IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 31 July 2023, accepted 24 August 2023, date of publication 31 August 2023, date of current version 6 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3310822

RESEARCH ARTICLE

Deep Learning Approach for Seamless Navigation in Multi-View Streaming Applications

TIAGO S. COSTA[1], PAULA VIANA[1,2], (Senior Member, IEEE), AND MARIA T. ANDRADE[1,3]

[1]Centre for Telecommunications and Multimedia, INESC TEC, 4200-465 Porto, Portugal
[2]School of Engineering, ISEP, Polytechnic of Porto, 4249-015 Porto, Portugal
[3]Faculty of Engineering, University of Porto, 4099-002 Porto, Portugal

Corresponding author: Tiago S. Costa (tiago.a.costa@inesctec.pt)

This work was supported by the National Funds through the Potuguese Funding Agency, Fundação para a Ciência e a Tecnologia (FCT), under Project LA/P/0063/2020. The work of Tiago S. Costa was supported by Fundação da Ciência e Tecnologia under Grant SFRH/BD/144874/2019.

ABSTRACT Quality of Experience (QoE) in multi-view streaming systems is known to be severely affected by the latency associated with view-switching procedures. Anticipating the navigation intentions of the viewer on the multi-view scene could provide the means to greatly reduce such latency. The research work presented in this article builds on this premise by proposing a new predictive view-selection mechanism. A VGG16-inspired Convolutional Neural Network (CNN) is used to identify the viewer's focus of attention and determine which views would be most suited to be presented in the brief term, i.e., the near-term viewing intentions. This way, those views can be locally buffered before they are actually needed. To this aim, two datasets were used to evaluate the prediction performance and impact on latency, in particular when compared to the solution implemented in the previous version of our multi-view streaming system. Results obtained with this work translate into a generalized improvement in perceived QoE. A significant reduction in latency during view-switching procedures was effectively achieved. Moreover, results also demonstrated that the prediction of the user's visual interest was achieved with a high level of accuracy. An experimental platform was also established on which future predictive models can be integrated and compared with previously implemented models.

INDEX TERMS Multimedia, streaming, multi-view, focus-of-attention, deep learning.

I. INTRODUCTION

Interactive multimedia streaming poses numerous challenges, notably those related to latency issues. In multi-view immersive scenarios, latency can be identified as the delay in the process of switching to a new view the user is expecting to see [1]. There is a general agreement that such latency is one of the factors that contributes the most to deteriorating the viewing quality of experience [2]. The research work here described targets precisely the development of a solution to minimize such latency. It proposes a new approach that enables multi-view streaming systems to preemptively gather media content that may suit users' future viewing interests. The main goal is to predict the user's viewing interests to reduce the delay that is normally introduced by traditional, non-predictive, post-tracking reactive systems. The authors have previously explored a reactive buffering strategy [3], [4], [5], interpreting potential viewing interests from tracking data collected in real-time. This research work goes one step forward by introducing a new paradigm focused on predicting which viewing interests might suit the user's needs in the near future and adjusting the system's behavior accordingly. To this aim, we propose the integration of a Deep Learning (DL) approach within already existing view-selection/buffering pipelines. The overall gains obtained confirm the benefits from our approach and the potential for enhancement of existing streaming systems.

The remaining of this paper has the following structure: After having introduced the problem in Section I, Section II presents the current State-of-the-Art on the most relevant

The associate editor coordinating the review of this manuscript and approving it for publication was P. K. Gupta.

topics for this work, namely immersive experiences, tracking applications, and focus estimation; Section III focuses on the introduction of the knowledge-based model for predictive view-selection; Section IV discusses the integration of the model on a multi-view streaming platform and actual performance gains verified throughout evaluation; Section V concludes the presentation of this research work, detailing its most relevant outcomes and outlining future work.

II. RELATED WORK

A. IMMERSIVE EXPERIENCES

Despite the recent advancements in computing capabilities, immersive experiences have yet to reach the state of mass consumption as initially envisioned. This conclusion can be traced to multiple aspects: 1) Usage of dedicated hardware (e.g., Virtual Reality headsets) to deliver such types of experiences [6]; 2) Significant end-user acquisition costs [7]; 3) Health distresses related to virtual experiences [8]. Not even the release of the latest generation of Head-Mounted Devices (HMDs) [9], with technical advancements aimed at improving the overall user experience [10] and bandwidth consumption [11], was capable of overturning the common opinion of the general public [12]. Nonetheless, immersive experiences are not restricted to particular hardware requirements. Other, more cost-effective solutions can also deliver such experiences to wider audiences without incurring any limiting factors. For example, multi-view applications [13] are capable of delivering scenes from multiple perspectives without requiring any additional equipment (e.g., wearable devices) [14]. When coupled with head-tracking hardware, multi-view applications can also deliver suitable perspectives (views) based on the user's viewing interests.

Despite these advantages, multi-view applications also face significant challenges, namely those related to smooth viewing experiences. By delivering specific portions of an integral scene (as opposed to HMDs ''all-or-nothing'' approach [15]), less stringent bandwidth requirements are imposed in these solutions. However, navigation through scenes is highly dependent on network conditions, user feedback, and subsequent view requests, which may introduce a significant delay on the presentation of new views. Thus, achieving an adequate balance between view-switching latency, quality adaptation, and bandwidth management is mandatory to deliver optimal QoE to end-users. Current contributions have yet to address the prediction of user behavior for subsequent time steps. By anticipating decisions based on visual cues from users, these solutions would be able to buffer perspectives which may be requested and reduce associated view-switching latency.

B. TRACKING APPLICATIONS

Accurate tracking of user interests plays a pivotal role within the view prediction process [16], [17]. In fact, tracking technologies were adopted by immersive experiences with the purpose of delivering personalized content based on the user's viewing behavior [18], [19]. Available solutions have already contributed to this goal by focusing on the challenges posed by multi-user detection: 1) Face detection using Haar-like features, enabling tridimensional content visualization with multiple users [20]; 2) Markerless solution for multi-user environments, combining voxelization of dense point cloud data with time-based tracking for accurate user detection [21]. In addition to these solutions, the introduction of complementary layers of data can lead to an improvement in QoE in immersive experiences. For instance, gaze data can be combined with depth information to allow estimation of temporal depth variation in close-range scenarios [22]. Another application exemplifying the usefulness of additional data concerns human-aware attentive object detection: by providing information related to surrounding environments, it enables the accurate identification of possible Regions-of-Interest (ROIs) [23].

While head-tracking devices are most commonly used in immersive experiences, eye-tracking technologies have also been explored in these scenarios. They can provide accurate data of the user's gaze position on screens, being useful in interactive scenarios where resorting to head tracking data is considered insufficient and/or limited. In particular, prior literature has confirmed that generation of heatmaps linked to real user data could potentially lead to a significant decrease in the average distance error for estimated gaze points [24]. Furthermore, the applicability of eye-tracking technologies can also be extended to HMDs. For example, through the correlation of user data with tile-based decoding, identification of suitable ROIs can be achieved with a moderate degree of success [25]. However, such a solution also introduces a significant limitation: segment files must be previously downloaded in order to perform bitstream stitching on each scene before actual decoding is completed. Consequently, applicability is limited to immersive scenarios, in particular those which require low latency and dynamic access to distributed content.

While these contributions have addressed challenges related to real-time tracking of users, accurate estimation of visual attention in immersive experiences remains to be achieved. A balanced association between tracking technologies and specialized solutions for predicting visual attention would allow these solutions to deliver accurate buffering decisions and, ultimately, contribute to a general improvement in QoE.

C. FOCUS ESTIMATION

The interactive nature of immersive experiences poses a significant challenge: gaze tracking, content adaptation, and prediction of visual attention must work in unison to guarantee low latency and optimal QoE for end users. Prior literature has already explored pertinent challenges related to visual attention and content adaptation in immersive experiences: 1) Characterization of eye behavior when exploring visual content [26]; 2) Identification of visual attention,

through combination of head pose data with joint estimation processes, to prevent score degradation in lower resolutions [27]; 3) Correlation between focus of attention and visual discomfort when visualizing immersive content [28]; 4) Differentiation between regions from the visual field, for selective improvement of video performance [29]; 5) Eye-gaze-based interaction techniques, tailored for immersive experiences and applicable to HMDs equipped with eye tracking capabilities [30]; 6) Comparison between gaze guidance techniques, applicable to immersive experiences using eye tracking technologies [31]; 7) Accurate estimation of point-of-gaze in tridimensional environments, through combination of binocular eye tracking devices with stereo stimuli [32].

Immersive experiences have also explored different paths to dynamically adapt content according to users interests. Recent contributions have strayed from decision-based solutions (e.g., tile-based decoding [33]) to DL-based models, trained with historical tracking data, to estimate the focus of attention from users and identify suitable ROIs [34], [35]. For example, Neural Networks (NNs) have been used to adjust resource consumption and optimize content delivery in 360-degree scenarios [36]. They have also been applied to HMDs for estimation of users' head trajectories using historical yaw, pitch, and roll data [37], [38]. However, low-rate accuracy (20%), hampered by non-existent strategies for outlier detection, confirms the existence of a margin for improvement. Additionally, performance from DL-based models can also be impacted by the limited availability (and quality) of datasets in certain scenarios. Most user behavior datasets for immersive experiences are tailored for 360-degree scenarios, using HMDs for the purpose of tracking and collecting Field-of-View (FoV) information from users [39], [40]. In these instances, data such as head movements, behavior patterns, view fixation, and focus direction is available for selected videos. While these datasets can be used by multi-view solutions (after significant adaptation), they reveal the gap in the availability of data for these scenarios. In spite of these limitations, several research efforts have explored DL with multi-view solutions: 1) identification, annotation, and summarization of events using a purpose-built framework, multi-camera setups, and complementary dataset [41]; 2) depth prediction in multi-view streaming, through coupling of encoder-decoder architecture with convolutional long short-term memory (LSTM) cells [42]; 3) estimation of gaze positioning in tridimensional environments, based on the combination of data fusion, NNs and gaze fixation datasets [43].

While these contributions validate the usefulness of DL in multi-view scenarios, they do not address one existing challenge: to what degree can the prediction of users' focus of attention promote the reduction of view-switching latency and improvement of QoE in multi-view solutions? It should be mentioned that, despite its impact on interactivity, latency resulting from predicting the users' focus of attention remains to be addressed by current literature. The following work aims to demonstrate which gains can be obtained, both in terms of view-switching latency and QoE, when combining DL with tracking technologies in multi-view streaming scenarios.

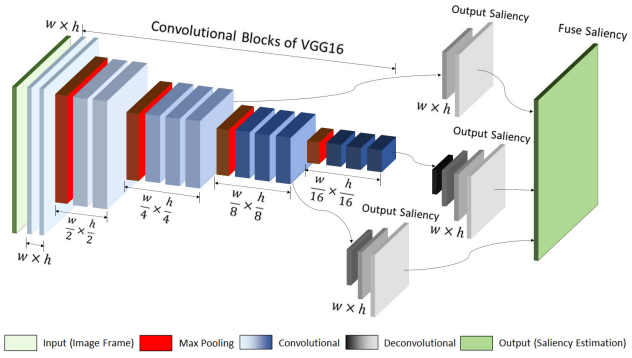

FIGURE 1. VGG16-based CNN, devised to extract potential cues and predict potential viewing interest from users.

III. PREDICTIVE VIEW-SELECTION

A. KNOWLEDGE-BASED MODEL

To guarantee that multi-view streaming solutions can deliver a high degree of user satisfaction with adequate resource consumption, they must be able to quickly respond to variations of user's viewpoint adjustments and adapt content accordingly. The lower the latency introduced during this view-switching process, the higher the offered QoE will be. For this purpose, we have introduced a knowledge-based model which aims to deliver predictions concerning view-selections for the following time steps and, hopefully, contribute to the reduction of existing latency when switching between views. Due to the selective attention nature of the human visual system [44], research on prediction mechanisms applied to human eye fixation has already been explored and well documented. For our case scenario, we have incorporated a knowledge-based model (Figure 1) which relies on skip architecture to capture multi-level saliency response, from local to global regions. The core trainable network adopts an encoder-decoder architecture, using image frames and their corresponding ground truth saliency data as inputs with a fixed size of 224 × 224 pixels. Images are fed through an encoder, composed of a stack of 13 convolutional layers (closely following the structure presented by VGG16), where small receptive fields are used (3 × 3). 4 max-pooling layers were introduced after the first 4 sets of convolutional layers, with 2 × 2 filter and stride 2. After generating feature maps (three-dimensional tensors) through encoding, these are fed through a decoder, consisting of 3 distinct sets of transpose convolutional layers for upsampling feature data (through trainable multi-channel kernels), to deliver an output which matches the original input data resolution. Dimensionality reduction is also achieved on these layers through the compression of encoder feature maps. The last

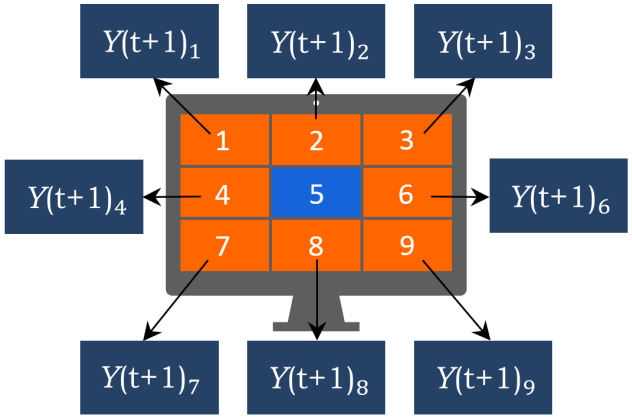

FIGURE 2. Hot&Cold matrix and surrounding views, selectable based on the section highlighted by users [3], [4].

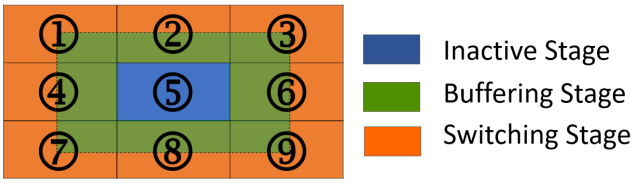

FIGURE 3. Stages from Hot&Cold matrix, indicating the state to be adopted for the next t + Buf segments [3], [4].

layer relies on deep fusion of saliency estimation, sorting, and concatenating data generated by previous layers with multiple scales. The goal of this layer is to determine the most salient region and deliver estimation predictions as the overall outcome of this model. These prediction outputs are then framed against our Hot&Cold matrix concept (Figure 2), which encompasses a predefined arrangement of views to be selected based on the user's gaze positioning within the screen. Identification of views likely to fall within the users' short-term focus of attention is accomplished by translating tridimensional gaze data (e.g. Euler angles) into a bidimensional matrix. Such a matrix maps 9 selectable views from the video to distinct screen areas, placing in the center the view corresponding to the current focus of attention. Additionally, 3 main regions of different area proportions are defined within the screen/matrix (Central, Intermediate and Peripheral) and associated with a specific state: Inactive, Buffering or Switching. By combining estimation predictions with section/state data from the matrix, it can be determined which view will most likely be selected for the following $t+Buf$ time period, as depicted in Figures 3 and 4. Additional details of this approach can be found in previous literature related to SmoothMV [3], [4].

With regard to model optimization, Stochastic Gradient Descent (SGD) was selected after the initial trial delivered better results when compared against other options (e.g., ADAM). Two alternative approaches to SGD were also considered: its standard implementation (progressively varying its initial learning rate throughout training cycles)

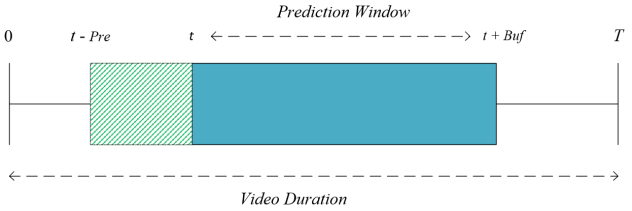

FIGURE 4. Proposed prediction window, applicable to view selection estimations for $t + Buf$ segments, based on real-time user tracking data.

and Stochastic Gradient Descent with Warm Restarts (SGDR) [45]. The latter combines an aggressive annealing schedule (using the cosine function to progressively decay its learning rate between maximum and minimum values) with periodic restarts to the original learning rate.

With η^i_{max} and η^i_{min} defining the optimal range for our learning rate, $T_{current}$ representing the number of epochs since the last restart, calculated at every iteration, and T_i defining the number of epochs within a cycle, we can characterize η_t, the learning rate at a given timestep t with SGDR, as:

$$\eta_t = \eta^i_{min} + \frac{1}{2}(\eta^i_{max} - \eta^i_{min})(1 + cos(\frac{T_{current}}{T_i}\pi)) \quad (1)$$

Regarding learning rates, two options were used with SGD: $1\times10^{(-3)}$ and $1\times10^{(-4)}$, without the application of gradient descent accelerators. As for SGDR, $1\times10^{(-3)}$ was defined as initial learning rate, progressively decaying into its minimum limit, $1\times10^{(-5)}$. For periodic restarts, 3 approaches were defined: 1) Restarts with settings which match starting conditions; 2) Restarts with 10% decay of the initial learning rate, after the completion of each cycle; 3) Restarts with an 1.5× multiplier applied to cycle duration. Graphical representation of the behaviors associated with each learning rate is visible in Figure 5. Hyperparameters (e.g., epochs, batch size) were also subject to consideration. With regard to epochs, after initial consideration (involving a limited set of trial runs with 10, 100 and 500 epochs), 100 epochs were determined to be the optimal value, both in terms of time consumption and performance gains. As for batch size, trial runs were conducted with 10, 20 and 100 batch size. 20 was considered the best compromise for the overall size of selected datasets and available memory (32GB VRAM, from a NVIDIA Tesla V100).

Based on this premise, the introduction of the knowledge-based model was expected to deliver a balanced combination of prediction performance (e.g., gradual reduction of feature dimensions, leading to higher computational efficiency) and accurate results for the proposed application, allowing for a significant reduction of view-switching latency in multi-view scenarios.

B. DATASET SELECTION

The degree of accuracy achieved by knowledge-based models is impacted by the quality of selected datasets.

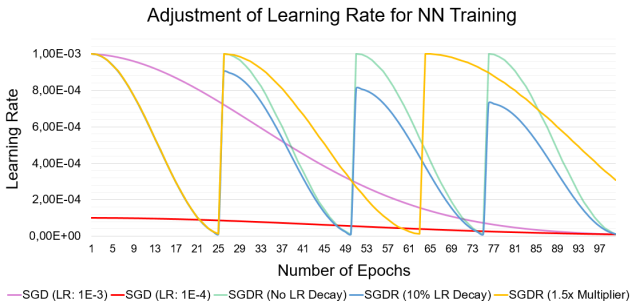

FIGURE 5. Adjustment of learning rates, resorting to Stochastic Gradient Descent (with progressive decay or combined with Warm Restarts).

FIGURE 6. Extraction of 9 views from the original Salient360! dataset material, encoded in HEVC with 1280 × 640 resolution.

TABLE 1. Frames processed from Salient360! dataset.

Source Video	Category	Motion	Total Number of Source Keyframes	Total Number of Cropped Keyframes
1_PortoRiverside	Outdoor, Urban, People	Static	500 frames	4.500 frames
10_Cows	Outdoor, Rural	Static	480 frames	4.320 frames
11_Abbotsford	Indoor, Urban, People	Static	600 frames	5.400 frames
12_TeatroRegioTorino	Indoor, Urban, People	Static	600 frames	5.400 frames
13_Fountain	Outdoor, Urban	Static	600 frames	5.400 frames
14_Warship	Indoor, Urban, People	Static	500 frames	4.500 frames
15_Cockpit	Indoor, Urban, People	Dynamic	500 frames	4.500 frames
2_Diner	Indoor, Urban, People	Static	600 frames	5.400 frames
3_PlanEnergyBioLab	Indoor, Urban, People	Static	500 frames	4.500 frames
4_Ocean	Outdoor, Water, People	Static	600 frames	5.400 frames
5_WaterPark	Outdoor, Urban, People	Dynamic	600 frames	5.400 frames
6_DroneFlight	Outdoor, Urban	Static	500 frames	4.500 frames
7_GazaFishermen	Outdoor, Urban, People	Static	500 frames	4.500 frames
8_Sofa	Indoor, Urban, People	Static	480 frames	4.320 frames
Total (Train/Validation Set)			7.650 frames	68.040 frames
9_MattSwift	Indoor, Urban, People	Static	600 frames	5.400 frames
16_Turtle	Outdoor, Rural, People	Static	600 frames	5.400 frames
17_UnderWaterPark	Outdoor, Natural	Dynamic	600 frames	5.400 frames
18_Bar	Indoor, Urban, People	Dynamic	500 frames	4.500 frames
19_Touvet	Outdoor, Urban	Dynamic	600 frames	5.400 frames
Total (Evaluation Set)			2.900 frames	26.100 frames
Total (Dataset)			10.463 frames	94.140 frames

The availability of appropriate datasets is often problematic for specific tasks. For the purpose of this work, two possibilities were considered: 1) A general-purpose dataset, readily available, built for most computer vision applications; 2) A custom dataset, built using available saliency data extracted from an existing collection of 360-degree video content. Validation of the knowledge-based model with these options would enable us to establish which gains could be obtained when using datasets elaborated for immersive applications. Regarding the first selection, the SALICON dataset [46] was chosen due to its focus on visual neuroscience and cognitive science scenarios. According to its authors, viewing data was collected using a mouse-contingent mechanism which simulated natural viewing behavior from humans for a total of 10.000 images selected from MS COCO [47]. Prior validation was conducted on 700 images, extracted from the OSIE dataset [48].

Despite the interest (and availability) of large-scale datasets (e.g., SALICON), their applicability to immersive experiences is questionable. Video content from these scenarios is composed of image frames, whose reproduction frequency is imposed by technical standards [49]. Frame shifting, along with the general pace set for each video, can lead to content variation which may fluctuate from non-existent to extremely disruptive. Consequently, prior content-based preparation remains critical in order to ensure that models are trained for specific genres [50], [51]. For this purpose, a custom dataset was built from saliency data available on the Salient360! dataset [52], [53], [54], [55]. Selection of this dataset was grounded on the following reasons: 1) Non-existence of multi-view datasets with adequate content duration for evaluation of user behavior (most datasets delivered a few seconds of usable content); 2) Unavailability of multi-view content in high-resolution (e.g., 720p or comparable resolution). According to its authors, source data was collected from 57 users, using HTC Vive VR headset and SensoMotoric Instruments (SMI) eye-tracker [56], with 110° FoV (vertically and horizontally) at 250hz and precision of 0.2°. While this multi-view dataset may not be strictly considered multi-view, its access to a wide variety of scenes in high resolution and 360-degree coverage, provided the most suitable alternative.

The following operations were conducted to source material to enable its utilization with the knowledge-based model:

1) Extraction of keyframes from source video content, focusing on significant changes within each video.
2) Splitting of keyframes into 9 independent sections (each representing a view), with a minimum size of 1280 × 640, as illustrated in Figure 6.
3) Extraction of saliency data for every keyframe, available on complementary binary files for each video, and repetition of steps 1) and 2) to obtain matching heatmaps.
4) Resizing of keyframes and saliency data into 224 × 224 pixels, in order to be fed into the knowledge-based model.
5) Conversion of saliency data to grayscale format using OpenCV [57], with slight adjustments to brightness (4.8% reduction) and contrast (3.9% increase).
6) Refinement of saliency data through measurement of black pixel ratios for every frame. A custom filter, consisting of 95% black region ratio, was applied to eliminate potential learning bias during DL-based training procedures.

These operations translated into an increment in the amount of keyframes capable of being used by the knowledge-based

model. A detailed analysis of the total amount of keyframes extracted from each video is presented in Table 1. Since image content remained mostly unmodified between keyframes on each video, only these were used for training, validation, and evaluation purposes. This would improve overall efficiency by avoiding unnecessary frames, which would not deliver new data compared to what is already available. From the 19 equirectangular videos available in the original dataset, a pool of 14 videos was selected for training/validation during the build process of the custom dataset. This decision was supported by two reasons: 1) Selection of a representative keyframe sample for the custom dataset (e.g., 7.650 keyframes from a grand total of 10.463 keyframes, representing 73.11% of available content); 2) Content selection which mirrored the motion distribution from the original dataset (12 static videos from a total of 14 videos, accounting for 85.71% of selected content). Motion characteristics and category distribution for selected videos can be consulted on Table 1. Lastly, division of training/validation/evaluation data from the selected two datasets was conducted using the following proportion: 1) For the SALICON dataset, 10.000 images were used for training, 9.998 images were selected for validation, and 5.000 images were available for testing; 2) Regarding the Salient360! dataset, after application of the custom filter (black region ratio), 19.578 images were used during training, 6.526 images were selected for validation, and 9.801 images were accessible for evaluation.

C. EVALUATION METRICS

To determine the prediction performance of the knowledge-based model with representative content, 3 metrics were selected. Accuracy [58] and loss [59], [60] (for training, validation, and evaluation), along with Earth Mover's Distance (EMD) [61] were determined for each of the datasets. Specific details are presented as follows:

1) **Accuracy**: Measures the proportion between valid and non-valid solutions (based on available ground truth data), according to a confusion matrix composed by four individual states: True Positive (T_P), True Negative (T_N), False Positive (F_P) and False Negative (F_N). To correctly determine such proportion, differentiation between valid and non-valid cases must be conducted in all evaluated cases, according to the following equation:

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \tag{2}$$

When determining accuracy, an optimal result can be considered when such a model maximizes correct predictions (identified through True Positive or Negative outcomes), while minimizing incorrect predictions (indicated by False Positive or Negative results). Accuracy values range between 0 and 1, with the latter translating into a more accurate solution.

2) **Loss**: Objective function, used for the evaluation of possible candidate solutions based on their lowest error. For each of the previously defined attention layers (4 in total), loss was calculated as described:

$$Loss = \frac{\sum_{i=1}^{j} Loss_{At^i}}{j} \tag{3}$$

To deliver better results with the knowledge-based model, binary cross entropy was used for loss calculation in each layer At^i (identified as $Loss_{At^i}$). In short, it compares predicted probabilities against actual outputs, with their respective distances being used as measurement parameters for the application of potential penalties. Considering x as a logit (function which maps outputs from [0,1] to [-∞,∞]) and z as outputs from models (also known as labels), $Loss_{At^i}$ can be expressed as:

$$Loss_{At^i} = max(x_i, 0) - x_i z_i + log(1 + e^{-|x_i|}) \tag{4}$$

After $Loss_{At^i}$ is determined for each attention layer, an aggregate average is then calculated and used for back propagation. Loss results closer to 0 confirm that a model can evaluate correctly any candidate solution, while reducing possible errors.

3) **Earth Mover's Distance**: Measures the distance between two bidimensional maps, saliency map x and fixation map y, and the least amount of work required to transform the probability distribution from x into y. Let $\mathcal{F}(x, y)$ define the set of flows between the two maps, with matrix $F \in \mathcal{F}(x, y)$ being one feasible flow with matching x and y. The necessary work required by F can be determined with the following equation:

$$WORK(F, x, y) = \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} d_{ij} \tag{5}$$

With $d_{ij} = d(x_i; y_j)$ being the distance between x_i and y_j, the least amount of work required to match x and y can be established, a task which is also referred to as EMD. Such goal is achieved through the following equation:

$$EMD(x, y) = \frac{min_{F=(f_{ij}) \in \mathcal{F}(x,y)} WORK(F, x, y)}{min(w_{\sum}, u_{\sum})} \tag{6}$$

For easier comprehension of results emanating from EMD, values closer to 0 will guarantee a closer connection between two bidimensional maps. Results substantially higher than 0 show that generated predictions are different from ground truth data.

D. EVALUATION RESULTS

Performance results collected during evaluation have confirmed that the knowledge-based model is capable of delivering accurate predictions for multi-view applications.

TABLE 2. Accuracy, Loss and EMD results, with SALICON and custom Salient360! datasets.

Dataset	Training Set	Validation Set	Evaluation Set	Learning Rate	Batch Size	Epochs	Accuracy (Train)	Accuracy (Validation)	Accuracy (Evaluation)	Loss (Train)	Loss (Validation)	Loss (Evaluation)	EMD
SALICON	10.000	9.998	5.000	$1\times10^{(-3)b}$	20	100	0.634	0.634	0.579	0.289	0.289	0.321	0.067
				$1\times10^{(-4)c}$			0.634	0.634	0.579	0.345	0.345	0.384	0.077
				$1\times10^{(-3)d}$			0.634	0.634	0.579	0.620	0.620	0.703	0.115
				$1\times10^{(-3)e}$			0.634	0.634	0.579	0.289	0.289	0.321	0.062
				$1\times10^{(-3)f}$			0.634	0.634	0.579	0.289	0.289	0.324	0.057
Salient360!	19.576^a	6.526^a	9.801^a	$1\times10^{(-3)b}$	20	100	0.928	0.923	0.924	0.068	0.077	0.066	0.010
				$1\times10^{(-4)c}$			0.928	0.923	0.924	0.096	0.106	0.090	0.016
				$1\times10^{(-3)d}$			0.928	0.923	0.924	0.069	0.076	0.065	0.014
				$1\times10^{(-3)e}$			0.928	0.923	0.924	0.069	0.077	0.066	0.012
				$1\times10^{(-3)f}$			0.928	0.923	0.924	0.069	0.077	0.065	0.014

[a] Total amount of selected frames, after refinement of saliency data (black pixel ratio measurement).

[b] Stochastic Gradient Descent, with LR set at $1\times10^{(-3)}$ and $9.40\times10^{(-4)}$ decay applied throughout training.

[c] Stochastic Gradient Descent, with LR set at $1\times10^{(-4)}$ and $4.63\times10^{(-4)}$ decay applied throughout training.

[d] SGD with Warm Restarts, with LR decay between $1\times10^{(-3)}$ and $1\times10^{(-5)}$ and 3 restarts: 25, 50 and 75 epochs.

[e] SGD with Warm Restarts, with LR decay between $1\times10^{(-3)}$ and $1\times10^{(-5)}$, 3 restarts and 10% decay of initial LR after 500 iterations.

[f] SGD with Warm Restarts, with LR decay between $1\times10^{(-3)}$ and $1\times10^{(-5)}$, 2 restarts and 1.5x multiplication applied to cycle duration after restarts.

TABLE 3. View selection comparison, using Salient360! ground truth data and generated predictions.

Videos			Matt Swift	Turtle	Underwater Park	Bar	Touvet
Motion			Static	Static	Dynamic	Dynamic	Dynamic
Category			Indoor, Urban, People	Outdoor, Rural, People	Outdoor, Natural	Indoor, Urban, People	Outdoor, Urban
Distribution	Generated Predictions	View Selection 1	2.400	3.933	2.900	606	2.893
		View Selection 2	0	0	92	449	164
		View Selection 3	1.049	1.130	870	348	576
		View Selection 4	585	20	246	142	143
		View Selection 6	88	10	267	593	354
		View Selection 7	615	34	599	1.556	285
		View Selection 8	45	0	20	264	21
		View Selection 9	618	273	406	542	964
	Ground Truth Data	View Selection 1	1.640	2.445	2.020	2.135	2.378
		View Selection 2	482	396	187	513	493
		View Selection 3	364	443	286	235	148
		View Selection 4	622	487	545	441	625
		View Selection 6	379	356	487	363	382
		View Selection 7	858	231	576	247	529
		View Selection 8	633	750	620	267	514
		View Selection 9	422	292	679	299	331
Frames (Total)			5.400	5.400	5.400	4.500	5.400
Match (%)			23, 4	37, 1	36, 4	9, 3	39, 7

Comparison between the best results from both datasets (depicted in Table 2) has also shown that, when matched against the SALICON dataset, significant gains can be obtained with the custom dataset: 1) Increase of 37.12% in accuracy (e.g. validation accuracy of 0.923, compared to 0.634); 2) Reduction of 109.68% in loss (e.g., validation loss of 0.077, as opposed to 0.264); 3) Improvement of 0.0047 in EMD (e.g., 0.004, compared against 0.051). Furthermore, accuracy and loss curve behavior has depicted a smooth and continuous increase (in accuracy) and decrease (in loss) as the knowledge-based model progressed through each epoch during training, validation, and evaluation tasks. Closer inspection of EMD scores also revealed that measured distances between ground truth data and generated predictions are relatively close with both datasets.

An analysis of the correlation between view-selections determined by generated predictions and ground truth data was also conducted. For this purpose, 5 distinct videos (identified as Evaluation Set in Table 1) were selected, ground truth data was acquired, and the knowledge-based model,

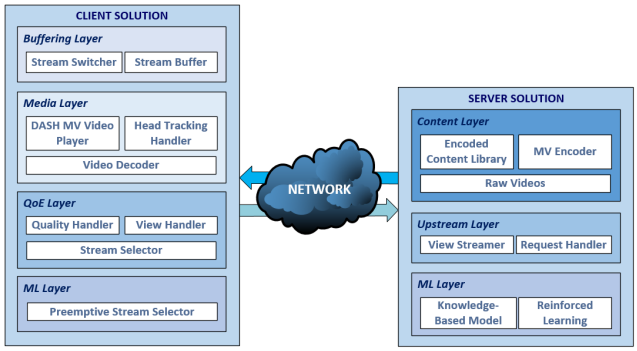

FIGURE 7. SmoothMV architecture, split into client and server roles, with independent functionalities [3], [4].

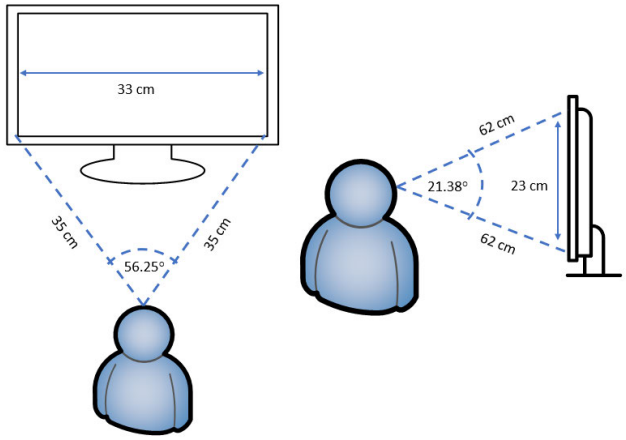

FIGURE 8. Representation of viewing conditions with our multi-view streaming system, with an LCD display installed on a computer desk.

trained with the custom Salient360! dataset and the best performing combination of learning rate ($1\times10^{(-3)}$), decay ($9.40\times10^{(-4)}$) and optimizer (SGD), was used to generate predictions. The most salient point from each image was determined for ground truth data and generated predictions, and framed against the Hot&Cold matrix (Figures 2 and 3), in order to establish potential view-selections. The following methodology was applied for these tasks: A) Split each frame into 9 sections (as per Figure 6); B) Generate predictions for each of those sections using the knowledge-based model, trained with selected learning rate variation, and establish comparisons with ground truth data (also split accordingly into 9 sections). One additional step was also considered: according to Figure 3, the central region from the Hot&Cold matrix is considered as Inactive Stage. As such, salient points enclosed within this region were discarded for view-selection purposes, and most salient points outside of this region were considered.

Result analysis (presented in Table 3) has confirmed a significant correlation between view-selections established by generated predictions and ground truth data. Despite this, matching results delivered by the knowledge-based model also exhibited scope for improvement. This outcome can be explained by the preparation tasks applied to the source material from the Salient360! dataset, and its applicability to the Hot&Cold matrix: the original ground truth data encompassed full-sized frames, which were consequently split into 9 distinct regions. Existing saliency data was then extracted from those regions and used as input data for the model. While accuracy and loss results delivered by the knowledge-based model have validated this approach, saliency variations from generated predictions have resulted in different view-selections established by the Hot&Cold matrix. Ultimately, these conclusions highlight the need for visual attention datasets developed for multi-view scenarios. Saliency data must be acquired for each individual view so that correlations between user visual interest, specific viewing angles, and corresponding view-selections can be established. Detailed comparison results can be consulted in Table 3.

IV. DYNAMIC VIEW MANAGEMENT

A. SYSTEM INTEGRATION

The knowledge-based model was integrated within our multi-view streaming platform, SmoothMV [3], [4], [5] (whose architecture is depicted in Figure 7), to assess the impact on view-switching latency. In its initial form, this streaming platform relied on a rule-based approach to dynamically decide which view should be presented to the user in the immediate future. Its core mechanism was centered on the Hot&Cold matrix: depending on which section of the screen corresponded to the current user's focus of attention, the system state was inferred and decisions were taken concerning the need to request/buffer additional views as well as to switch the view being presented to the user. Despite delivering improved QoE, the adopted solution still presented significant shortcomings (e.g., latency) due to its reactive approach and limited predictive-ability. For the purpose of this work, a new twin-fold approach was introduced: while view-selection predictions from the DL-based model are considered the prime option, the rule-based mechanism is still available as a fallback solution in case these predictions do not adequately represent users' interests. The algorithm encompassing the twin-fold approach is described in Algorithm 1. View-selection predictions originating from the model pM are compared against the user's behavior for each time t. If they match, buffering/switching characteristics specified by stored view predictions are followed: content segments which may be requested by users for the following $t + Buf$ periods will be downloaded into the corresponding buffering (qB) and playback (qP) queues. Since it replicates a sliding window, the procedure must be continuously repeated during content playback in order to confirm if stored view predictions follow the attention behavior demonstrated by users. If they do not match, the algorithm resorts to a preestablished set of rules used to determine which

Algorithm 1 Preemptive View Buffering/Selection

Input: Time $t \geq 0$, buffer interval Buf, playback view vS_t, buffered view vB_t, *Hot&Cold* matrix location mL, inner state iL_t, predictive model pM

Output: $qB_{(t+Buf)}$, $qP_{(t+Buf)}$

Initialization:
$qB_t \leftarrow \varnothing$
$qP_t \leftarrow InitializeQueue(vS_t)$
$vB_t \leftarrow ComputeBufferView(vS_t, mL)$
$lockView, sigFlag \leftarrow InitializeFlag(false)$Plan
if $lockView = false$ **then**
  if $mL = 5$ **then**
    $qB_{(t+Buf)}, qP_{(t+Buf)} \leftarrow Buffer(0, vS_t)$
  else
    foreach $sP_t \in pM$ **do**
      if $sP_t > 0$, $sP_{(t+Buf)} > 0$ **and** $sP_t == vS_t$ **then**
        case iL_t **do**
          1: $qB_{(t+Buf)}, qP_{(t+Buf)} \leftarrow Buffer(sP_{(t+Buf)}, vS_t)$
          2: $qB_{(t+Buf)}, qP_{(t+Buf)} \leftarrow Buffer(vS_t, sP_{(t+Buf)})$
        $sigFlag \leftarrow SignalizeBuffer(true)$
        $lockView \leftarrow SignalizeLocker(true)$
    if $sigFlag == false$ **then**
      case iL_t **do**
        1: $qB_{(t+Buf)}, qP_{(t+Buf)} \leftarrow Buffer(vB^{(mL)}_{(t+Buf)}, vS_t)$
        2: $qB_{(t+Buf)}, qP_{(t+Buf)} \leftarrow Buffer(vS_t, vB^{(mL)}_{(t+Buf)})$
      $lockView \leftarrow SignalizeLocker(true)$

segments should be requested based on the user's positioning within the Hot&Cold matrix (mL) and its inner states (iL). Since this fallback option does not consider which type of content is being presented, it may introduce significant QoE degradation and must only be used as last resort. A locking flag (lockView) was also introduced to prevent undesirable jerkiness. By not allowing any alteration of selected views during a preestablished period of time, the proposed solution diminishes the impact of erratic behavior, maintaining smooth transitions during view-switching procedures.

B. CONTENT PREPARATION

To evaluate view-switching latency within SmoothMV, two multi-view videos were prepared from original, 360-degree content with a corresponding size of 3840 × 2048 pixels: (A) Mercedes-Benz E-Class showcase in Lisbon [62]; (B) a Blue Angels demonstration run [63]. These videos were gathered from Youtube (available under Creative Commons copyright) in equi-rectangular format and encoded in 4K resolution (3840 × 2160 pixels), with 25 frames per second. The reasoning for the selection of these videos was: 1) thematic representativity (e.g., real-life, interactive environments, multiple points of interest); 2) adequacy to evaluation procedures (e.g., high-resolution content, adequate for view extraction, selection, and usage); and 3) similarity to prior content used with the knowledge-based model. Both videos were subjected to cutting/cropping operations, where 6 contiguous views with a size of 1280 × 1024 were extracted, fulfilling cubemap projection requirements for environmental mapping [64] as depicted in Figure 9. Views were subjected to HEVC [65] compression with FFMPEG and libx265 encoder [66]. Despite the higher decoding complexity identified when using HEVC, the benefits of using this codec (e.g., higher PSNR, average bitrate savings, smaller file sizes) outweighed the existing disadvantages. While bitrates from the (A) sequence were contained between 3964-5859 kbps, the (B) sequence delivered bitrates ranging between 3438-4834 kbps. The Group of Pictures (GOP) size was set to 1, to ensure the generation of I-frame-only views and discard the effects of potential GOP-based latency during view-switching procedures. As MPEG-DASH [67], [68] was selected for media streaming purposes, sequence representations were segmented according to the DASH main profile. To determine the correlation between segment sizes and perceived latency, multiple versions of the same input video were defined, each with a distinct segment size (between 33 milliseconds and 10 seconds).

C. SETUP DESCRIPTION

Prior specification of the experimental setup was conducted to provide a level playing field during evaluation. With regard to network conditions, while core networks represent the most significant bottleneck in HTTP-based streaming services, a Gigabit Ethernet Local Area Network was selected. This decision was grounded in the following reasons: 1) Ability to conduct evaluation under controlled conditions (e.g., injection of fake load traffic for congestion simulation) and compare results with previous implementations; 2) Availability of network resources to conduct such experiments. Specifications for hardware and network used throughout evaluation are depicted in Table 4. To assess the impact of buffering, 3 testing variants were defined: A) Streaming without active buffering; B) Streaming with active buffering, resorting to rule-based settings; C) Streaming with active buffering, resorting to DL-based predictions or, in the worst-case scenario, to default view-selection presets (e.g., base estimations). The generation of DL-based predictions was conducted using the best-performing combinations of learning rate, decay, and optimizer for each dataset, as depicted in Table 2: 1) For the SALICON dataset, DL-based predictions were generated using the model trained with the learning rate set to $1 \times 10^{(-3)}$, SGDR optimizer, decay between

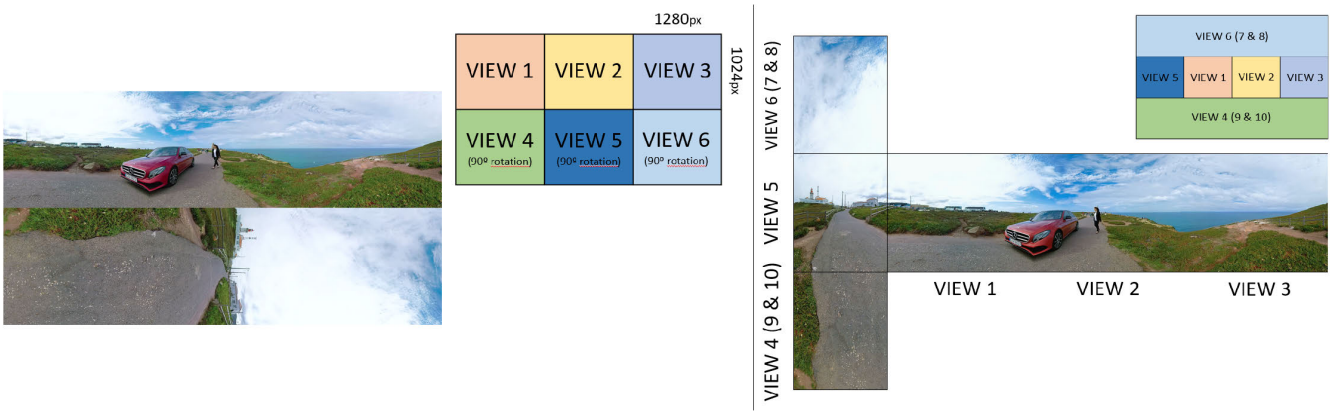

FIGURE 9. One of the multi-view videos used during the evaluation procedures. In this case, the Mercedes-Benz E-Class showcase in Lisbon, with its original format and corresponding cubemap projection conversion.

TABLE 4. Tested specifications for evaluation procedures.

Computer Specifications		
	Server	Client
Role	Server	Client
Device	Desktop	Laptop
O.S.	Windows 8.1 Pro x64	Windows 10 Education x64
CPU	Intel i7-3770 3, 4 GHz	Intel i7-5600U 2, 60 GHz
RAM	16 GB	16 GB
Network Specifications		
Communication Type	Gigabit Ethernet Network	
Router	TP-Link TL-WR1043ND	
Throughput	911 MB/s	
RTT (Minimum)	0 milliseconds	
RTT (Maximum)	0 milliseconds	
RTT (Average)	0 milliseconds	

$1\times10^{(-3)}$ and $1\times10^{(-5)}$ and 10% learning rate decay after 500 iterations; 2) With regard to the Salient360! dataset, predictions were generated using the model trained with a learning rate of $1\times10^{(-3)}$, decay of $9.40\times10^{(-4)}$ and SGD optimizer. In addition, an automated evaluation mechanism was developed and deployed. It enables the replication of users' head tracking movements within the Hot&Cold matrix, with specific order and timing, guaranteeing that evaluations are repeated under identical conditions. Viewing conditions faced by users while observing video content with desktop displays (and replicated by the automated evaluation mechanism) are depicted in Figure 8.

Additional parameter adjustment was also conducted to establish specifications for minimum QoE and guarantee that graphical artifacts and/or interruptions are not verified during playback. Two types of buffering parameters are utilized in SmoothMV: initial segment buffer (sets the number of segments downloaded during the preliminary stages of viewing sessions) and buffering proportion (defines the number of segments buffered during actual streaming sessions). With regard to initial segment buffer, 3 segments were defined as the minimum threshold value. Lowering it below this amount did not decrement QoE nor did it provide any boost in latency performance. Nonetheless, the initial segment buffer had to be increased to 50 and 25 segments when segment size was defined as 33 milliseconds and 100 milliseconds, respectively, to guarantee minimum QoE under these circumstances. As for buffering proportion, it was affected by the combination of the initial segment buffer and segment size. When the lowest segment size was considered (33 milliseconds), buffering proportion was set at 50 segments to ensure minimum QoE. On the opposite end of the spectrum, when segment size was defined as 10 seconds, buffering proportion was decremented to 3 segments. An overview of optimal combinations between segment sizes, initial segment buffer and continuous buffering proportions can be found in Table 5.

D. PERFORMANCE EVALUATION

Measurement of view-switching latency was conducted through video capturing of each evaluation with a tripod-mounted GoPro Hero 3. Resulting videos were subjected to post-processing with VirtualDub [69], where acquisition and registration of timestamps from each given moment (preliminary view request and actual view-switching) were conducted. From this data, the best and worst Top-3 results were collected, and mean latency scores were determined for each condition. The total running time from each evaluation procedure was 150 seconds: the automated evaluation procedure was composed of 50 individual steps, each with a fixed duration of 3 seconds.

Segment loss was also determined for each segment size, with multiple buffering proportion combinations: 3, 25 and 50 segments. Segment loss was considered when a given segment sent from a server was not received and/or played on the client. The assessment of segment loss results would provide an additional layer of data, enabling us to understand

TABLE 5. Latency results from automated switching procedure.

Segment Duration	Buffer Value	Initial Segment Buffer	Buffering Proportion	Active Buffering	Prediction Priority	Dataset Type	Locker Threshold: 6 seconds — 1st Result	2nd Result	3rd Result	4th Result	5th Result	6th Result	Average Delay
33ms	33ms	50 segments	50 segments	-	None	None	520ms	480ms	160ms	800ms	550ms	275ms	464ms
				Active	Rule-Based	None	551ms	250ms	150ms	684ms	467ms	267ms	395ms
					Base Estimations	SALICON	707ms	367ms	651ms	334ms	67ms	200ms	388ms
					Base Estimations	Salient360!	584ms	450ms	333ms	100 ms	80ms	400ms	325ms
					DL-Based Predictions	SALICON	640ms	600ms	476ms	350ms	100 ms	200ms	394ms
					DL-Based Predictions	Salient360!	520ms	200ms	120ms	120ms	480ms	360ms	300ms
					Previous Iteration [4]		1.900ms	1.650ms	1.830ms	1.700ms	1.960ms	2.150ms	1.865ms
100 ms	100 ms	15 segments	25 segments	-	None	None	640ms	760ms	560ms	680ms	480ms	760ms	647ms
				Active	Rule-Based	None	560ms	760ms	760ms	880ms	720ms	920ms	767ms
					Base Estimations	SALICON	520ms	560ms	600ms	560ms	520ms	680ms	573ms
					Base Estimations	Salient360!	520ms	720ms	280ms	560ms	680ms	560ms	553ms
					DL-Based Predictions	SALICON	480ms	640ms	640ms	560ms	600ms	840ms	627ms
					DL-Based Predictions	Salient360!	440ms	680ms	720ms	520ms	360ms	360ms	513ms
					Previous Iteration [4]		1.320ms	1.560ms	2.450ms	2.740ms	2.760ms	2.790ms	2.270ms
500 ms	**100 ms**	**2 segments**	**3 segments**	-	None	None	80ms	200ms	200ms	400ms	280ms	560ms	287ms
				Active	Rule-Based	None	200ms	480ms	520ms	240ms	80ms	80ms	267ms
					Base Estimations	SALICON	120ms	160ms	80ms	200ms	240ms	480ms	213ms
					Base Estimations	Salient360!	120ms	160ms	320ms	120ms	280ms	80ms	180ms
					DL-Based Predictions	SALICON	80ms	200ms	160ms	200ms	80ms	160ms	147ms
					DL-Based Predictions	**Salient360!**	160**ms**	40**ms**	280**ms**	120**ms**	80**ms**	140**ms**	137**ms**
					Previous Iteration [4]		1.930ms	2.390ms	2.420ms	2.800ms	2.300ms	2.830ms	2.445ms
1.000 ms	100 ms	3 segments	3 segments	-	None	None	400ms	640ms	1.120ms	400ms	560ms	840ms	660ms
				Active	Rule-Based	None	360ms	800ms	280ms	520ms	520ms	920ms	567ms
					Base Estimations	SALICON	400ms	680ms	400ms	600ms	280ms	520ms	480ms
					Base Estimations	Salient360!	400ms	720ms	400ms	600ms	240ms	280ms	440ms
					DL-Based Predictions	SALICON	400ms	560ms	320ms	640ms	360ms	600ms	480ms
					DL-Based Predictions	Salient360!	400ms	280ms	640ms	720ms	600ms	440ms	513ms
					Previous Iteration [4]		3.300ms	3.400ms	2.930ms	2.120ms	3.390ms	3.330ms	2.910ms
3.000 ms	100 ms	3 segments	3 segments	-	None	None	4.320ms	4.710ms	4.440ms	4.600ms	2.120ms	1.520ms	3.618ms
				Active	Rule-Based	None	4.600ms	4.680ms	4.880ms	4.440ms	4.640ms	1.600ms	4.140ms
					Base Estimations	SALICON	4.400ms	4.680ms	4.960ms	2.760ms	4.440ms	4.600ms	4.307ms
					Base Estimations	Salient360!	4.360ms	4.600ms	1.120ms	4.320ms	4.720ms	2.480ms	3.600ms
					DL-Based Predictions	SALICON	4.440ms	2.000ms	1.320ms	2.200ms	4.600ms	2.960ms	2.920ms
					DL-Based Predictions	Salient360!	4.400ms	5.040ms	2.480ms	5.440ms	2.400ms	1.920ms	3.613ms
					Previous Iteration [4]		1.320ms	1.480ms	1.500ms	2.930ms	3.980ms	5.080ms	3.050ms
6.000 ms	100 ms	3 segments	3 segments	-	None	None	1.760ms	1.520ms	1.960ms	2.440ms	1.680ms	2.480ms	1.973ms
				Active	Rule-Based	None	1.720ms	1.600ms	2.440ms	480ms	1.560ms	2.320ms	1.687ms
					Base Estimations	SALICON	1.600ms	1.480ms	2.200ms	320ms	1.760ms	200ms	1.260ms
					Base Estimations	Salient360!	1.680ms	1.600ms	2.440ms	1.600ms	2.320ms	320ms	1.660ms
					DL-Based Predictions	SALICON	1.560ms	2.440ms	2.360ms	2.360ms	1.760ms	2.440ms	2.153ms
					DL-Based Predictions	Salient360!	1.600ms	2.320ms	1.840ms	2.440ms	1.680ms	320ms	1.700ms
					Previous Iteration [4]		3.880ms	3.490ms	3.230ms	3.280ms	4.350ms	4.930ms	3.860ms
10.000 ms	100 ms	3 segments	3 segments	-	None	None	1.480ms	2.680ms	1.520ms	1.720ms	720ms	1.760ms	1647ms
				Active	Rule-Based	None	1.600ms	2.840ms	640ms	1.640ms	1.840ms	3.040ms	1.933ms
					Base Estimations	SALICON	2.600ms	640ms	3.880ms	3.400ms	3.440ms	520ms	2.413ms
					Base Estimations	Salient360!	2.520ms	480ms	2.520ms	4.600ms	4.600ms	1.440ms	2.693ms
					DL-Based Predictions	SALICON	4.360ms	600ms	4.080ms	520ms	4.360ms	400ms	2.387ms
					DL-Based Predictions	Salient360!	4.120ms	360ms	2.640ms	3.920ms	2.520ms	3.880ms	2.907ms
					Previous Iteration [4]		8.910ms	8.030ms	7.360ms	7.120ms	6.700ms	7.890ms	7.670ms

TABLE 6. Percentage of segment loss during playback (per buffer size).

Initial Seg. Buffer (per Seg. Size)	Segment Duration (ms)																				
	33			100			500			1.000			3.000			6.000			10.000		
	Buffering Proportion (segments)																				
	3	25	50	3	25	50	3	25	50	3	25	50	3	25	50	3	25	50	3	25	50
1	99,9	99,9	99,9	99,7	99,7	99,6	99,2	97,2	77,8	99,0	12,1	99,0	97,0	36,4	74,2	90,9	72,7	100	10,0	100	100
100	98,5	91,5	78,3	95,2	0,3	0,3	0,0	0,0	2,0	0,5	0,5	4,6	1,5	6,1	12,1	3,0	15,2	15,2	5,0	15,0	15,0
500	93,4	86,5	62,4	74,9	0,4	0,3	0,3	0,3	2,0	0,5	0,5	4,0	1,5	7,6	12,1	3,0	15,2	15,2	5,0	15,0	15,0
1.000	85,0	74,1	57,7	49,7	0,3	0,3	0,3	0,3	2,3	0,5	0,5	4,6	1,5	7,6	10,6	3,0	15,2	15,2	5,0	15,0	15,0
1.500	76,6	66,1	53,9	29,8	0,3	0,3	0,3	0,3	2,3	0,5	0,5	4,6	1,5	7,6	12,1	3,0	15,2	15,2	5,0	15,0	15,0
2.500	57,8	39,9	40,0	17,5	0,3	0,5	0,3	0,0	2,0	0,5	0,5	4,6	1,5	7,6	12,1	3,0	15,2	15,2	5,0	15,0	15,0
3.000	51,3	33,8	34,3	17,6	0,3	0,2	0,3	0,0	2,3	0,5	0,0	4,0	1,5	7,6	12,1	3,0	15,2	15,2	5,0	15,0	15,0
4.000	35,7	23,3	23,9	17,6	0,3	0,3	0,3	0,0	2,0	0,5	0,5	4,6	1,5	7,6	10,6	3,0	15,2	15,2	5,0	15,0	15,0
5.000	26,3	17,9	17,1	17,6	0,3	0,3	0,3	0,0	2,3	0,5	0,5	4,6	1,5	7,6	12,1	3,0	15,2	15,2	5,0	15,0	15,0
6.000	18,2	12,3	12,5	17,4	0,3	0,3	0,3	0,3	2,3	0,5	0,5	4,0	1,5	7,6	12,1	3,0	15,2	15,2	5,0	15,0	15,0

the effects of potential combinations between segment size, initial segment buffer, and buffering proportion on the overall performance of the multi-view streaming platform.

E. RESULTS ANALYSIS

Analysis of evaluation results (depicted in Table 5) has shown that optimal view-switching latency was achieved with the following parameters: 1) Segment size set as 500 milliseconds; 2) Initial segment buffer of 2 segments; 3) Continuous buffer set to 3 segments; 4) Predictions from the knowledge-based model, trained with the Salient360! dataset. Under these conditions, minimum view-switching latency of 40 milliseconds was achieved, along with an average view-switching latency of 137 milliseconds. Using the SALICON dataset, the minimum latency was increased to 80 milliseconds, with average latency reaching 147 milliseconds. These results confirm the conclusions from recent works, which delved into the impact of lag on perceived QoE [70]. By utilizing shorter buffering periods (e.g., 500 milliseconds or less), lag effects are minimized and quicker response times to user requests are guaranteed. As for the worst-case scenario, it was encountered when segment size was set to 3 seconds, with initial buffering and continuous buffering locked at 3 segments. Under these conditions, the best minimum latency was achieved with the rule-based mechanism: 1.120 milliseconds. However, the best average latency score (2.920 milliseconds) was obtained with the knowledge-based model trained with the SALICON dataset.

With regard to comparisons between decision-based mechanism and knowledge-based model, the latter was able to deliver greater latency gains. In the best-case scenario (segment size set at 500 milliseconds), the non-DL mechanism delivered an average latency of 287 milliseconds, significantly higher compared to the 137 milliseconds of the DL-based approach. Nevertheless, in one particular scenario, this outcome was not verified: when segment size was set to 10 seconds. Under these conditions, the non-DL mechanism reached an average latency of 1.467 milliseconds, an improvement over the 2.387 milliseconds from the knowledge-based model trained with the SALICON dataset. Comparing to the previous iteration of the multi-view streaming system, significant latency gains were disclosed [4]: decrease both in terms of minimum latency (from 1.320 milliseconds to 40 milliseconds) and average latency (from 1.865 milliseconds to 137 milliseconds). Results from the previous iteration were obtained with buffering activated, segment size set to 100 milliseconds and using the decision-based mechanism. Despite the theoretical disadvantage from the current, best-case scenario (due to the larger segment size, 500 milliseconds), introduction of the knowledge-based model, coupled with on-going development efforts on SmoothMV, has led to significant boost in view-switching latency performance.

With regard to segment loss (depicted in Table 6), an increase of initial segment buffer was positively reflected in the decrease of non-reproduced segments at the client. This reduction was stabilized when segment duration and buffering proportion were set at 100 milliseconds and 25 segments, respectively. With these conditions and the initial buffering proportion set at 100 segments, 99% of available segments for playback were delivered and reproduced. On the opposite end, when the initial segment buffer was set at 1 segment, high segment losses were encountered, surpassing 90% in most cases. These results indicate that, to provide a smooth viewing experience, an adequate amount of pre-buffering must be conducted beforehand. This ensures a smooth transition between initial preparation and actual reproduction of multimedia content due to the continuous delivery of requested segments.

In conclusion, these evaluation results have validated the applicability of the knowledge-based model in multi-view streaming scenarios and the performance gains obtained with its integration. Additionally, they have also confirmed that fine-tuning parameters such as initial segment buffer, segment duration, and buffering proportion can lead to an effective increase in QoE, through optimization of view-switching latency and segment loss.

V. CONCLUSION

In the work presented throughout this article, we were able to establish the following conclusion: the introduction of knowledge-based models has the potential to unlock relevant gains in view-switching latency, segment loss, and QoE, when applied to multi-view streaming platforms. In addition, a set of potential research cues were identified derived from the general lack of visual attention datasets tailored for multi-view applications. Future research work will delve into the development of a brand-new visual attention dataset by gathering tracking data from real users with multi-view content. To achieve such a purpose, a custom testing scenario will be specified, with testing procedures being conducted at INESC TEC's premises with a broad selection of individuals. The adoption of complementary techniques to increase the availability and quantity of visual attention data will also be explored, namely transfer learning and data augmentation techniques. Hopefully, by following these approaches, the knowledge-based model can be further optimized to deliver more accurate view-selection predictions. Furthermore, by potentially expanding the custom dataset used in this work with new data, its relevance for the development of new DL-based models could be significantly increased, particularly when multi-view applications are considered. Aside from the development of this dataset, research on recurrent models will also be conducted. While the current knowledge-based model has delivered predictions with a high level of accuracy, the introduction of recurrent learning mechanisms with complementary temporal data (depicted as $t - Pre$ in Figure 4) will increase the existing knowledge on user interests and provide more accurate view-selection predictions under identical conditions. Based on these assumptions, research work on recurrent models will be conducted alongside the development efforts for the generation of the new multi-view dataset.

REFERENCES

[1] A. Yaqoob, T. Bi, and G.-M. Muntean, "A survey on adaptive 360° video streaming: Solutions, challenges and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 4, pp. 2801–2838, 4th Quart., 2020.

[2] T. Siglin. (2022). *The State of Real-Time Streaming*. Streaming Media. Accessed: 1, 2023. [Online]. Available: https://www.streamingmedia.com/Articles/Editorial/Featured-Articles/The-State-of-Real-Time-Streaming-2022-152089.aspx

[3] T. S. Costa, M. T. Andrade, and P. Viana, "Predictive multi-view content buffering applied to interactive streaming system," *Electron. Lett.*, vol. 55, no. 15, pp. 837–838, 2019.

[4] T. S. Costa, M. T. Andrade, and P. Viana, "SmoothMV: Seamless content adaptation through head tracking analysis and view prediction," in *Proc. Int. Workshop Immersive Mixed Virtual Environ. Syst.*, 2021, pp. 8–13.

[5] T. S. Costa and M. T. Andrade, "Gaze-based personalized multi-view experiences," *J. Media Mass Commun.*, vol. 1, no. 1, pp. 43–47, 2015.

[6] M. Y. Huang and K. Webb. (2018). *The Reason Virtual Reality Still Hasn't Taken Off*. Business Insider. Accessed: Aug. 1, 2023. [Online]. Available: https://www.businessinsider.com/reason-virtual-reality-hasnt-taken-off-future-technology-2018-11

[7] D. Karpf. (2021). *Virtual Reality is the Rich White Kid of Technology*. Wired Magazine. Accessed: Aug. 1, 2023. [Online]. Available: https://www.wired.com/story/virtual-reality-rich-white-kid-of-technology/

[8] A. Jenkins. (2019). *The Fall and Rise of VR: The Struggle to Make Virtual Reality Get Real*. Fortune Magazine. Accessed: Aug. 1, 2023. [Online]. Available: https://fortune.com/longform/virtual-reality-struggle-hope-vr/

[9] C. Flavián, S. Ibáñez-Sánchez, and C. Orús, "The impact of virtual, augmented and mixed reality technologies on the customer experience," *J. Bus. Res.*, vol. 100, pp. 547–560, Jul. 2019.

[10] J. Radianti, T. A. Majchrzak, J. Fromm, and I. Wohgenannt, "A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda," *Comput. Eduv.*, vol. 147, Apr. 2020, Art. no. 103778.

[11] A. Pennington. (2021). *VR? AR? Today, It's All About XR*. Streaming Media. Accessed: Aug. 1, 2023. [Online]. Available: https://www.streamingmedia.com/Articles/Editorial/Featured-Articles/VR-AR-Today-Its-All-About-XR-145251.aspx

[12] J. Weatherbed. (2023). *Playstation VR2 Sales Expectations Reportedly Halved After Disappointing Preorders*. The Verge. Accessed: Aug. 1, 2023. [Online]. Available: https://www.theverge.com/2023/1/31/23579292/sony-playstation-vr2-psvr2-sales-low-preorders-price

[13] C. Ozcinar, E. Ekmekcioglu, and A. Kondoz, "Dynamic adaptive 3D multi-view video streaming over the internet," in *Proc. ACM Int. Workshop Immersive Media Experiences*, Oct. 2013, pp. 51–56.

[14] D. Yun and K. Chung, "DASH-based multi-view video streaming system," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1974–1980, Aug. 2018.

[15] L. Wang, X. Shi, and Y. Liu, "Foveated rendering: A state-of-the-art survey," *Comput. Vis. Media*, vol. 9, no. 2, pp. 195–228, Jun. 2023.

[16] J. Steil, P. Müller, Y. Sugano, and A. Bulling, "Forecasting user attention during everyday mobile interactions using device-integrated and wearable sensors," in *Proc. 20th Int. Conf. Human-Computer Interact. Mobile Devices Services*, Sep. 2018, pp. 1–24.

[17] P. Elbe, D. E. Sörman, E. Mellqvist, J. Brändström, and J. K. Ljungberg, "Predicting attention shifting abilities from self-reported media multitasking," *Psychonomic Bull. Rev.*, vol. 26, no. 4, pp. 1257–1265, Aug. 2019.

[18] Z. Hu, S. Li, and M. Gai, "Temporal continuity of visual attention for future gaze prediction in immersive virtual reality," *Virtual Reality Intell. Hardw.*, vol. 2, no. 2, pp. 142–152, Apr. 2020.

[19] A. Palazzi, D. Abati, S. Calderara, F. Solera, and R. Cucchiara, "Predicting the driver's focus of attention: The DR(eye)VE project," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1720–1733, Jul. 2019.

[20] S.-J. Kang, Y.-W. Jeong, J.-J. Yun, and S. Bae, "Real-time eye tracking technique for multiview 3D systems," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2016, pp. 1–2.

[21] D. Bicho, P. Girão, J. Paulo, L. Garrote, U. J. Nunes, and P. Peixoto, "Markerless multi-view-based multi-user head tracking system for virtual reality applications," in *Proc. IEEE Int. Conf. Syst., Man Cybern. (SMC)*, Oct. 2019, pp. 2645–2652.

[22] T.-S. Kuo, K.-T. Shih, S.-L. Chung, and H. H. Chen, "Depth from gaze," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 2910–2914.

[23] D. Y. Cho and M. K. Kang, "Human gaze-aware attentive object detection for ambient intelligence," *Eng. Appl. Artif. Intell.*, vol. 106, Nov. 2021, Art. no. 104471.

[24] X. Luan, B. Zhang, D. Liu, X. Liu, X. Tong, and K. Li, "A lightweight heatmap-based eye tracking system," in *Proc. Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2021, pp. 1–9.

[25] H. Kim, J. Yang, J. Lee, S. Yoon, Y. Kim, M. Choi, J. Yang, E.-S. Ryu, and W. Park, "Eye tracking-based 360 VR foveated/tiled video rendering," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2018, p. 1.

[26] K. Rayner, "Eye movements and attention in reading, scene perception, and visual search," *Quart. J. Experim. Psychol.*, vol. 62, no. 8, pp. 1457–1506, 2009.

[27] Y. Huang, D. Duan, J. Cui, F. Davoine, L. Wang, and H. Zha, "Joint estimation of head pose and visual focus of attention," in *Proc. Int. Conf. Image Process.*, 2014, pp. 3332–3336.

[28] S. Ahn, J. Kim, H. Kim, and S. Lee, "Visual attention analysis on stereoscopic images for subjective discomfort evaluation," in *Proc. Int. Conf. Multimedia Expo*, 2016, pp. 1–6.

[29] L. Zhang, Q. Sun, S. Wang, S. Su, and F. Yang, "Towards video quality of experience and selective attention: A subtitle-based measurement study," in *Proc. IEEE Int. Conf. Services Comput. (SCC)*, Jun. 2016, pp. 872–875.

[30] T. Piumsomboon, G. Lee, R. Lindeman, and M. Billinghurst, "Gaze guidance in immersive environments," in *Proc. Int. Symp. 3D User Interfaces*, 2017, pp. 36–39.

[31] S. Grogorick, G. Albuquerque, and M. Maqnor, "Gaze guidance in immersive environments," in *Proc. Int. Conf. Virtual Reality 3D User Interfaces*, 2018, pp. 563–564.

[32] J. Sun, Z. Wu, H. Wang, P. Jing, and Y. Liu, "A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–15, 2022.

[33] P. Maniotis, E. Bourtsoulatze, and N. Thomos, "Tile-based joint caching and delivery of 360° videos in heterogeneous networks," in *Proc. IEEE 21st Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2019, pp. 1–6.

[34] A. Nguyen, Z. Yan, and K. Nahrstedt, "Your attention is unique: Detecting 360-degree video saliency in head-mounted display for head movement prediction," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1190–1198.

[35] Y. Xu, Y. Dong, J. Wu, Z. Sun, Z. Shi, J. Yu, and S. Gao, "Gaze prediction in dynamic 360° immersive videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5333–5342.

[36] M. Xu, Y. Song, J. Wang, M. Qiao, L. Huo, and Z. Wang, "Predicting head movement in panoramic video: A deep reinforcement learning approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2693–2708, Nov. 2019.

[37] C. C. Flintoft, C. Barentine, J. Touryan, and A. J. Ries, "A case for studying naturalistic eye and head movements in virtual environments," *Frontiers Psychol.*, vol. 12, Dec. 2021, Art. no. 650693.

[38] S. Petrangeli, G. Simon, and V. Swaminathan, "Trajectory-based viewport prediction for 360-degree virtual reality videos," in *Proc. IEEE Int. Conf. Artif. Intell. Virtual Reality (AIVR)*, Dec. 2018, pp. 157–160.

[39] C. Wu, Z. Tan, Z. Wang, and S. Yang, "A dataset for exploring user behaviors in VR spherical video streaming," in *Proc. 8th ACM Multimedia Syst. Conf.*, Jun. 2017, pp. 193–198.

[40] C.-L. Fan, J. Lee, W.-C. Lo, C.-Y. Huang, K.-T. Chen, and C.-H. Hsu, "Fixation prediction for 360° video streaming in head-mounted virtual reality," in *Proc. 27th Workshop Netw. Operating Syst. Support Digit. Audio Video*, Jun. 2017, pp. 67–72.

[41] M. Elfeki, L. Wang, and A. Borji, "Multi-stream dynamic video summarization," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 185–195.

[42] A. Duzçeker, S. Galliani, C. Vogel, P. Speciale, M. Dusmanu, and M. Pollefeys, "DeepVideoMVS: Multi-view stereo on video with recurrent spatio-temporal fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15319–15328.

[43] A. Ali and Y.-G. Kim, "Deep fusion for 3D gaze estimation from natural face images using multi-stream CNNs," *IEEE Access*, vol. 8, pp. 69212–69221, 2020.

[44] D. Bull and F. Zhang, *Intelligent Image and Video Compression*, 2nd ed. London, U.K.: Academic Press, 2021.

[45] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," 2016, *arXiv:1608.03983*.

[46] M. Jiang, S. Huang, J. Duan, and Q. Zhao, "SALICON: Saliency in context," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1072–1080.

[47] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[48] J. Xu, M. Jiang, S. Wang, M. S. Kankanhalli, and Q. Zhao, "Predicting human gaze beyond pixels," *J. Vis.*, vol. 14, no. 1, pp. 1–20, 2014.

[49] A. Mackin, F. Zhang, and D. R. Bull, "A study of high frame rate video formats," *IEEE Trans. Multimedia*, vol. 21, no. 6, pp. 1499–1512, Jun. 2019.

[50] E. Fish, J. Weinbren, and A. Gilbert, "Rethinking genre classification with fine grained semantic clustering," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 1274–1278.

[51] J. Cha, "Does genre type influence choice of video platform? A study of college student use of internet and television for specific video genres," *Telematics Informat.*, vol. 30, no. 2, pp. 189–200, May 2013.

[52] J. Gutiérrez, E. J. David, A. Coutrot, M. P. Da Silva, and P. L. Callet, "Introducing UN salient360! Benchmark: A platform for evaluating visual attention models for 360° contents," in *Proc. 10th Int. Conf. Quality Multimedia Exper. (QoMEX)*, May 2018, pp. 1–3.

[53] E. J. David, J. Gutiérrez, A. Coutrot, M. P. Da Silva, and P. L. Callet, "A dataset of head and eye movements for 360° videos," in *Proc. 9th ACM Multimedia Syst. Conf.*, Jun. 2018, pp. 1–6.

[54] Y. Rai, P. Le Callet, and P. Guillotel, "Which saliency weighting for omni directional image quality assessment?" in *Proc. 9th Int. Conf. Quality Multimedia Exper. (QoMEX)*, May 2017, pp. 1–6.

[55] Y. Rai, J. Gutiérrez, and P. Le Callet, "A dataset of head and eye movements for 360 degree images," in *Proc. 8th ACM Multimedia Syst. Conf.*, Jun. 2017, pp. 205–210.

[56] P. A. Punde, M. E. Jadhav, and R. R. Manza, "A study of eye tracking technology and its applications," in *Proc. 1st Int. Conf. Intell. Syst. Inf. Manag. (ICISIM)*, Oct. 2017, pp. 86–90.

[57] I. Culjak, D. Abram, T. Pribanic, H. Dzapo, and M. Cifrek, "A brief introduction to OpenCV," in *Proc. 35th Int. Conv. MIPRO*, May 2012, pp. 1725–1730.

[58] A. F. Gad. (2020). *Evaluating Deep Learning Models: The Confusion Matrix, Accuracy, Precision, and Recall*. Accessed: Aug. 1, 2023. [Online]. Available: https://blog.paperspace.com/deep-learning-metrics-precision-recall-accuracy/

[59] TensorFlow. (2022). *Sigmoid Cross Entropy With Logits*. Accessed: Aug. 1, 2023. [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/nn/sigmoid_cross_entropy_with_logits/

[60] R. Khan. (2019). *How do Tensorflow and Keras Implement Binary Classification and the Binary Cross-Entropy Function?* Accessed: Aug. 1, 2023. [Online]. Available: https://rafayak.medium.com/how-do-tensorflow-and-keras-implement-binary-classification-and-the-binary-cross-entropy-function-e9413826da7

[61] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *Int. J. Comput. Vis.*, vol. 40, no. 2, pp. 99–121, Nov. 2000.

[62] Mercedes-Benz. (2016). *360° Video Drive in the Mercedes-Benz EClass | Lisbon*. Youtub. Accessed Aug. 1, 2023. [Online]. Available: https://www.youtube.com/watch?v=Za78gJU5Tfc

[63] USA TODAY. (2015). *Experience the Blue Angels in 360-Degree Video | USA TODAY*. YouTube. Accessed: Aug. 1, 2023. [Online]. Available: https://www.youtube.com/watch?v= H6SsB3JYqQg

[64] T. El-Ganainy and M. Hefeeda, "Streaming virtual reality content," 2016, *arXiv:1612.08350*.

[65] IT Union. (2015). *H.265: High Efficiency Video Coding*. Accessed: Aug. 1, 2023. [Online]. Available: http://www.itu.int/rec/T-REC-H.265

[66] (2022). *x265 HEVC Encoder/X265 Video Codec*. Accessed: Aug. 1, 2023. [Online]. Available: http://x265.org/

[67] T. Stockhammer, "Dynamic adaptive streaming over HTTP—Standards and design principles," in *Proc. 2nd Annu. ACM Conf. Multimedia Syst.*, Feb. 2011, pp. 133–144.

[68] X. Zhang, L. Toni, P. Frossard, Y. Zhao, and C. Lin, "Adaptive streaming in interactive multiview video systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 4, pp. 1130–1144, Apr. 2019.

[69] (2022). *VirtualDub*. Accessed: Aug. 1, 2023. [Online]. Available: https://www.virtualdub.org/

[70] T. Nunome and T. Ito, "The effect of time lag between servers on QoE of multi-view video and audio transmission," in *Proc. IEEE Int. Conf. Consum. Electronics-Taiwan (ICCE-TW)*, Sep. 2021, pp. 1–2.

TIAGO S. COSTA was born in Porto, Portugal, in 1985. He received the B.S. and M.S. degrees in informatics engineering from the School of Engineering, Polytechnic of Porto, Portugal, in 2007 and 2009, where he is currently pursuing the Ph.D. degree with the Faculty of Engineering. Since 2010, he has been a Conduction Active Researcher with the INESC TEC Research Institute, Porto, in both national and European level projects. He was nominated as an Invited Lecturer with the School of Technology and Management, Polytechnic of Porto, from 2017 to 2020, and Fernando Pessoa University, Porto, from 2021 to 2022. He has been the author of multiple international publications. His research interests include cutting-edge immersive technologies, adaptive content delivery, multimedia streaming protocols, 3-D and multiview video streaming, interactive solutions for multimedia scenarios, bleeding edge game development, mobile computing, and user-oriented software development.

PAULA VIANA (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Porto, in 2008. She is currently a Coordinator Professor with the School of Engineering, Polytechnic of Porto, and the Head of the Multimedia Communication Technologies, INESC TEC. She has over 30 years of experience in the area of multimedia content analysis and management, computer vision, and multimedia metadata. She has been coordinating the participation of INESC TEC in several national and European projects. She is the author of several publications and an active reviewer of journals, conferences, and European and Portuguese research projects. She has involved in the organization of several scientific events, including the Immersive Media Experiences Workshop Series (2013–2015) at ACM Multimedia.

MARIA T. ANDRADE was born in Porto, Portugal, in 1963. She received the B.S., M.S., and Ph.D. degrees in electrotechnical and computing engineering from the Telecommunications Branch, University of Porto, Portugal, in 1986, 1990, and 2008, respectively. Since 1986, she has been conducting research within the framework of national and European level projects, as a Researcher with the INESC TEC Research Institute, Porto. She has been with the Electrotechnical and Computing Department, Faculty of Engineering, University of Porto, since 1996, as a Lecturer until 2008 and currently an Assistant Professor. She is the coauthor of three books and more than 80 articles. Her research interests include context-awareness, mobile and adaptable multimedia applications in heterogeneous environments, 3-D and multiview video streaming, quality of service and experience in multimedia services, semantic technologies and content recommendation, digital television, digital cinema and new media, and the Internet of Multimedia Things (IoMMT).