Supplementary Material for Spherical Vision Transformer for 360° Video Saliency Prediction

Mert Cokelek[1]
mcokelek21@ku.edu.tr
Nevrez Imamoglu[2]
https://nevrez.github.io
Cagri Ozcinar[3]
https://cagriozcinar.netlify.app
Erkut Erdem[4]
https://web.cs.hacettepe.edu.tr/~erkut
Aykut Erdem[1]
https://aykuterdem.github.io

[1] KUIS AI Center, Koç University
Istanbul, TR
[2] AIST
Tokyo, JP
[3] MSK.AI
London, UK
[4] Hacettepe University
Istanbul, TR

The purpose of this document is to provide extra material to complement the main paper. In Sec. 1, we present additional experiments on the proposed *Viewport Spatio-Temporal Attention (VSTA)* mechanism. In Sec. 2, we give the implementation details of our proposed *Viewport Augmentation Consistency (VAC)* loss. Finally, in Sec. 3, we evaluate our method and the competing approaches on a downstream task, which involves assessing the visual quality of omnidirectional videos. The code and pre-trained models will be available.

1 Viewport Spatio-Temporal Self-Attention

1.1 Temporal Window Size

To investigate the impact of temporal window size (F) on the representational power of our omnidirectional video saliency prediction model, we vary the number of frames in a video clip, and analyze its effect. These experiments are performed on our VSTA baseline, which consists of 6 transformer blocks with an embedding dimension of $D = 512$ and 8 attention heads. We present the results of our experiments using four saliency evaluation metrics in Fig. 1, providing insights into the performance of our model across different temporal window sizes.

Fig. 1 shows that increasing temporal window size gradually leads to a performance boost in three metrics and an insignificant performance drop in NSS for $F > 2$. We conclude our experiments at $F = 8$, considering the memory limit of a single Tesla V100 GPU.

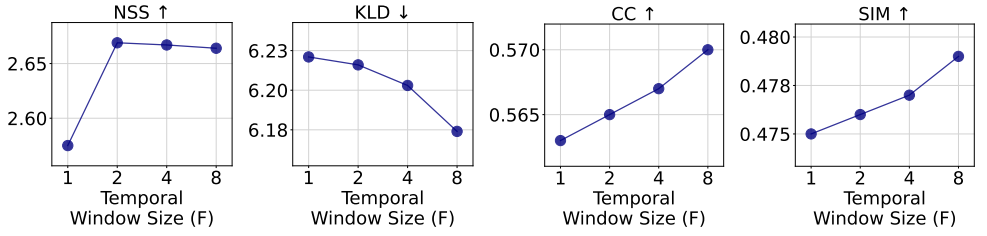

Figure 1: Performance of our ODV saliency prediction model in terms of four evaluation metrics *(NSS, KLD, CC, SIM)* as a function of temporal window size (*F*) on the validation split of VR-EyeTracking [1] dataset.

1.2 Transformer Depth

We analyze the influence of the number of transformer blocks on the performance and the computational complexity of our saliency prediction model for omnidirectional videos. In Fig. 2, we present our experimental results, showing how the performance varies with different numbers of transformer blocks.

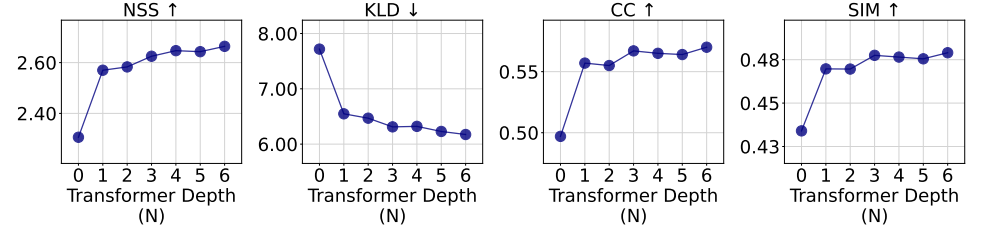

Figure 2: Performance of our ODV saliency prediction model in terms of four evaluation metrics *(KLD, NSS, CC, SIM)* as a function of VSTA depth (*N*) on the validation split of VR-EyeTracking dataset.

In Fig. 2, the performance gain from $N = 0$ to $N = 1$, highlights the effectiveness of our proposed VSTA mechanism for saliency prediction in omnidirectional videos. Notably, even with a single VSTA transformer block, our model shows the ability to capture rich 360° spatio-temporal features. As the depth of the transformer blocks increases, the model performance continues to improve. However, we conclude our experiments after $N = 6$ transformer blocks, taking into consideration the model size as reported in Table 1.

1.3 Comparison with Joint Spatio-Temporal Attention

Table 1 compares our proposed VSA and VSTA mechanisms with joint spatio-temporal attention, which computes self-attention among all frames and tokens in a video clip. Since VSTA computes spatio-temporal attention in two stages (time and space), the model size becomes larger than VSA / JSTA. On the other hand, VSTA is computationally more efficient as its complexity grows linearly with respect to temporal window size, where it grows quadratically in JSTA.

Attention	# params	GFLOPs	NSS↑	KLD↓	CC↑	SIM↑
None	11.81M	0.00	2.306	7.718	0.497	0.434
VSA	30.78M	57.08	2.575	6.221	0.563	0.475
VSTA	37.07M	63.30	2.664	6.174	0.570	0.479
JSTA	30.78M	77.57	n/a	n/a	n/a	n/a

Table 1: **Quantitative comparison** for space-only attention, the proposed Viewport Spatio-Temporal Attention and existing Joint Spatio-Temporal attention for omnidirectional video saliency prediction on the validation split of VR-EyeTracking dataset. Due to memory limitations, JSTA model could not be trained on our GPU.

2 Viewport Augmentation Consistency (VAC)

In this section, we describe the proposed *Viewport Augmentation Consistency* in more detail. To address the discrepancies in overlapping regions of tangent predictions, we propose to use a second *-augmented-* tangent image set and minimize the difference between the predictions of these pairs with an additional loss term. Following [2], we sampled $T = 18$ tangent images at four latitudes: $-67.5°, -22.5°, 22.5°, 67.5°$ for the original set. The tangent images are sampled for each latitude level with $90°$ apart in longitude. We extracted each tangent image with a resolution of 224×224 and field-of-view (FOV) of $80°$. We generated the augmented tangent image set under three configurations: (1) horizontally shifting viewports, (2) using a larger FOV for each viewport, and (3) varying the number, position, and FOV of the tangent viewports. It is important to emphasize that the augmented tangent images share weights with the original set, which neither requires extra parameters nor increases model complexity during training. Our experimental results in Fig. 4 demonstrate that each augmentation method improves model consistency significantly.

2.1 Approaches for Augmenting Tangent Images

Shifting Viewport Centers. In this setting, we keep the number and FOV of tangent images the same. We obtain the shifted tangent image set by applying a $45°$ horizontal shift on each viewport.

FOV Augmentation. In the second set, we keep the position of each tangent image the same and generate the augmented set by increasing their FOV to $120°$. Augmented FOV also provides the model with a multi-scale representation for the same input.

Viewport Augmentation. In the last setting, we generate the second set with $T' = 10$ tangent images with a FOV of $120°$, located in three latitudes: $-60°, 0°, 60°$. We sample $3, 4, 3$ viewports for each latitude.

2.2 Mask-weighted VAC Loss.

We use an optional weight for the proposed $\mathcal{L}_{VAC}(P,\ P')$ loss to increase consistency, especially on the overlapping regions on ERP. In Fig. 5, we provide the weight mask computed from the gnomonic projection for the original tangent image set. The performance comparison in Table 2 demonstrates the effectiveness of the proposed masking operation.

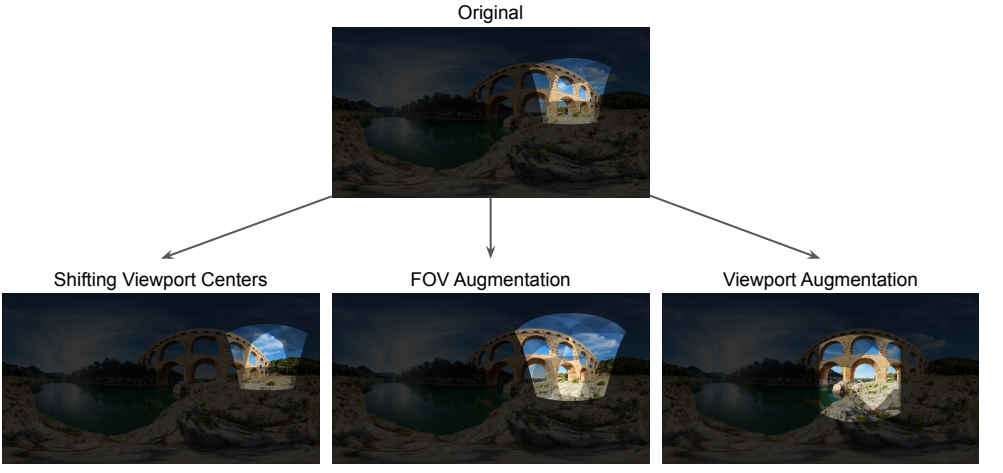

Figure 3: A tangent viewport from the original projection, highlighted on Equirectangular Projection (ERP) *(top)*, compared with three augmentation methods *(bottom)*.

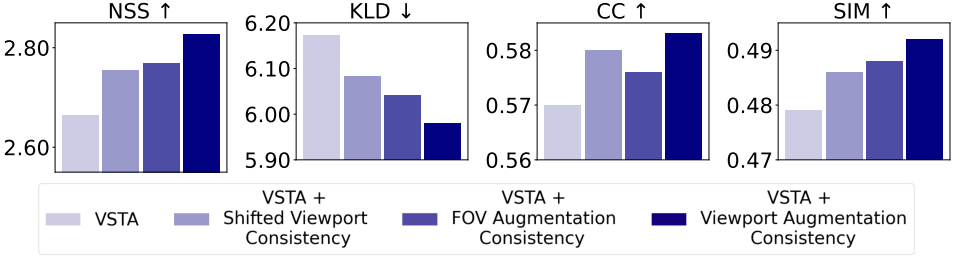

Figure 4: Performance of our VSTA baseline compared with three augmentation consistency methods on four metrics, on the validation split of VR-EyeTracking dataset.

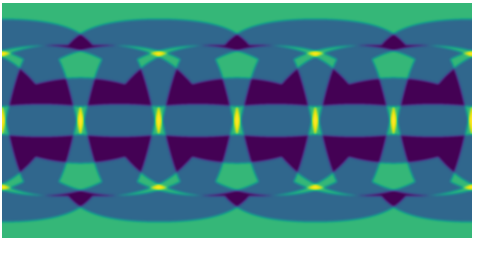

Figure 5: **Weight mask used for VAC Loss.** Each pixel coordinate in ERP takes a value based on the number of tangent viewports it is projected onto (max. 4). Brighter colours represent increasing overlaps.

Model	NSS↑	KLD↓	CC↑	SIM↑
VSTA + VAC *(w/o mask)*	2.624	6.011	0.576	0.490
VSTA + VAC *(w/ mask)*	2.630	5.744	0.586	0.492

Table 2: **Quantitative comparison** for the proposed VAC Loss with and without mask, on the validation split of VR-EyeTracking dataset.

3 Use Case: Saliency-Guided Omnidirectional Video Quality Assessment

In Table 3, we report the performance of two PSNR variants compared to ground-truth DMOS values in the VQA-ODV [3] dataset. Each row corresponds to saliency weights that supply human-perceptual information to PSNR and WS-PSNR metrics. The ground truth head movement (HM) maps refer to the viewports that human subjects have viewed while rating the visual quality of ODVs. Predicting saliency maps that better capture human head movements will result in better performance in the PSNR metrics. The table demonstrates that our proposed saliency prediction model better highlights the perceptually important regions in 360° videos compared to the state-of-the-art for omnidirectional video quality assessment downstream task.

Following the prior work [4], the saliency-weighted PSNR and WS-PSNR values are calculated as:

$$\mathrm{PSNR}_{sal} = 10\log_{10}\frac{Y_{max}^2 \cdot \sum_{p\in\mathbb{P}} w_{sal}(p)}{\sum_{p\in\mathbb{P}}(Y(p)-Y'(p))^2 \cdot w_{sal}(p)} \tag{1}$$

$$\mathrm{WS-PSNR}_{sal} = 10\log_{10}\frac{Y_{max}^2 \cdot \sum_{p\in\mathbb{P}} w_{sal}(p)\cos\theta_p}{\sum_{p\in\mathbb{P}}(Y(p)-Y'(p))^2 \cdot w_{sal}(p)\cos\theta_p} \tag{2}$$

where Y_{max} is the maximum intensity of the frames, $Y(p)$ and $Y'(p)$ denote the intensities for of pixel p in the reference and impaired videos, and θ_p is the latitude at pixel p.

	PSNR			WS-PSNR [5]		
Weight	PCC↑	SRCC↑	RMSE↓	PCC↑	SRCC↑	RMSE↓
None	0.650	6.664	7.502	0.671	0.686	7.233
PAVER [4]	0.661	0.667	7.481	0.679	0.691	6.914
Djilali et. al. [6]	0.648	0.721	7.336	0.684	0.721	6.829
SalViT360 (ours)	0.688	0.733	7.295	0.689	0.737	6.673
HM (Supervised)	0.764	0.759	6.601	0.759	0.756	6.612

Table 3: **Comparison with state-of-the-art** saliency models for *saliency-guided omnidirectional video quality assessment* on VQA-ODV [3] dataset, as a downstream task.

References

[1] Yanyu Xu, Yanbing Dong, Junru Wu, Zhengzhong Sun, Zhiru Shi, Jingyi Yu, and Shenghua Gao. Gaze prediction in dynamic 360 immersive videos. In *Proc. IEEE/CVF CVPR*, pages 5333–5342, 2018.

[2] Yuyan Li, Yuliang Guo, Zhixin Yan, Xinyu Huang, Ye Duan, and Liu Ren. Omnifusion: 360 monocular depth estimation via geometry-aware fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2801–2810, 2022.

[3] Chen Li, Mai Xu, Xinzhe Du, and Zulin Wang. Bridge the gap between vqa and human behavior on omnidirectional video: A large-scale dataset and a deep learning model. In *Proceedings of the 26th ACM International Conference on Multimedia*, MM '18, pages 932–940, New York, NY, USA, 2018. ACM.

[4] Heeseung Yun, Sehun Lee, and Gunhee Kim. Panoramic vision transformer for saliency detection in 360-degree videos. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXV*, pages 422–439. Springer, 2022.

[5] Yule Sun, Ang Lu, and Lu Yu. Weighted-to-spherically-uniform quality evaluation for omnidirectional video. *IEEE Signal Processing Letters*, 24:1408–1412, 2017.

[6] Yasser Abdelaziz Dahou Djilali, Tarun Krishna, Kevin McGuinness, and Noel E. O'Connor. Rethinking 360deg image visual attention modelling with unsupervised learning. In *Proc. IEEE/CVF ICCV*, pages 15414–15424, October 2021.