2023 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB) | 979-8-3503-2152-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/BMSB58369.2023.10211169

User Profile-Based Viewport Prediction Using Federated Learning in Real-Time 360-Degree Video Streaming

Syed Mohammad Haseeb ul Hassan
School of Electronic Engineering
Dublin City University
Dublin, Ireland
syed.haseebulhassan2@mail.dcu.ie

Attracta Brennan
School of Computer Science
University of Galway
Galway, Ireland
attracta.brennan@nuigalway.ie

Gabriel-Miro Muntean, Jennifer McManis
School of Electronic Engineering
Dublin City University
Dublin, Ireland
{gabriel.muntean,jennifer.mcmanis}@dcu.ie

***Abstract*—Streaming 360-degree videos has become increasingly popular due to the growth in demand for immersive media in recent years. Companies such as Youtube, Facebook, and Netflix already use 360-degree video streaming. In order to reduce the amount of data transmitted only the part of the video at which the user looks is streamed at high resolution and enabling this requires accurate viewport prediction. However, recent approaches to streaming 360-degree video do not characterize user profiles or have low viewport prediction accuracy when either historical data is unavailable for the user or when the user starts watching a new video. This paper proposes a novel approach to User Profile-Based Viewport Prediction Using Federated Learning (UVPFL) in 360-degree Real-Time Video Streaming. UVPFL profiles users based on their head movements for different categories of videos. For high viewport prediction accuracy of a new user or a user with no historical data, UVPFL bases its viewport prediction on the viewport of similar users. Testing UVPFL in 360-degree real-time video streaming has resulted in an accuracy of up to 86% for the first seven seconds of video play. UVPFL also achieved an average accuracy of up to 96% for the complete length of video play. UVPFL has outperformed three state-of-the-art available viewport prediction solutions by 1.12% to 64.9% for a 1 second prediction horizon.**

***Index Terms*—Virtual Reality, Viewport Prediction, Multimedia Streaming, Federated Learning, 360-degree Video Streaming**

I. Introduction

The popularity of 360-degree videos has continued to increase, as signaled by social media trends such as the transformation of Facebook to Meta and the creation of the Metaverse. A Head Mounted Display (HMD) or a high-specification mobile device is primarily used to watch these videos [1] and use of devices continues to trend upwards. For instance, the number of Virtual Reality (VR)-based apps downloaded on Oculus Quest has increased by 153% from December 2020 to December 2021 [2]. According to Cisco [3], the minimum bandwidth required for High Definition (HD) VR streaming and Ultra High Definition (UHD) VR streaming is 150 Mbps and 500 Mbps, respectively. Conventional video streaming solutions used by social media platforms such as Youtube and Facebook deliver the entire frame to the user simultaneously, with a minimum bandwidth requirement of 20 Mbps [4].

The high bandwidth requirements for 360-degree video streaming have motivated interest in techniques to reduce the amount of sent information, particularly by employing viewport-based adaptive streaming approaches. The idea behind viewport-based adaptive streaming is that for a 360-degree video, the user watches a portion of the video (i.e., the viewport) displayed on their HMD. While the 360-degree video offers a panoramic 360-degree horizontal view and a 180-degree vertical view, a viewport is typically less than 20% of the 360-degree video and allows the user to view about 90-degree to 120-degree horizontally and 90-degree vertically [5]. Muntean *et al*. [6] and Ciubotaru *et al*. [7] proposed a region of interest-based adaptive multimedia streaming scheme for 2D videos and achieved up to 28% gain in peak signal-to-noise ratio as compared to competitors. Similar solutions that stream the viewport area at high resolution whilst streaming other parts at low resolution may significantly decrease the bandwidth requirement and improve the Quality of Experience (QoE) for the user [8]. In order to successfully maintain high QoE levels for the user, the viewport prediction algorithm must be able to quickly and accurately predict the user's viewport. Viewport prediction during live video streaming includes analyzing the user's eyes and/or head movements in real-time using computer vision techniques and Machine Learning (ML) algorithms to predict what parts of the video users are most likely to look at next. Feng *et al*. [9] predicted the viewport in live video streaming using a simple Convolutional Neural Network (CNN) approach and achieved an accuracy of 70-80% depending on the complexity of the video. Chen *et al*. [10] used a white box explainable approach for viewport

This publication has emanated from research conducted with the financial support of the Science Foundation Ireland under Grant number 18/CRT/6224 (SFI Centre for Research Training in Digitally-Enhanced Reality D-REAL).

979-8-3503-2152-4/23/$31.00 ©2023 IEEE

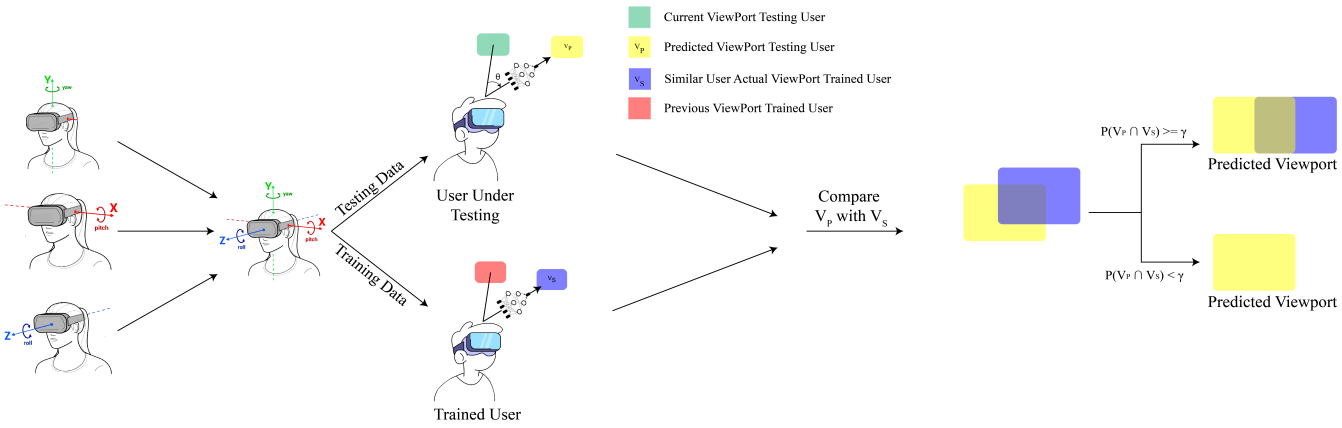

Fig. 1. UVPFL Viewport Prediction

prediction during live streaming and achieved an accuracy of 94% for a 1 second horizon.

A number of works have attempted to improve on the accuracy and bandwidth reduction in the context of viewport prediction. Feng *et al*. [11] predicted the viewport for lifelong video streaming using the CNN model visual geometry group and achieved an accuracy between 50% and 90% depending on the video type. In non-live video streaming, it is very important to anticipate which parts of a pre-recorded video a user will most likely watch based on watching habits and history. This is done to enhance the video quality and decrease the lag when playing back the video. Park *et al*. [5] introduced CNN + Long Term Short Term Memory (LSTM) and 3DCNN for viewport prediction they also used the saliency heat map of the video as an input. They achieved an average accuracy of 92.21% for a 1 second horizon. The saliency of a pixel/part of a picture defines how likely the pixel can get the attraction of the viewer. Yaqoob *et al*. [12] proposed a combined Field-of-View (FoV) tile-based adaptive streaming solution; they measured accuracy for different tiling patterns (i.e., 4×3, 6×4, and 8×6) and for different segments lengths (i.e., 1 second, 2 second, and 3 second) They achieved an accuracy of 58% to 78%. He *et al*. [13] worked on FoV prediction for smartphone streaming and achieved an average throughput of 0.15Mbps to 2.1Mbps for 4k videos. Wu *et al*. [14] used spherical CNN for viewport prediction for stored streaming and achieved an accuracy of 40% for 10 seconds. Szabó *et al*. [15] used mulsemedia (olfactory and haptic sensors) using CNN to improve the QoE of the end user in 360-degree videos.

A. Related Works

Most current viewport prediction approaches do not address live 360-degree video streaming for users with no historical data [5] [16] [10]. The approach of Feng *et al*. [9] using the simple CNN model AlexNet resulted in low viewport prediction accuracy during the first 20 seconds of the video play due to a lack of user data. It is also reported that Facebook downloads 5 seconds of playout time [4]. However, downloading data for playout also has limitations [17]; 1) the user needs good network conditions to download data and, 2) the user needs to wait until the data is downloaded; this results in a low QoE level.

Understandably, achieving accurate early (i.e., the first few seconds of video play) predictions of a user's head movement is challenging. However, Stefano *et al*. [18] achieved an accuracy of up to 70%, 50%, and 13% for the time horizons of 2 second, 5 second, and 10 second, respectively by tracking the user head movement trajectories using their trajectory-based viewport prediction approach. Flare [16] used Linear Regression, Ridge Regression, and Supporting Vector Machine for viewport prediction and adaptation, resulting in an accuracy of up to 90.5%, 58.2%, and 46.7% for the 0.2 second, 1 second and 2 second horizon respectively. Mosaic [5] used LSTM+CNN and 3DCNN for viewport prediction and adaptation and achieved an accuracy of up to 92.21% and 88.44% for 1 second and 2 second horizons, respectively. Sparkle [10] used a white box explainable approach for viewport prediction during live streaming and achieved an accuracy of 94% and 92.21% for 1 second and 2 second horizons, respectively.

Despite such high accuracies recorded by these studies, the methods used cannot accurately predict viewport for the first few seconds of a video play when no historical user data is available. This limitation can be potentially mitigated through the use of user profiling. In this context, user profiling refers to the consideration of user characteristics such as interests, age, gender, or region and how these characteristics build user behaviour for a particular type of content [19] [20]. Currently, available solutions lack user profiling for viewport prediction in 360-degree video delivery.

Edge computing and distributive approaches can be used for viewport prediction as live or stored video streaming solutions. Edge computing solutions can serve to reduce decision time by reducing the dependency on a centralized server and bringing the decision-making process closer to the Edge node. This approach is only applicable when many users are watching the

same content. However, in Edge computing, one node works as a primary node and other nodes work as secondary nodes. Secondary nodes always depend upon primary nodes for data processing and data storage, potentially leading to bottleneck and processing delays. Unlike Edge computing, our approach involves the processing and storing of data individually at each node. Hence, we have proposed Federated Learning (FL) solution which is a decentralized approach for the storage and processing of the data individually at each node. Liyang *et al*. [21] used caching on the Edge server for the FoV prediction. Here a live video is streamed to multiple users watching the same video on a local network. These users are categorized or flocked into groups based on their latency tolerance and how much their FoV overlapped. This approach reduces the traffic on the core network by 80% but increases the load on Edge nodes. Uddin *et al*. [22] used caching at the Edge server to decrease latency and bandwidth loss in 360-degree videos. These approaches help to reduce the core network's data transmission by performing tasks at the edge nodes.

What is Federated Learning and why it is required? FL is a ML-based approach, where the data is stored on the client side for training. The benefit of FL is that each user can be trained separately and can be profiled independently. FL is widely used in data protection, privacy, edge computing, and wireless networks [23]. Chao *et al*. [24] used FL for privacy preservation in 360-degree video streaming and identified users based on historical user data with an accuracy of up to 25.23%. Chen *et al*. [25] used FL for user profiling in the aviation industry to identify high-value passengers and achieved the Area Under Curve value of 0.85 and Kolmogorov–Smirnov value of 0.55, showing improvements of 9% and 31%, respectively, on currently available solutions. Wang *et al*. [26] used FL for user characteristics identification in order to learn the electricity consumers' electricity consumption pattern; the approach achieved an accuracy of 68.5% for character identification which was a 1.69% improvement on conventional approaches. Zhou *et al*. [27] used FL for user profiling for wearable sensors and focused on network adaptation, data consumption, and network traffic. This approach has decreased the amount of upload data by 29.77% and improved model accuracy by 20%. In our approach, we used FL for profiling users separately on the basis of their head movement in 360-degree videos.

B. Our Contribution

In this paper, we use FL to predict the user's viewport by profiling their head movement behaviour for different types of videos. We also propose a novel algorithm UVPFL that bases the prediction on both the observed user head movement and the historical viewport data. After viewport prediction, we compare our viewport to a similar user viewport or a historical viewport. We merge and update the predicted viewport if the overlapping area is higher than γ. Using this approach, we have achieved an average accuracy of up to 96% for the overall video. During the first 7 seconds of video play, we achieved an average accuracy of up to 86%.

II. Proposed Approach

A. Methodology

The motivation behind UVPFL is that similar users tend to have similar viewing behaviour while watching 360-degree videos. UVPFL uses FL to observe each user's head movement behaviour separately for different types of 360-degree videos. The data relating to the user's head movement behaviour is stored on the client side. Our ML algorithm is performed separately on each client. To test UVPFL, we used a publicly available dataset [28] containing the head movement of 50 users and 10 videos. We modified the dataset by separating it for each user and video category. This modification helped us test UVPFL approach and get high accuracy for each video and video category separately.

We have two categories of users: trained users and users under testing. A trained user is a user who has already watched a video using UVPFL. A test user is a person who is currently watching the video. Data from users similar to the test users may be used as historical data to improve the test user's viewport prediction. As shown in Fig. 1, Yaw (Y), Pitch (PI), and Roll (R) angles are used to calculate the user's current and predicted viewport. We used a pre-trained ML model to predict the viewport for test users; θ is the expected change of angles (Y, PI, and R). Our key contribution involves the comparison between the current user's predicted viewport (V_P) and that of a similar user (V_S) to initially extract prediction errors but ultimately to achieve higher accuracy during the whole video. This is shown in Fig. 1. If the overlapping area of V_P and V_S is greater than γ, then UVPFL merges V_P and V_S. For example, if a user is watching a video and the overlapping area of V_P and V_S is higher than γ, then UVPFL merges the two viewports.

B. UVPFL Algorithm

Our UVPFL Algorithm 1 is divided into three phases. In Phase 1, Video processing all of the processes are performed on the server side. With the 360-degree video and saliency heat map provided as input, we divide the video into segments and frames. Each frame is further divided into an lxk grid of tiles.

In Phase 2: Viewport Prediction, all of the processes are performed on the client side. Here the Tiles T_{lk} of the current frame and saliency data of the current and similar user are provided as input. We get the Y, PI, and R data of the user and calculate the head movement speed of the current user. This information is then sent as input to our pre-trained ML model. Next, we predict the viewport V_P using the pre-trained ML algorithm. After a viewport is predicted, our algorithm compares V_P with a similar user viewport V_S. If the overlapping is greater than γ, then we merge both viewports and update our predicted viewport. Using the equation provided by Zou *et al*. [29], as shown in Eq.(1), we calculate the visibility probability of tiles in V_P and neighbouring V_P.

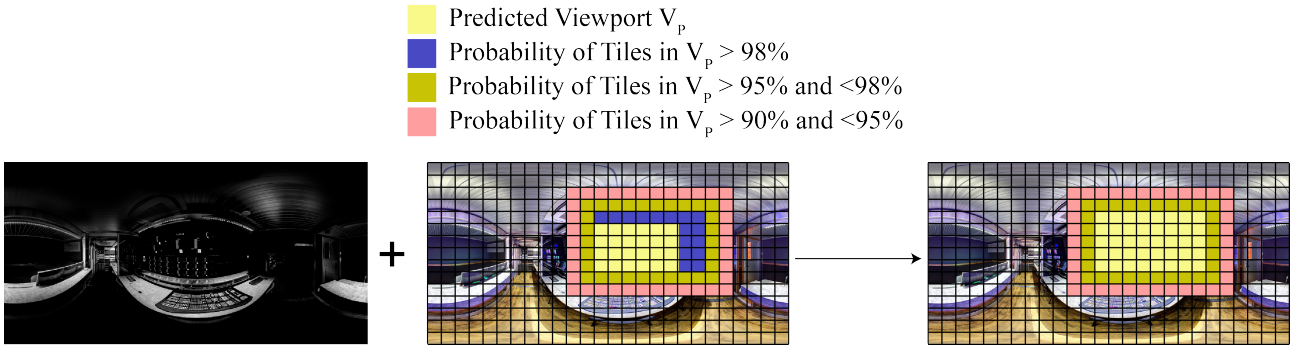

Fig. 2. Probability Of Tiles

As shown in Fig. 2, we create different groups of tiles based on their probabilities of being in the viewport. Group one tiles represent V_P; Group two tiles have a probability higher than 98%; Group three tiles have a probability between 95% and 98%; Group four tiles have a probability between 90% and 95%. We next merge tiles having a probability higher than 98% with viewport tiles. After this, we select T_{lk} related to V_P and identify tiles T^N neighbouring to Viewport. After calculating how much time the user spent on the previous T_{lk}^{j-1}, we update the saliency heat map of the user.

$$P_{lk} \triangleq P_{lk}^{\theta} \cdot P_{lk}^{\varphi} = \int_{\max\left\{\theta_{lk}^{lower}-\theta,-90^{\circ}\right\}}^{\min\left\{\theta_{lk}^{upper}-\theta,90^{\circ}\right\}} p\theta(\Delta\theta)d\Delta\theta \cdot \int_{[\varphi_{lk}^{left}-\varphi]l_0}^{[\varphi_{lk}^{right}-\varphi]l_0} p\varphi(d\varphi)d\Delta\varphi \quad (1)$$

In Phase 3: Tile adaptation is performed. We used a simple hierarchical tile adaptation approach for improved QoE. The tiles in the coordinate set V_P are assigned the highest quality, the tiles in the coordinate set neighboring viewport V_N are assigned medium quality, and the tiles in the coordinate set V_Z are assigned a low quality.

III. Experiments

A. Training Setup

For training the model, we used Resnet50 Gated Recurrent Unit (GRU), where first we chose pre-trained Resnet50, then added time distributed layers, and then we added GRU as shown in Fig. 3. The final output of Fig. 3 depicts the predicted viewport, which has been used in Algorithm 1. We used different models and parameters such as AlexNet, MobileNet, and Resnet50 for 100 epochs. We observed that we get the best results on parameters such as the Adam optimizer, mean_squared_error loss, and 50 epochs. After 50 epochs, there is much less change in accuracy. The accuracy achieved by Alexnet, MobileNet, and Resnet50 is 73%, 56%, and 96%, respectively. The input shape of the model was (30,120,240,3), where 30, 120, 240, and 3 are frames, height, width, and channels, respectively. We used different frames, i.e., 30, 60, and 90, for further investigation purposes.

B. Dataset

As already mentioned, we used a publicly available dataset [28] containing the head movement data of 50 users and

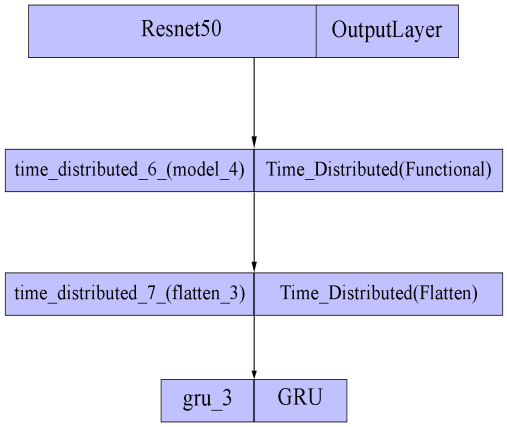

Fig. 3. Architecture Diagram for UVPFL

10 videos, with the videos categorized as (1) Computer-Generated (CG) fast-paced, (2) Natural-Image (NI), and fast-paced. and (3) NI, slow-paced. The length of each video was approximately 60 seconds. In user profiling, we noted the user head movement behaviour for each video category separately.

C. Results

We measured the accuracy for different values of the overlapping area γ such as 70%, 80%, and 90% for a 1 second horizon. We achieved the best accuracy in the least computational time of 648 ms at 80%. Overlapping areas less than 80% brought little change in accuracy but required more computational time, i.e., time for viewport prediction. Fig. 4 shows the comparison of the different γ values for the 1 second horizon. Based on these results, we performed all our experiments with an overlapping area of γ = 80%.

We calculated the average accuracy of the video for each user with respect to time for video category and we achieved an average accuracy of 86% within the first 7 seconds of the

Algorithm 1: UVPFL algorithm

Goal: Viewport prediction
Input: 360 degree video;
Input: Saliency feature data;
PHASE 1:
Server Side Process
Divide video into N segments of time S;
foreach *Segment i* **do**
foreach *Frame j* **do**
Get saliency heat map of each $Frame_j$;
Get T_{lk}^j from $Frame_j$ ▷ Client Side Process
$T_{lk} \in$ T $\leftarrow T_{lk}$
l={1,2,3,.....L}, k={1,2,3,.....K}
Phase 2: Viewport Prediction
Client Side Process
Input Saliency heat map data of $Frame_j$ of $User$;
Input: $\mathbb{T}_{lk}$ in $Frame_j$
Calculate Y, R, PI for User;
Calculate head movement speed of $User$;
Predict viewport V_P;
if $V_P \cap V_S \geq \gamma$ % **then**
Merge V_P and V_S;
Else
V_P remain same ;
end

Update V_P;
Calculate $\mathbb{P}T_{lk}$ in V_P $Frame_j$ [29]
Select T_{lk} related to V_P;
Identify T^N to V_P;
Calculate how much time user spent T_{lk}^{j-1}
Update user saliency heat map $Frame_{j-1}$;

Phase 3: Tile Adaptation
Input: Coordinates of V_P, V_N, V_Z (i.e., the coordinate sets of the viewport, neighboring to the viewport, and other areas, respectively);
Input: $\mathbb{T}_{lk}$ in $Frame_j$
Assign low quality to all V_Z;
Assign medium quality to all V_N;
Assign high quality to all V_P;
end
end

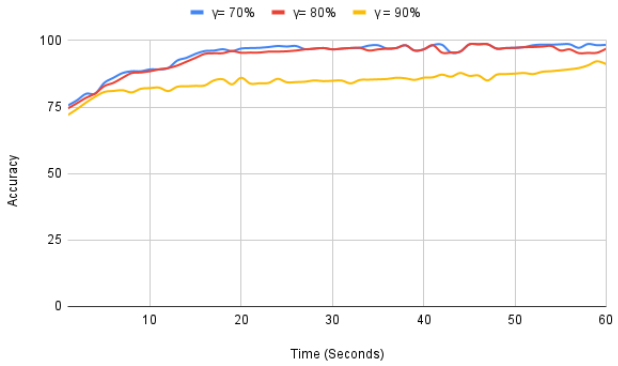

Fig. 4. Accuracy Of Video For The 1 Second Horizon For Different Overlapping Values Of γ Is 70% 80% And 90%

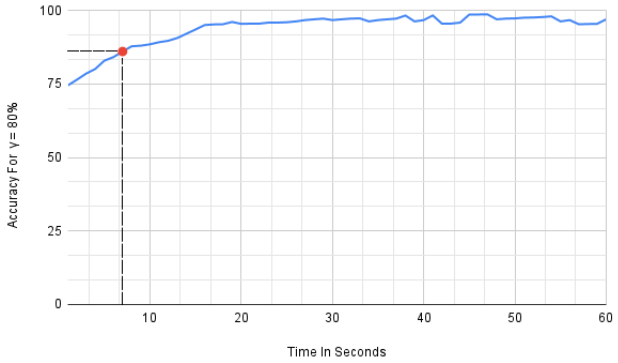

Fig. 5. Accuracy For First 7 Seconds, γ 80% For 1 Second Horizon

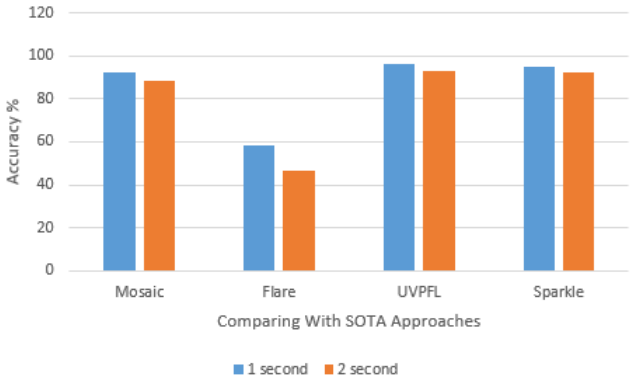

Fig. 6. Accuracy Comparison with SOTA Approaches for 1 Second and 2 Second Horizons

video for the 1 second horizon and γ 80% (Fig. 5). We tested UVPFL three times to validate and took the average accuracy.

We compared the proposed UVPFL approach with three other state-of-the-art (SOTA) available approaches (1) Mosaic [5], (2) Flare [16], and (3) Sparkle [10]. As shown in Fig. 6, UVPFL outperformed all available approaches for a 1 second and 2 second horizon. For a 1 second horizon, UVPFL has shown 4.1% better results than Mosaic, 64.9% better results

than Flare, and 1.12% better results than Sparkle. For a 2 second horizon, UVPFL also showed 5.32% better results than Mosaic, 99% better results than Flare, and 0.96% better results than Sparkle. We tested UVPFL three times for the validation process. As shown in Table I, we achieved an average accuracy of 96%, 93.15%, and 89% across all our videos for the 1 second, 2 second, and 4 second horizon, respectively. Additionally, we calculated the precision and recall values for UVPFL. Precision is the ratio of the number of tiles correctly predicted to be viewed to the number of tiles to be viewed both correctly and incorrectly [5]. The higher the value of precision, the less the prediction error. Recall computes the difference between the predicted and the actual number of tiles viewed. A high value of recall means that fewer tiles are predicted incorrectly. We observed that while the precision and recall values were very high for the 1 second horizon, they decreased when we increased the horizon time to 2 seconds and 4 seconds.

TABLE I
AVERAGE PREDICTION ACCURACY FOR DIFFERENT HORIZONS IN SECONDS WITH γ VALUE 80%

Frames	*Accuracy*	*Precision*	*Recall*
1 second	96	92.41	90.15
2 second	93.15	89.21	85.23
4 second	89	75.60	73.75

Furthermore, we calculated the accuracy for each video and category separately, as shown in Fig. 7. We repeated the test three times for validation for each video and related category. We achieved an average accuracy of 96.6%, 95.25%, and 96.33% for the NI fast-paced, NI slow-paced, and CG fast-paced categories, respectively. UVPFL performs best in fast-paced videos and provides us with the highest accuracy compared to slow-paced videos. We observed that UVPFL performed well where user head movements were more frequent. We achieved the best results in the video Roller Coaster from category NI fast-paced and Pacman from CG fast-paced with an average accuracy of up to 98%. We achieved the lowest average accuracy in the SFR Sports from category NI slow-paced with an average accuracy of 94%. Notwithstanding, this accuracy is higher than the accuracy of SOTA approaches.

IV. CONCLUSION

In this work, we proposed a novel approach to **U**ser **P**rofile-**B**ased **V**iewport **P**rediction **U**sing **F**ederated **L**earning (UVPFL) in 360-degree real-time video streaming. UVPFL addresses viewport prediction accuracy issues when historical data is unavailable for the user or when the user starts watching a new video. UVPFL profiles the user and performs viewport prediction with high accuracy for new users or users with no historical data. First, UVPFL profiles the user by head movement data for different video categories. Next, it compares the predicted viewport with a similar user viewport. If the overlapping area is higher than 80%, it merges the viewports.

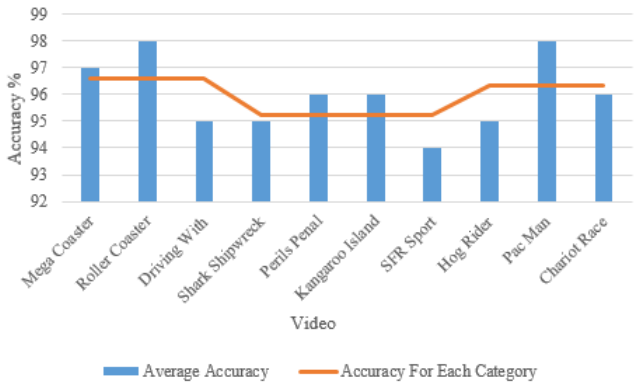

Fig. 7. Accuracy for each video

To check the effectiveness of UVPFL, we performed several experiments using a publicly available dataset [28] containing the head movement data of 50 users and 10 videos. UVPFL achieved an average accuracy of 96% for viewport prediction. UVPFL outperformed the existing approaches Mosaic, Flare, and Sparkle by 4.1%, 6.07%, and 1.12%, respectively. However, we believe that there are still opportunities for improvement in UVPFL. Due to data limitations, we only profile users' head movements in different video categories. In the future, we plan to profile users based on eye movement data, regional impact on user response, and user behaviour to interact with different objects to improve performance.

REFERENCES

[1] S. Hollister, "Youtube's ready to blow your mind with 360-degree videos," Mar 2015. [Online]. Available: https://gizmodo.com/youtubes-ready-to-blow-your-mind-with-360-degree-videos-1690989402

[2] "Global Oculus mobile app new installs 2021 — Statista — statista.com," https://www.statista.com/statistics/1283725/oculus-mobile-app-global-new-installs-christmas/, [Accessed 24-Jan-2023].

[3] "Cisco annual internet report - cisco annual internet report (2018–2023) white paper," Jan 2022. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[4] C. Zhou, Z. Li, and Y. Liu, "A measurement study of oculus 360 degree video streaming," in *Proc. of the 8th ACM on Multimedia Systems Conference*, ser. MMSys'17. New York, USA: ACM, 2017, p. 27–37. [Online]. Available: https://doi.org/10.1145/3083187.3083190

[5] S. Park, A. Bhattacharya, Z. Yang, S. R. Das, and D. Samaras, "Mosaic: Advancing user quality of experience in 360-degree video streaming with machine learning," *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 1000–1015, 2021.

[6] G.-M. Muntean, G. Ghinea, and T. N. Sheehan, "Region of interest-based adaptive multimedia streaming scheme," *IEEE Transactions on Broadcasting*, vol. 54, no. 2, pp. 296–303, 2008.

[7] B. Ciubotaru, G.-M. Muntean, and G. Ghinea, "Objective assessment of region of interest-aware adaptive multimedia streaming quality," *IEEE Transactions on Broadcasting*, vol. 55, no. 2, pp. 202–212, 2009.

[8] 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.

[9] X. Feng, Z. Bao, and S. Wei, "Exploring cnn-based viewport prediction for live virtual reality streaming," in *2019 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, 2019, pp. 183–1833.

[10] J. Chen, X. Luo, M. Hu, D. Wu, and Y. Zhou, "Sparkle: User-aware viewport prediction in 360-degree video streaming," *IEEE Transactions on Multimedia*, vol. 23, pp. 3853–3866, 2021.

[11] X. Feng, Y. Liu, and S. Wei, "Livedeep: Online viewport prediction for live virtual reality streaming using lifelong deep learning," in *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, 2020, pp. 800–808.

[12] A. Yaqoob and G.-M. Muntean, "A combined field-of-view prediction-assisted viewport adaptive delivery scheme for 360° videos," *IEEE Transactions on Broadcasting*, vol. 67, no. 3, pp. 746–760, 2021.

[13] J. He, M. A. Qureshi, L. Qiu, J. Li, F. Li, and L. Han, "Rubiks: Practical 360-degree streaming for smartphones," in *Proc. of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '18. New York, USA: ACM, 2018, p. 482–494. [Online]. Available: https://doi.org/10.1145/3210240.3210323

[14] C. Wu, R. Zhang, Z. Wang, and L. Sun, "A spherical convolution approach for learning long term viewport prediction in 360 immersive video," *Proceedings of the AAAI Conference on Artificial Intelligence*. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/7377

[15] P. Szabo, A. Simiscuka, S. Masneri, M. Zorrilla, and G.-M. Muntean, "A cnn-based framework for enhancing 360 vr experiences with multi-sensorial effects," *IEEE Transactions on Multimedia*, pp. 1–1, 2022.

[16] F. Qian, B. Han, Q. Xiao, and V. Gopalakrishnan, "Flare: Practical viewport-adaptive 360-degree video streaming for mobile devices," in *Proc. of the 24th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '18. New York, USA: ACM, 2018, p. 99–114. [Online]. Available: https://doi.org/10.1145/3241539.3241565

[17] X. Che, B. Ip, and L. Lin, "A survey of current youtube video characteristics," *IEEE MultiMedia*, vol. 22, no. 2, pp. 56–63, 2015.

[18] S. Petrangeli, G. Simon, and V. Swaminathan, "Trajectory-based viewport prediction for 360-degree virtual reality videos," in *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, 2018, pp. 157–160.

[19] A. Dirin, A. Alamäki, and J. Suomala, "Gender differences in perceptions of conventional video, virtual reality and augmented reality," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 13, 06 2019.

[20] F. Needle, "Youtube demographics amp; data to know in 2023 [+ generational patterns]," Aug 2022. [Online]. Available: https://blog.hubspot.com/marketing/youtube-demographics

[21] L. Sun, Y. Mao, T. Zong, Y. Liu, and Y. Wang, "Flocking-based live streaming of 360-degree video," in *Proc. of the 11th ACM Multimedia Systems Conference*, ser. MMSys '20. New York, USA: ACM, 2020, p. 26–37. [Online]. Available: https://doi.org/10.1145/3339825.3391856

[22] M. Milon Uddin and J. Park, "360 degree video caching with lru lfu," in *2021 IEEE 12th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*, 2021, pp. 0045–0050.

[23] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020.

[24] F.-Y. Chao, C. Ozcinar, and A. Smolic, "Privacy-preserving viewport prediction using federated learning for 360° live video streaming," in *2022 IEEE 24th International Workshop on Multimedia Signal Processing (MMSP)*, 2022, pp. 1–6.

[25] S. Chen, D.-L. Xu, and W. Jiang, "High value passenger identification research based on federated learning," in *2020 12th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 1, 2020, pp. 107–110.

[26] Y. Wang, I. L. Bennani, X. Liu, M. Sun, and Y. Zhou, "Electricity consumer characteristics identification: A federated learning approach," *IEEE Transactions on Smart Grid*, vol. 12, no. 4, pp. 3637–3647, 2021.

[27] P. Zhou, H. Xu, L. H. Lee, P. Fang, and P. Hui, "Are you left out? an efficient and fair federated learning for personalized profiles on wearable devices of inferior networking conditions," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 6, no. 2, jul 2022. [Online]. Available: https://doi.org/10.1145/3534585

[28] 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," in *Proc. of the 8th ACM on Multimedia Systems Conference*, ser. MMSys'17. New York, USA: ACM, 2017, p. 211–216. [Online]. Available: https://doi.org/10.1145/3083187.3083219

[29] 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 Journal of Selected Topics in Signal Processing*, vol. 14, no. 1, pp. 161–176, 2020.