted eye tracking using egocentric scene image and eye movement features,” ACM Symposium on Eye Tracking Research and Applications, 2019.

[14] C. Fan, S. Yen, C. Huang, and C. Hsu, “Optimizing fixation prediction using recurrent neural networks for 360° video streaming in head-mounted virtual reality,” IEEE Trans. Multimedia, vol. 22, no. 3, pp. 744–759, March 2020.

[15] C. Guo, L. Zhao, Y. Cui, Z. Liu, and D. W. K. Ng, “Power-efficient wireless streaming of multi-quality tiled 360 VR video in MIMO-OFDMA systems,” IEEE Trans. Wireless Commun., vol. 20, no. 8, pp. 5408–5422, 2021.

[16] 3GPP, “Extended reality (XR) in 5G,” 2020, 3GPP TR 26.928 version 16.0.0 release 16.

[17] X. Wei, C. Yang, and S. Han, “Prediction, communication, and computing duration optimization for VR video streaming,” IEEE Trans. Commun., vol. 69, no. 3, pp. 1947–1959, 2021.

[18] C. Li, W. Zhang, Y. Liu, and Y. Wang, “Very long term field of view prediction for 360-degree video streaming,” IEEE MIPR, 2019.

[19] iLab, “Cloud VR network solution whitepaper,” Huawei Technologies CO., LTD., Tech. Rep., 2018. [Online]. Available: https://www.huawei.com/minisite/pdf/ilab/cloud_vr_network_solution_white_paper_en.pdf

[20] M. F. R. Rondón, L. Sassatelli, R. Aparicio-Pardo, and F. Precioso, “A unified evaluation framework for head motion prediction methods in 360° videos,” ACM MMSys, 2020.

[21] J. Zou, C. Li, C. Liu, Q. Yang, H. Xiong, and E. Steinbach, “Probabilistic tile visibility-based server-side rate adaptation for adaptive 360-degree video streaming,” IEEE J. Sel. Topics Signal Process., vol. 14, no. 1, pp. 161–176, 2020.

[22] C.-L. Fan, W.-C. Lo, Y.-T. Pai, and C.-H. Hsu, “A survey on 360° video streaming: Acquisition, transmission, and display,” ACM Comput. Surv., vol. 52, no. 4, Aug. 2019.

[23] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-efficient learning of deep networks from decentralized data,” PMLR AISTATS, 2017.

[24] J. Dupuy, E. Heitz, and L. Belcour, “A spherical cap preserving parameterization for spherical distributions,” ACM Trans. Graph., vol. 36, no. 4, July 2017.

[25] A. Tovchigrechko and I. A. Vakser, “How common is the funnel-like energy landscape in protein-protein interactions?” Protein science, vol. 10, no. 8, pp. 1572–1583, 2001.
A PPENDIX A

PROOF OF COROLLARY 1

When \( \Pr_e^{\min} = 1 \), \( \Pr_e \leq \Pr_e^{\min} \) can always be satisfied, hence \( e \in [0, \pi] \). When \( \Pr_e^{\min} \in [0, 1) \), by substituting \( \Pr_e \leq \Pr_e^{\min} \) into (7), we have \( \sin(e) \geq \frac{e}{\Pr_e^{\min} \pi} \). To ensure the feasible solution, \( \Pr_e^{\min} \leq 1 \), from which we obtain \( \Pr_e^{\min} \geq \Pr_e \). In this case, from \( \sin(e) \geq \frac{e}{\Pr_e^{\min} \pi} \), we obtain the range of prediction error as \( e \in [e_{\min}^{\min}, e_{\max}^{\min}] \).

When \( \Pr_e^{\min} \in [\Pr_e^{\min}, 1) \), by substituting \( \Pr_e > \Pr_e^{\min} \) into (7), we have \( \sin(e) < \frac{e}{\Pr_e^{\min} \pi} \), from which we obtain \( e \in [0, e_{\min}^{\min}) \cup (e_{\max}^{\min}, \pi] \).

A PPENDIX B

PROOF OF PROPOSITION 1

When the area of SFoV is zero, \( r_{sv} = 0 \), the overlapped area of FoV and SFoV is \( A_{ol} = 0 \), QoE = 0. When the SFoV is the whole sphere, \( r_{sv} = \pi \), \( A_{ol} = A_{fov} \). According to (5), QoE = 100\%. In the sequel, we consider the cases where \( r_{sv} \in (0, \pi) \).

A. FoV \( \subset \) SFoV: As shown in Fig. 4a, \( A_{ol} = A_{fov} \). Hence, QoE = 100\%. To ensure FoV being included in SFoV, the radius of SFoV should be no less than the radius of FoV plus the spherical distance between \( O_v \) and \( O_p \), i.e., \( r_{sv} \geq r_{fov} + e \).

B. SFoV \( \subset \) FoV: As shown in Fig. 4b, \( A_{ol} = A_{sv} \). Hence, QoE = \( \frac{A_{sv}}{A_{fov}} = \frac{1 - \cos(r_{sv})}{1 - \cos(r_{fov})} \). Similarly to the above analysis, To ensure that SFoV being included in FoV, \( r_{fov} \geq r_{sv} + e \).

C. FoV \( \cap \) SFoV = \( \emptyset \): As shown in Fig. 4c, \( A_{ol} = 0 \), QoE = 0. Similarly to the above analysis, the condition is \( e \geq r_{fov} + r_{sv} \).

D. SFoV-C \( \subset \) FoV, i.e., the complement of SFoV is included in FoV. This is the case that has been overlooked in previous work of overlapped area of two spherical caps [24]. As shown in Fig. 4d, the overlapped area of FoV and SFoV can be expressed by the area of FoV minus the complement of SFoV (abbreviated as SFoV-C in the following), i.e., \( A_{ol} = A_{fov} - (4\pi - A_{sv}) \).

Hence, QoE = \( \frac{A_{fov} + A_{sv} - 4\pi}{A_{fov}} = \frac{-\cos(r_{sv}) - \cos(r_{fov})}{1 - \cos(r_{fov})} \). Denote the center of SFoV-C as \( O_c \), the spherical
distance from \( O_c \) to the viewpoint \( O_v \) can be expressed as \( e^c = \pi - e \). The radius of SFoV-C can be expressed as \( r^c_{sv} = \pi - r_{sv} \). Then, the radius of FoV should be no less than the radius of SFoV-C plus the spherical distance between their centers, i.e., \( r_{fov} \geq r^c_{sv} + e^c \). By substituting \( r^c_{sv} \) and \( e^c \) into \( r_{fov} \), we obtain \( r_{fov} + r_{sv} + e \geq 2\pi \).

E. Remaining case: As shown in Fig. 2c, FoV intersects with SFoV, FoV and SFoV cannot contain each other, and the FoV cannot contain the SFoV-C. The overlapped area of FoV and SFoV can be obtained from [24], [25] as in (12). Hence,

\[
QoE = \frac{A_{ol}(r_{sv}, e)}{2\pi(1 - \cos(r_{fov}))} = \frac{A_{ol}}{A_{fov}} > 0.
\]

The remaining case holds when all of the conditions in the four cases are violated. After some regular derivations, the condition of this case can be expressed as \( r_{sv} \in [r_{sv,\min}(e), r_{sv,\max}(e)] \).

This completes the proof.

APPENDIX C

PROOF OF PROPOSITION 2

Proof of (13a): By substituting \( QoE = 100\% \) into (5), we have \( A_{ol} = A_{fov} \). That is to say, the overlapped area of FoV and SFoV is the area of FoV. This indicates that \( FoV \subset SFoV \). Then from (11) we have \( e \leq r_{sv} - r_{fov} \). Since \( e \geq 0 \), we obtain \( e \in [0, r_{sv} - r_{fov}] \).

Proof of (13b): \( QoE = \frac{1 - \cos(r_{sv})}{1 - \cos(r_{fov})} = \frac{2\pi(1 - \cos(r_{sv}))}{2\pi(1 - \cos(r_{fov}))} = \frac{A_{ol}}{A_{fov}} > 0 \). Besides, from (5) we have \( QoE = \frac{A_{ol}}{A_{fov}} \). We can find that \( A_{ol} = A_{sv} \). This indicates that \( SFoV \subset FoV \). Then, from (11) we have \( e \leq r_{fov} - r_{sv} \). Since \( e \geq 0 \), we obtain \( e \in [0, r_{fov} - r_{sv}] \).

Proof of (13c): By substituting \( QoE = 0 \) into (5), we have \( A_{ol} = 0 \), i.e., \( FoV \cap SFoV = \emptyset \). Then from (11) we have \( e \geq r_{fov} + r_{sv} \). Since \( e \leq \pi \), we obtain \( e \in [r_{fov} + r_{sv}, \pi] \).

Proof of (13d): \( QoE = \frac{-2\pi(1 - \cos(r_{sv}))}{2\pi(1 - \cos(r_{fov})) - 4\pi} = \frac{A_{ol}}{A_{fov}} - \frac{4\pi - A_{sv}}{A_{fov}} \). With \( QoE = \frac{A_{ol}}{A_{fov}} \) we have \( A_{ol} = A_{fov} - (4\pi - A_{sv}) \). We can observe that the overlapped area of FoV and SFoV is the difference of the areas of FoV and SFoV-C. That is to say, \( SFoV-C \subset FoV \). Then, from (11) we have \( e \geq 2\pi - r_{fov} - r_{sv} \). Since \( e \leq \pi \), we obtain \( e \in [2\pi - r_{fov} - r_{sv}, \pi] \).

Proof of (13e): When \( r_{sv} \in (0, \pi) \), by comparing (11c)-(11g) and (13a)-(13d), we can find that
the case that can satisfy $QoE \neq 100\%$, $QoE \neq \frac{1-\cos(r_{sw})}{1-\cos(r_{fov})}$, $QoE \neq 0$, and $QoE \neq \frac{-\cos(r_{sw})-\cos(r_{fov})}{1-\cos(r_{fov})}$ is the remaining case. Then, the value of $e$ can be determined by bisection search.

This completes the proof.