PANOWEAVER: GENERATIVE VIRTUAL CAMERA FOR PANORAMIC REFRAMING

Anonymous authors
Paper under double-blind review

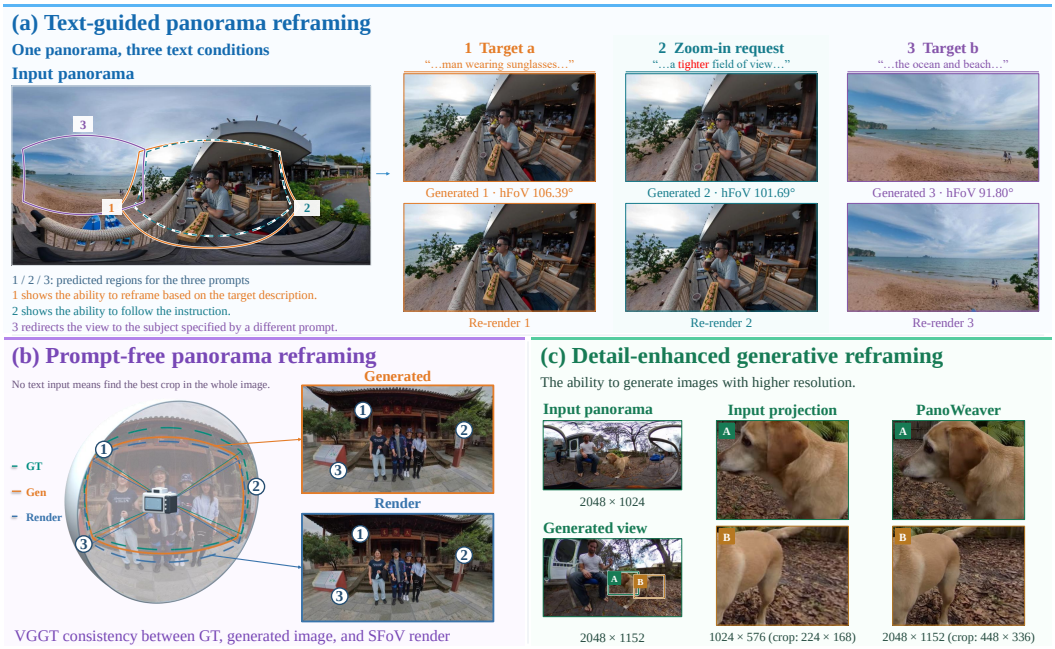

Figure 1: **Panorama reframing capabilities of PanoWeaver.** (a) shows text-guided subject selection and dewarping from the same panorama, with generated views compared against SFoV projections. The output follows the zoom-in prompt by changing its FoV, while another prompt redirects it toward a new subject. (b) shows automatic composition and dewarping without text input. (c) compares SFoV crops with high-resolution generations.

ABSTRACT

Panoramic cameras capture a complete field of view in a single shot, yet converting such omnidirectional observations into well-composed photographs remains challenging. To address this issue, we introduce panorama reframing, a new photography-oriented task that jointly determines *where to look* and *what the resulting perspective view should look like*. Then, we propose PanoWeaver, a coupled camera-image generation framework that treats the spherical field of view (SFoV) as a virtual camera action and the perspective image as its visual outcome. Built upon a diffusion transformer (DiT) backbone, PanoWeaver jointly generates both through bidirectional interaction, coupling explicit camera inference with generative visual modeling to learn camera–image correspondence. Moreover, we construct PanoComp-100K, a large-scale panoramic composition dataset containing 100,200 samples with SFoV states, perspective views, and human composition annotations. Extensive experiments demonstrate that PanoWeaver outperforms existing alternatives in panorama reframing, enabling a practical shoot-first and reframe-later paradigm for 360° photography. The source code, trained models, and dataset will be made available to the public.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

With advancements in image stitching, panoramic photography (Zhu et al., 2026) has become increasingly accessible, leading to a growing volume of user-generated omnidirectional content. By capturing a complete 360° field of view (FoV), it has enabled a wide range of applications, including virtual tours (Rahimizhian et al., 2020), immersive media (Amin et al., 2016; Makransky & Mayer, 2022), and virtual cinematography (Su & Grauman, 2017; Wang et al., 2021).

Although panoramas capture the complete FoV, their inherent view redundancy makes them difficult to directly use for photographic presentation. Perspective views, by selecting an appropriate viewpoint and field of view, can provide a more effective way to highlight salient subjects and achieve desired compositions. This raises a question: can a single-shot panorama enable the automatic generation of well-composed perspective views?

To address this issue, existing methods mainly focus on predicting pixel (Lai et al., 2018)- or box-level (Sawabe et al., 2022) salient regions in panoramas, without explicitly modeling the resulting perspective views and thus remaining unaware of their actual visual outcomes. Moreover, the salient regions considered by existing methods (Yun et al., 2022), including Think360 (Chen et al., 2026), are primarily defined from an attention- or action-oriented perspective, i.e., where one should look or act when situated in the scene, rather than from the perspective of photographic composition.

Motivated by this observation, we formulate this photography-oriented setting as panorama reframing, which aims to automatically identify a photographically meaningful region from a complete omnidirectional observation and render it into a perspective view. As shown in Figure 2, we represent the panoramic observation using equirectangular projection (ERP) and parameterize the target photographic view as a spherical field of view (SFoV). Furthermore, we can optionally provide a text prompt to specify the target subject, semantic region, or photographic intent, further enhancing the flexibility of panorama reframing.

To realize this formulation, we propose PanoWeaver, a coupled camera-and-image generation framework for reframing omnidirectional observations, inspired by world-action models (Yuan et al., 2026) in which predicted visual outcomes provide informative signals for action inference. Built upon a DiT backbone, PanoWeaver treats the SFoV as a virtual camera action and the perspective image as its visual outcome, jointly inferring both through bidirectional interaction. Unlike prior methods that either discriminatively predict the SFoV state or perform generative image-to-image translation, our model learns camera–image correspondence under shared panorama and text conditions.

Finally, we construct PanoComp-100K, a large-scale dataset of 100,200 samples for training and evaluation, each pairing an ERP panorama with a human-annotated SFoV state, its corresponding perspective view, and a view-specific caption. Extensive experiments on PanoComp-100K show that PanoWeaver outperforms existing alternatives on the panorama reframing task while jointly predicting camera states and perspective views.

The main contributions of our work are as follows:

- We introduce panorama reframing, a new photography-oriented task that automatically selects and renders well-composed perspective views from complete omnidirectional observations, enabling a shoot-first and reframe-later paradigm for 360° cameras.
- We propose PanoWeaver, a coupled camera-image generation framework that jointly predicts SFoV camera states and perspective images, bridging explicit camera inference with generative visual modeling.
- We construct PanoComp-100K, a large-scale panoramic composition dataset with 100,200 samples, and extensive experiments demonstrate the effectiveness of PanoWeaver for panorama reframing.

2 RELATED WORK

Automatic Composition Automatic composition aims to improve subject arrangement and visual aesthetics by adjusting image framing. Early methods rank candidate crops (Chen et al., 2017; Wei et al., 2018; Zeng et al., 2022) or predict crop coordinates (Hong et al., 2021; Jia et al., 2022). To in-

corporate user intent, conditioned cropping introduces textual descriptions or reference images (Horanyi et al., 2022; Zhong et al., 2023), while multimodal large language models enable interpretable cropping through instruction tuning and crop-rationale supervision (Sheng et al., 2025; Du et al., 2026), or iterative refinement using feedback on cropped results (Lee et al., 2025). However, these methods express aesthetic knowledge mainly through scoring or textual reasoning. Generative composition guidance further exploits pretrained image-generation priors by jointly producing target images and photographic instructions (You et al., 2026), but does not specify the corresponding viewing direction and field of view within an input panorama. Our framework uses the pretrained generator's compositional and aesthetic priors to guide SFoV inference, linking image generation with explicit panoramic framing.

Panoramic Viewpoint Selection Panoramic viewpoint selection identifies suitable views using predefined candidates (Berkiten et al., 2018; Su & Grauman, 2017), saliency cues (Djilali et al., 2021; Sawabe et al., 2022; Wang et al., 2021), semantic cues (Hu et al., 2017; Rao et al., 2022), and aesthetic assessment (Chen et al., 2025). However, these methods remain largely driven by scene content and learned preferences, with limited support for explicitly specifying the user's photographic intent. Language-guided methods further support object localization and navigation (Xu et al., 2026; Sheng et al., 2026). These approaches primarily determine where to look without jointly modeling the resulting perspective image. Our framework targets photographic reframing under optional text guidance, modeling the resulting perspective image as part of the view-selection process.

Joint Multimodal Diffusion Prior methods assign independent noise levels to different modalities for joint image–text modeling (Bao et al., 2023), joint modeling across text, images, and audio (Li et al., 2025b), and RGB–depth prediction (Duisterhof et al., 2026). These methods model correlations among perceptual modalities, whereas our model couples an explicit SFoV camera state with its perspective image as a decision–outcome pair. During joint denoising, image-space evidence informs SFoV inference, while the evolving camera state guides image synthesis.

3 PanoComp-100K Dataset

Building on the panorama reframing formulation in Section 1, training a model that jointly predicts the SFoV state and its perspective view requires aligned framing decisions, projected images, and view-specific descriptions, which existing datasets do not provide. We construct **PanoComp-100K**, a large-scale dataset containing 100,200 aligned examples with diverse scenes, viewpoints, and photographic compositions. More details can be found in Appendix B.1.

3.1 Data Formulation

Each example comprises a native ERP panorama P_i^{nat}, a human-preferred camera state $s_i^\star$, its projected image $I_i^\star$, and a caption c_i:

$$\mathcal{D} = \{(P_i^{\text{nat}}, s_i^\star, I_i^\star, c_i)\}_{i=1}^N, \qquad I_i^\star = \mathcal{R}(P_i^{\text{nat}}, s_i^\star), \tag{1}$$

where $N = 100{,}200$ and $\mathcal{R}$ is the spherical-to-planar projector. Unlike the horizontal and vertical location annotations in PCARD (Li et al., 2025a), our six-parameter SFoV state records direction, extent, projection geometry, and in-plane orientation:

$$s = (\psi, \vartheta, \alpha_h, \alpha_v, d, \omega). \tag{2}$$

Here ψ, ϑ are yaw and pitch, α_h, α_v are horizontal and vertical camera FoVs, and ω is roll. The dimensionless projection-shape parameter d controls the radial mapping through a virtual projection-center offset; it represents neither physical translation nor scene depth. Appendix C gives the FoV conversion and native-ERP sampling rule.

3.2 Collection and Annotation

We extract 300,000 ERP panoramas from YouTube 360° videos and retain those with native resolution above 1K. We further filter out badly exposed images, NSFW content, and images with poor visual quality. The final collection contains 100,000 training panoramas and 200 validation panoramas. For each panorama, human annotators select their preferred view, represented by the SFoV

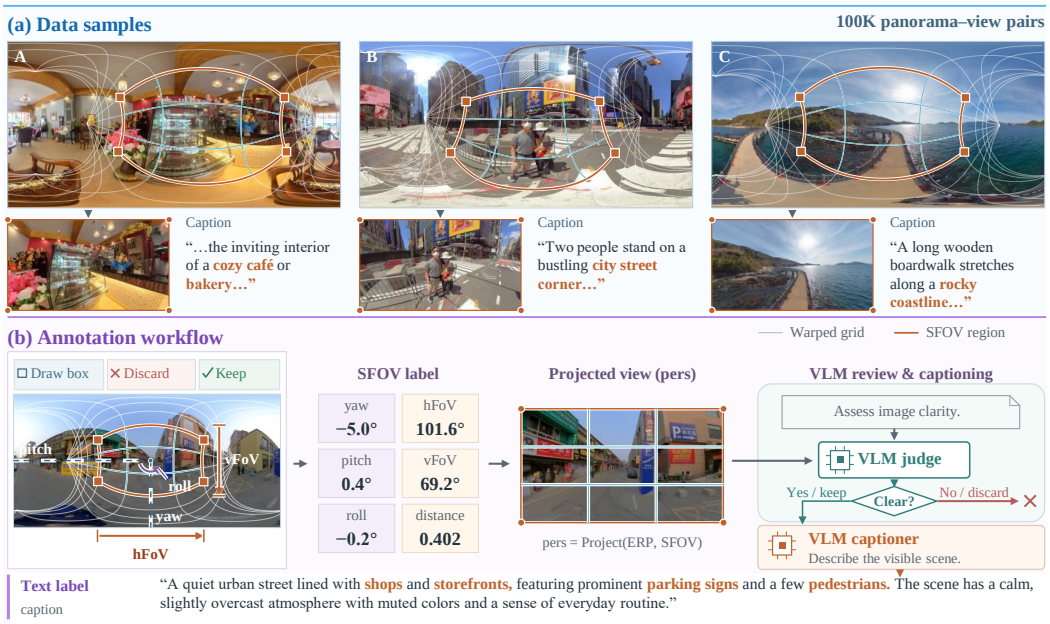

Figure 2: Overview of the PanoComp-100K dataset. (a) Representative panorama–perspective pairs with captions. Orange outlines mark the selected SFoV regions. (b) Annotation workflow. Human annotators select a preferred view in each ERP panorama and record its six-parameter SFoV state. The projected view is then filtered and captioned by VLM.

state $s_i^\star$. We project the native ERP using this state and caption the resulting view with Qwen3-VL-8B-Instruct (Bai et al., 2025). Captioning the selected view ties c_i to its visible content, yielding the aligned tuples in Equation 1.

4 Method

PanoWeaver models the joint distribution $p_\theta(I, s \mid P, c)$ of a perspective image I and camera state s, conditioned on a resized panorama P and optional text c, with model parameters θ. As shown in Figure 3, the camera interacts with the evolving image throughout denoising through a compact camera token, joint attention, and independent noise levels.

4.1 Joint Camera–Image Representation

We jointly represent a composition by its camera state and image, coupling the framing decision with its visual outcome. This allows the model to refine the selected view together with the image it produces. The physical SFoV state $s = (\psi, \vartheta, \alpha_h, \alpha_v, d, \omega)$ is mapped to a normalized representation $\mathbf{s}_0 = \mathcal{N}(s)$, while the target image is encoded as a latent z_0. The panorama latent z_P and optional text embedding h_c provide fixed context for both generation targets. Only the camera state and target image are noised and jointly denoised; the panorama remains clean throughout.

4.2 Coupled Denoising Transformer

As shown in Figure 3, we augment FLUX.2-klein-base-9B (Black Forest Labs, 2025) with a dedicated SFoV pathway that couples camera inference with image synthesis throughout denoising. Through repeated joint attention, the camera token reads panoramic context and the emerging image, while image tokens attend to the evolving camera state. Panorama and target-image tokens share the visual branch. More details are provided in Appendix D. To accommodate different camera and image noise levels, visual and text tokens are modulated by t_{img}, while the camera token uses t_{cam}. The panorama remains clean throughout this process. Separate output heads predict the

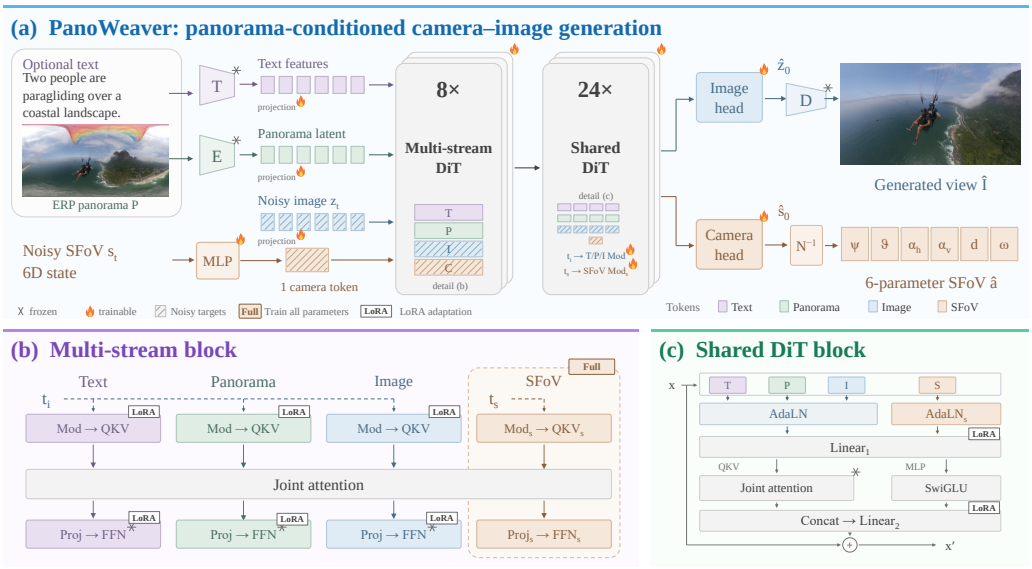

Figure 3: PanoWeaver architecture. (a) The clean panorama and optional text condition the joint denoising of the target image and SFoV camera state. A dedicated camera branch interacts with the text, panorama, and image tokens through joint attention. (b) Four token types use separate branches and participate in joint attention. (c) All tokens are processed with shared attention and feed-forward weights.

clean image latent and normalized camera state:

$$(\hat{z}_0, \hat{\mathbf{s}}_0) = F_\theta(z_P, h_c, z_{t_{\text{img}}}, \mathbf{s}_{t_{\text{cam}}}, t_{\text{img}}, t_{\text{cam}}). \tag{3}$$

4.3 Learning with Decoupled Noise Levels

We use independent noise levels so that a clearer image can guide camera inference, while a clearer camera state can guide image denoising. Following modality-specific diffusion (Bao et al., 2023; Zhu et al., 2025), we independently sample t_{img} and t_{cam}, together with independent standard Gaussian noises ϵ_{img} and ϵ_{cam}. The noisy targets follow straight interpolation paths (Lipman et al., 2022):

$$z_{t_{\text{img}}} = (1 - t_{\text{img}})z_0 + t_{\text{img}}\epsilon_{\text{img}}, \qquad \mathbf{s}_{t_{\text{cam}}} = (1 - t_{\text{cam}})\mathbf{s}_0 + t_{\text{cam}}\epsilon_{\text{cam}}, \tag{4}$$

where $t = 0$ denotes a clean target and $t = 1$ denotes noise. For training, we supervise image predictions in velocity space and camera predictions in normalized clean-state space. Written in terms of the clean predictions, the joint objective is

$$\mathcal{L} = \mathbb{E}\left[\frac{\text{MSE}(\hat{z}_0, z_0)}{\bar{t}_{\text{img}}^2} + \lambda_s \,\text{MSE}(\hat{\mathbf{s}}_0, \mathbf{s}_0)\right], \qquad \lambda_s = 10, \tag{5}$$

where $\bar{t}_{\text{img}} = \max(t_{\text{img}}, 10^{-4})$ and each MSE averages over its own target elements. The image weighting follows from converting clean predictions to velocities, as derived in Appendix D.2. Camera–image correspondence is learned through paired supervision and joint attention, without a differentiable reprojection constraint. At inference, both targets start from independent Gaussian noise and are jointly refined with 50 Euler steps on a shared decreasing time schedule. Each update uses the current states of both targets. Image decoding and inverse camera normalization recover $\hat{I}$ and $\hat{s}$. Time sampling and the exact Euler update are detailed in Appendix D.

4.4 Two-Stage Training

The first stage adapts a pretrained image generator to panorama-conditioned joint camera–image generation. We train for 30,000 steps on the 100,000 training tuples described in Section 3.1, learning the correspondence between panoramic context, human-preferred SFoV states, and their projected planar views. Under the joint denoising objective, image and camera states are independently

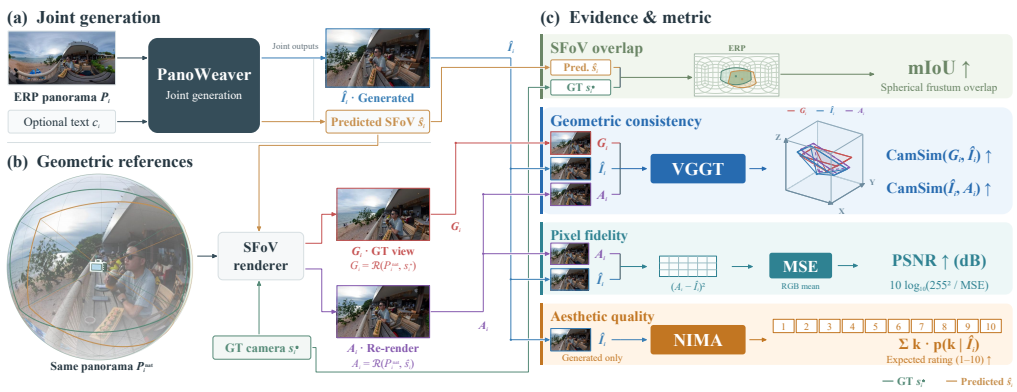

Figure 4: Evaluation pipeline. (a) Joint generation of the image and SFoV state. (b) Projection of the native panorama into the GT view and predicted-SFoV view. (c) Examples of evaluation metrics.

corrupted and predicted together, conditioned on the panorama and optional caption. In the second stage, we continue training for 3,000 updates with the panorama input fixed at 2048×1024 and target images at multiple resolutions, aiming to restore the backbone's native multi-resolution generation capability after adaptation to panorama reframing.

5 Experiments

We evaluate PanoWeaver against saliency, cropping, grounding, vision–language, and generative baselines on the fixed 200-panorama validation set. We report text-conditioned and text-free results, then analyze perspective-stream feedback and output resolution.

5.1 Implementation Details

As shown in Figure 3, the model uses 8 multi-stream blocks followed by 24 shared DiT blocks, with separate SFoV and image prediction heads and LoRA adaptation. Both training stages use LoRA rank 16. Base training runs for 30,000 updates with a learning rate of 1×10^{-4}; the separate 3,000 step native multi-resolution continuation uses 5×10^{-5}. The quantitative studies use 50 denoising steps in joint sampling mode. More details can be found in Appendix D.

5.2 Evaluation Protocol

For a native panorama P_i^{nat}, annotated camera $s_i^\star$, and generated pair $(\hat{s}_i, \hat{I}_i)$, we construct

$$G_i = \mathcal{R}(P_i^{\text{nat}}, s_i^\star), \qquad A_i = \mathcal{R}(P_i^{\text{src}}, \hat{s}_i), \tag{6}$$

where $\mathcal{R}$ is the SFoV projector. G_i is the ground-truth (GT) projection rendered from the native panorama, and A_i is the saved predicted-SFoV projection. P_i^{src} denotes the panorama source recorded for that output, which may be the native ERP or its cached resized version; Appendix E.1 specifies these source differences.

As shown in Figure 4, we evaluate framing, image fidelity, and semantic consistency separately. The mean intersection over union (mIoU) measures overlap between predicted and annotated SFoV regions. NIMA evaluates the aesthetic quality of each method's output image. For text-conditioned evaluation, a VLM scores consistency between that output and the input text; it is not used as a defect or aesthetic score.

Because generated images do not expose an explicit camera state, their camera accuracy cannot be measured directly. We therefore use a frozen VGGT-1B (Wang et al., 2025) to compute a viewpoint-agreement score. For a projection row, we evaluate the predicted-SFoV projection A_i against the GT projection G_i using $[G_i, A_i]$; for a generation row, we evaluate the generated image $\hat{I}_i$ against G_i using $[G_i, \hat{I}_i]$. From the jointly estimated camera parameters, we use cosine similarity to measure the



Figure 5: Qualitative comparison of text-conditioned panorama composition.

Table 1: Text-conditioned composition on standard val200.

Method	Type	Venue/Year	mIoU ↑	VGGT ↑	SIFT ↓	PSNR ↑	NIMA ↑	VLM ↑
Pano-CLIP (Radford et al., 2021)	VLM	ICML '21	0.274	0.852	0.304	11.062	4.582	82.375
PhotoFramer (You et al., 2026)	Generation	CVPR '26	N/A	0.854	0.363	11.337	**5.561**	94.625
Grounding DINO (Liu et al., 2024)	Grounding	ECCV '24	0.330	0.883	0.341	11.130	4.835	81.750
EAGLE-360 (Xu et al., 2026)	Grounding	arXiv '26	0.478	0.931	0.181	11.927	4.899	92.250
Venus* (Du et al., 2026)	VLM	CVPR '26	0.540	0.952	0.126	12.500	4.870	95.000
Thinking in 360°* (Yu et al., 2026)	VLM	CVPR '26	<u>0.626</u>	0.953	0.122	12.941	5.150	**97.875**
GPT-5.5	VLM	API '26	0.530	0.951	0.132	12.338	4.913	96.750
Gemini 3.1 Pro	VLM	API '26	0.534	0.951	0.134	12.234	4.890	96.875
Ours (SFoV)	Generation	This work	**0.672**	<u>0.965</u>	**0.104**	<u>13.258</u>	4.991	<u>97.625</u>
Ours (Generated Image)	Generation	This work	N/A	**0.972**	<u>0.111</u>	**13.651**	<u>5.158</u>	97.125

Best available in **bold**; second-best <u>underlined</u>. * denotes adaptation on our dataset. ↑: higher is better; ↓: lower is better.

agreement of virtual cameras. Details are provided in Appendix E. SIFT measures normalized displacement of geometrically verified feature matches on the same resized images. For fast inference, the text-conditioned consistency evaluator is Qwen3-VL-Flash.

5.3 Main Results

We compare our method with grounding, saliency, cropping, VLM, and generative baselines on the fixed 200-panorama validation set, under both text-conditioned and text-free settings. Tables 1 and 2 report the quantitative results for the two settings, respectively. Figures 5 and 6 show the corresponding qualitative comparisons. As shown in Figure 4, image-level metrics evaluate each method's final output O_i, generated images from our method and PhotoFramer, and camera-projected views for camera-selection baselines.

Text-conditioned composition. Table 1 shows that our method achieves more accurate framing than retrieval, grounding, and VLM-based approaches while maintaining competitive text–image consistency. Our camera projections exhibit lower SIFT displacement, while the generated images achieve higher PSNR and competitive aesthetic quality compared with camera-selection methods. Compared with the generative baseline, our generated images show closer agreement with the reference in both PSNR and SIFT displacement, although their aesthetic scores are lower. The qualitative comparisons in Figure 5 illustrate the resulting compositions: our method includes multiple subjects in the bedroom view and keeps the person and surrounding guests within the banquet frame.

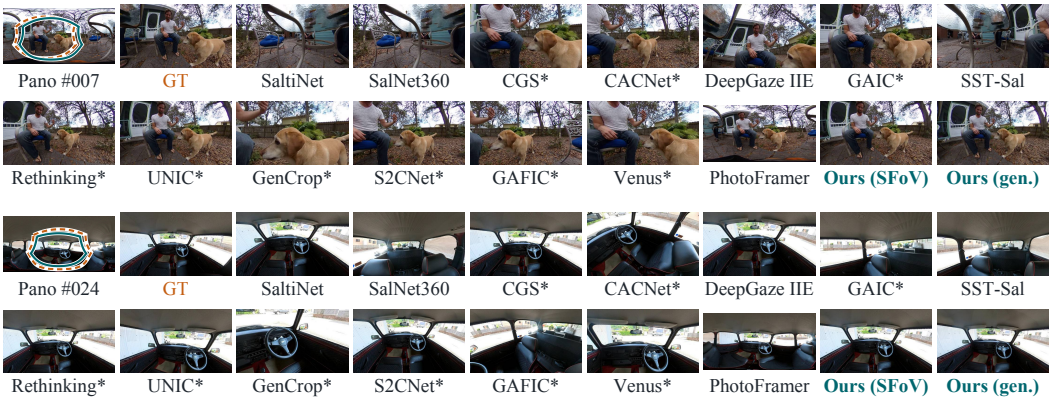

Figure 6: Qualitative comparison of text-free panorama composition.

Table 2: Text-free composition on standard val200.

Method	Type	Venue/Year	mIoU ↑	VGGT ↑	SIFT ↓	PSNR ↑	NIMA ↑
SaltiNet (Assens Reina et al., 2017)	Saliency	ICCVW '17	0.304	0.829	0.390	11.309	5.172
SalNet360 (Monroy et al., 2018)	Saliency	SPIC '18	0.318	0.836	0.400	11.353	5.205
CGS* (Li et al., 2020)	Cropping	CVPR '20	0.383	0.870	0.324	11.463	5.166
CACNet* (Hong et al., 2021)	Cropping	CVPR '21	0.377	0.872	0.320	11.257	5.170
DeepGaze IIE (Linardos et al., 2021)	Saliency	ICCV '21	0.396	0.868	0.366	11.600	5.152
GAIC* (Zeng et al., 2022)	Cropping	TPAMI '22	0.346	0.846	0.368	11.310	5.158
SST-Sal (Bernal-Berdun et al., 2022)	Saliency	C&G '22	0.309	0.841	0.399	11.166	5.156
Rethinking* (Jia et al., 2022)	Cropping	CVPR '22	0.449	0.887	0.295	12.025	5.144
UNIC* (Liu et al., 2023)	Cropping	ICCV '23	**0.468**	0.895	0.302	12.131	5.050
GenCrop* (Hong et al., 2024)	Cropping	AAAI '24	0.227	0.836	0.446	10.536	4.712
S2CNet* (Su et al., 2024)	Cropping	AAAI '24	0.392	0.863	0.334	11.541	5.175
GAFIC* (Yang et al., 2026)	Cropping	arXiv '26	0.384	0.853	0.343	11.624	5.208
Venus* (Du et al., 2026)	VLM	CVPR '26	0.414	**0.900**	0.290	11.788	4.890
PhotoFramer (You et al., 2026)	Generation	CVPR '26	N/A	0.826	0.310	11.074	**5.636**
Ours (SFoV)	Generation	This work	0.465	0.885	**0.288**	12.143	5.122
Ours (Generated Image)	Generation	This work	N/A	0.892	0.352	**12.413**	5.218

Best available in **bold**; second-best underlined. * denotes adaptation on our dataset. ↑: higher is better; ↓: lower is better.

Text-free composition. To assess the ability to find the best crop in the panorama reframing task, we remove the text input without further adaptation. Table 2 shows that our method achieves framing accuracy comparable to the strongest cropping approaches and better than saliency-based selection. Our camera projections obtain the lowest SIFT displacement, while the generated images achieve the highest PSNR and competitive aesthetic scores. These strengths differ across output types: several cropping and VLM-based approaches have lower SIFT displacement than our generated images, and the generative baseline retains an advantage in aesthetic scores. Figure 6 illustrates our method's framing preferences, retaining both the seated person and the dog and including the steering wheel and dashboard in the car example. Since multiple compositions can be valid without text guidance, agreement with the annotated view captures only one aspect of photographic quality.

5.4 Ablation Studies

We study perspective-stream feedback and multi-resolution training that are central to PanoWeaver. The former determines how image-space evidence influences camera prediction, while the latter determines the range of output resolutions supported by the model.

Effect of perspective-stream feedback. To examine whether the generated perspective image provides useful evidence for camera prediction, we remove the feedback path from the perspective stream at inference time. Specifically, text, ERP, and SFoV queries are prevented from attending to the keys and values of the perspective stream in all multi-stream and shared DiT blocks, while the perspective queries remain unchanged. The model is evaluated without retraining, so this intervention isolates the effect of the learned feedback mechanism.

Table 3: Ablation of perspective-stream feedback and multi-resolution training. A checkmark indicates that the component is enabled and a cross indicates that it is disabled.

Variant	Feedback	Multi-res.	Output	mIoU ↑	VGGT ↑	SIFT ↓	PSNR ↑	NIMA ↑	VLM ↑
Full model	✓	✓	Projection	0.672	0.965	0.104	13.258	4.991	97.625
Full model	✓	✓	Generated	N/A	0.972	0.111	13.651	5.158	97.125
w/o multi-resolution	✓	×	Projection	0.683	0.965	0.101	13.404	5.027	97.750
w/o multi-resolution	✓	×	Generated	N/A	0.967	0.118	13.713	5.213	97.750
w/o perspective feedback	×	✓	Projection	0.264	0.812	0.417	11.166	5.086	76.750

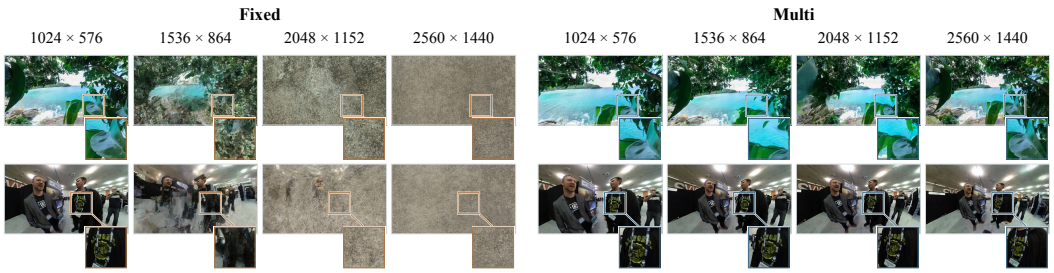

Figure 7: Qualitative comparison across output resolutions. Fixed and Multi denote fixed-resolution and multi-resolution training, respectively. Each column corresponds to one of the four evaluated output scales.

As shown in Table 3, removing perspective-stream feedback substantially degrades camera prediction. The mIoU drops, while the viewing-direction error increases. The quality of the corresponding perspective output also decreases, as reflected by the lower VGGT and VLM scores. These results indicate that the perspective stream provides informative visual evidence for refining the SFoV prediction, rather than serving only as a downstream rendering branch. We further visualize the joint denoising process in Appendix D.5, where the SFoV and perspective streams progressively become more consistent during generation.

Generation across output resolutions. We compare two models, one trained with multi-resolution data and one without it. Fixed uses $1\times$ training targets, while Multi uses all four resolution scales. We use the same inputs for both models. As shown in Figure 7, both models produce coherent scenes at $1\times$. At larger output sizes, Fixed develops severe structural and texture artifacts, whereas Multi preserves recognizable subjects and scene layout up to $2.5\times$.

6 Conclusions

We introduce panorama reframing, a new task of composing perspective views from panoramas, and propose PanoWeaver, a dual-stream diffusion model that jointly generates virtual camera states and perspective images. To train and evaluate PanoWeaver, we construct the PanoComp-100K dataset and introduce evaluation metrics, including a VGGT-based score for viewpoint consistency. Experiments demonstrate leading performance on several metrics, including mIoU and PSNR in the text-conditioned setting. Future work will explore personalized composition and temporally consistent reframing of panoramic videos.

7 AI Use Statement

In this work, generative AI tools were used to assist with software-code development and debugging, as well as the interpretation and analysis of experimental results. They were also used to improve the language, clarity, and presentation of the manuscript. We also used a generative model as a data captioner, as described in Section 3.2. The authors carefully reviewed all AI-assisted outputs and remain fully responsible for the accuracy, integrity, scientific claims, experimental results, and conclusions presented in this work.

REFERENCES

Ashfaq Amin, Diane Gromala, Xin Tong, and Chris Shaw. Immersion in cardboard vr compared to a traditional head-mounted display. In *International conference on virtual, augmented and mixed reality*, pp. 269–276. Springer, 2016. 2

Marc Assens Reina, Xavier Giro-i Nieto, Kevin McGuinness, and Noel E. O'Connor. SaltiNet: Scan-Path Prediction on 360 Degree Images Using Saliency Volumes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, pp. 2331–2338, 2017. doi: 10.1109/ICCVW.2017.275. URL https://openaccess.thecvf.com/content_ICCV_2017_workshops/w34/html/SaltiNet_marc.a.r95gmail.com_kevin.mcguinnessgmail.com_xavier.giroupc.edu_ICCV_2017_paper.html. 8

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. URL https://arxiv.org/abs/2511.21631. 4, 14

Fan Bao, Shen Nie, Kaiwen Xue, Chongxuan Li, Shi Pu, Yaole Wang, Gang Yue, Yue Cao, Hang Su, and Jun Zhu. One transformer fits all distributions in multi-modal diffusion at scale. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 1692–1717, 2023. 3, 5

Sema Berkiten, Rosália G. Schneider, and Jared M. Johnson. Automatic photo-from-panorama for google maps. In *ACM SIGGRAPH 2018 Talks*, pp. 72:1–72:2, 2018. doi: 10.1145/3214745.3214802. 3

Edurne Bernal-Berdun, Daniel Martin, Diego Gutierrez, and Belen Masia. SST-Sal: A spherical spatio-temporal approach for saliency prediction in 360° videos. *Computers & Graphics*, 106: 200–209, 2022. doi: 10.1016/j.cag.2022.06.002. URL https://www.sciencedirect.com/science/article/pii/S0097849322001042. 8

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. https://bfl.ai/blog/flux-2, 2025. 4

Chenglizhao Chen, Guangxiao Ma, Wenfeng Song, Shuai Li, Aimin Hao, and Hong Qin. Saliency-free and aesthetic-aware panoramic video navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3):2037–2054, 2025. doi: 10.1109/TPAMI.2024.3516874. 3

Mingrui Chen, Hexiong Yang, Haogeng Liu, Huaibo Huang, and Ran He. Think 360°: Evaluating the width-centric reasoning capability of MLLMs beyond depth. *arXiv preprint arXiv:2603.22689*, 2026. 2

Yi-Ling Chen, Jan Klopp, Min Sun, Shao-Yi Chien, and Kwan-Liu Ma. Learning to compose with professional photographs on the web. In *Proceedings of the ACM International Conference on Multimedia*, pp. 37–45, 2017. doi: 10.1145/3123266.3123274. 2

Yasser Abdelaziz Dahou Djilali, Tarun Krishna, Kevin McGuinness, and Noel E. O'Connor. Rethinking 360-degree image visual attention modelling with unsupervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15414–15424, 2021. 3

Tianxiang Du, Hulingxiao He, and Yuxin Peng. Venus: Benchmarking and empowering multimodal large language models for aesthetic guidance and cropping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 37766–37776, 2026. 3, 7, 8

Bardienus Pieter Duisterhof, Deva Ramanan, Jeffrey Ichnowski, Justin Johnson, and Keunhong Park. Modality forcing for scalable spatial generation. *arXiv preprint arXiv:2606.13676*, 2026. 3

Chaoyi Hong, Shuaiyuan Du, Ke Xian, Hao Lu, Zhiguo Cao, and Weicai Zhong. Composing photos like a photographer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7057–7066, 2021. 2, 8

James Hong, Lu Yuan, Michaël Gharbi, Matthew Fisher, and Kayvon Fatahalian. Learning subject-aware cropping by outpainting professional photos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 2175–2183, 2024. doi: 10.1609/aaai.v38i3.27990. 8

Nora Horanyi, Kedi Xia, Kwang Moo Yi, Abhishake Kumar Bojja, Aleš Leonardis, and Hyung Jin Chang. Repurposing existing deep networks for caption and aesthetic-guided image cropping. *Pattern Recognition*, 126:108485, 2022. doi: 10.1016/j.patcog.2021.108485. 3

Hou-Ning Hu, Yen-Chen Lin, Ming-Yu Liu, Hsien-Tzu Cheng, Yung-Ju Chang, and Min Sun. Deep 360 pilot: Learning a deep agent for piloting through 360-degree sports videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3451–3460, 2017. 3

Gengyun Jia, Huaibo Huang, Chaoyou Fu, and Ran He. Rethinking image cropping: Exploring diverse compositions from global views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2446–2455, 2022. 2, 8

Wei-Sheng Lai, Yujia Huang, Neel Joshi, Christopher Buehler, Ming-Hsuan Yang, and Sing Bing Kang. Semantic-Driven Generation of Hyperlapse from 360 Degree Video. *IEEE Transactions on Visualization and Computer Graphics*, 24(9):2610–2621, sep 2018. ISSN 1077-2626, 1941-0506, 2160-9306. doi: 10.1109/TVCG.2017.2750671. URL https://ieeexplore.ieee.org/document/8031049/. 2

Seung Hyun Lee, Jijun Jiang, Yiran Xu, Zhuofang Li, Junjie Ke, Yinxiao Li, Junfeng He, Steven Hickson, Katie Datsenko, Sangpil Kim, Ming-Hsuan Yang, Irfan Essa, and Feng Yang. Cropper: Vision-language model for image cropping through in-context learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 30010–30019, 2025. 3

Debang Li, Junge Zhang, Kaiqi Huang, and Ming-Hsuan Yang. Composing Good Shots by Exploiting Mutual Relations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4213–4222, June 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Li_Composing_Good_Shots_by_Exploiting_Mutual_Relations_CVPR_2020_paper.html. 8

Jiawan Li, Fei Zhou, Zhipeng Zhong, Jiongzhi Lin, and Guoping Qiu. Towards smart point-and-shoot photography. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28242–28251, June 2025a. 3

Shufan Li, Konstantinos Kallidromitis, Akash Gokul, Zichun Liao, Yusuke Kato, Kazuki Kozuka, and Aditya Grover. OmniFlow: Any-to-any generation with multi-modal rectified flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13178–13188, 2025b. 3

Akis Linardos, Matthias Kümmerer, Ori Press, and Matthias Bethge. DeepGaze IIE: Calibrated Prediction in and Out-of-Domain for State-of-the-Art Saliency Modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12919–12928, 2021. doi: 10.1109/ICCV48922.2021.01268. URL https://openaccess.thecvf.com/content/ICCV2021/html/Linardos_DeepGaze_IIE_Calibrated_Prediction_in_and_Out-of-Domain_for_State-of-the-Art_Saliency_ICCV_2021_paper.html. 8

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. URL https://arxiv.org/abs/2210.02747. 5

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with Grounded Pre-training for Open-Set Object Detection. In *Computer Vision – ECCV 2024*, pp. 38–55. Springer Nature Switzerland, 2024. doi: 10.1007/978-3-031-72970-6_3. URL https://doi.org/10.1007/978-3-031-72970-6_3. 7

Xiaoyu Liu, Ming Liu, Junyi Li, Shuai Liu, Xiaotao Wang, Lei Lei, and Wangmeng Zuo. Beyond image borders: Learning feature extrapolation for unbounded image composition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13023–13032, 2023. 8

Guido Makransky and Richard E. Mayer. Benefits of taking a virtual field trip in immersive virtual reality: Evidence for the immersion principle in multimedia learning. *Educational Psychology Review*, 34:1771–1798, 2022. doi: 10.1007/s10648-022-09675-4. 2

Rafael Monroy, Sebastian Lutz, Tejo Chalasani, and Aljosa Smolic. SalNet360: Saliency maps for omni-directional images with CNN. *Signal Processing: Image Communication*, 69:26–34, 2018. doi: 10.1016/j.image.2018.05.005. URL https://doi.org/10.1016/j.image.2018.05.005. 8

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html. 7

Sima Rahimizhian, Ali Ozturen, and Mustafa Ilkan. Emerging realm of 360-degree technology to promote tourism destination. *Technology in Society*, 63:101411, 2020. 2

Anyi Rao, Linning Xu, and Dahua Lin. Shoot360: Normal view video creation from city panorama footage. In *ACM SIGGRAPH 2022 Conference Proceedings*, 2022. doi: 10.1145/3528233.3530702. 3

Yuuki Sawabe, Satoshi Ikehata, and Kiyoharu Aizawa. Saliency-based multiple region of interest detection from a single 360° image. *IEEE Access*, 10:89124–89133, 2022. 2, 3

Kai Sheng, Liuyi Wang, Haojie Dai, Jinlong Li, Yongrui Qin, Zongtao He, Chengju Liu, and Qijun Chen. P2DNav: Panorama-to-downview reasoning for zero-shot vision-and-language navigation. *arXiv preprint arXiv:2605.19634*, 2026. 3

Xiangfei Sheng, Pangu Xie, Weidong Zou, Pengfei Chen, Tong Zhu, and Leida Li. InstructCrop: Teaching multimodal large language models to crop aesthetic images. In *Proceedings of the ACM International Conference on Multimedia*, pp. 6830–6839, 2025. doi: 10.1145/3746027.3754931. 3

Yu-Chuan Su and Kristen Grauman. Making 360° video watchable in 2d: Learning videography for click free viewing. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1368–1376, 2017. doi: 10.1109/CVPR.2017.150. 2, 3

Yukun Su, Yiwen Cao, Jingliang Deng, Fengyun Rao, and Qingyao Wu. Spatial-Semantic Collaborative Cropping for User Generated Content. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(5):4988–4997, Mar 2024. doi: 10.1609/aaai.v38i5.28303. URL https://doi.org/10.1609/aaai.v38i5.28303. 8

Jianyi Wang, Mai Xu, Lai Jiang, and Yuhang Song. Attention-based deep reinforcement learning for virtual cinematography of 360-degree videos. *IEEE Transactions on Multimedia*, 23:3227–3238, 2021. doi: 10.1109/TMM.2020.3021984. 2, 3

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5294–5306, 2025. 6

Zijun Wei, Jianming Zhang, Xiaohui Shen, Zhe Lin, Radomir Mech, Minh Hoai, and Dimitris Samaras. Good view hunting: Learning photo composition from dense view pairs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5437–5446, 2018. doi: 10.1109/CVPR.2018.00570. 2

Jingtao Xu, Zizhuo Lin, Jianwen Sun, Yi Yang, and Yawei Luo. EAGLE-360: Embodied active global-to-local exploration in 360°. *arXiv preprint arXiv:2607.02479*, 2026. 3, 7

Haotian Yang, Zhile Yang, Kin-Man Lam, Patrick Le Callet, and Xin Sun. Global Attention-Fused Image Cropping with Attention-Guided and Global-Aligned Crop Evaluator, 2026. URL https://arxiv.org/abs/2608.04821. 8

Zhiyuan You, Ke Wang, He Zhang, Xin Cai, Jinjin Gu, Tianfan Xue, Chao Dong, and Zhoutong Zhang. PhotoFramer: Multi-modal image composition instruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10197–10207, 2026. 3, 7, 8

Heyang Yu, Yinan Han, Xiangyu Zhang, Baiqiao Yin, Bowen Chang, Xiangyu Han, Xinhao Liu, Jing Zhang, Marco Pavone, Chen Feng, Saining Xie, and Yiming Li. Thinking in 360°: Humanoid Visual Search in the Wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22445–22455, June 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Yu_Thinking_in_360deg_Humanoid_Visual_Search_in_the_Wild_CVPR_2026_paper.html. 7

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026. 2

Heeseung Yun, Sehun Lee, and Gunhee Kim. Panoramic vision transformer for saliency detection in 360° videos. In *European Conference on Computer Vision*, pp. 422–439. Springer, 2022. 2

Hui Zeng, Lida Li, Zisheng Cao, and Lei Zhang. Grid anchor based image cropping: A new benchmark and an efficient model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1304–1319, mar 2022. doi: 10.1109/TPAMI.2020.3024207. URL https://arxiv.org/abs/1909.08989. 2, 8

Zhihang Zhong, Mingxi Cheng, Zhirong Wu, Yuhui Yuan, Yinqiang Zheng, Ji Li, Han Hu, Stephen Lin, Yoichi Sato, and Imari Sato. ClipCrop: Conditioned cropping driven by vision-language model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 294–304, 2023. 3

Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. *arXiv preprint arXiv:2504.02792*, 2025. URL https://arxiv.org/abs/2504.02792. 5

Zhengdong Zhu, Weiyi Xue, Zuyuan Yang, Wenlve Zhou, and Zhiheng Zhou. Pano360: Perspective to panoramic vision with geometric consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7600–7609, June 2026. 2

A Supplementary Material

In this section, we offer more details regarding our method, dataset and results. Appendix B describes dataset construction and the native multi-resolution training set. Appendix C specifies the SFoV geometry and panorama projection. Appendix D documents the model, training, and sampling procedure. Appendix E defines the experimental protocols and metrics, and Appendix F examines joint denoising and additional qualitative comparisons. For reference, Table 4 summarizes the core notation.

Table 4: Core notation. Geometry and evaluator-specific variables are defined locally in the corresponding sections.

Symbol	Meaning
P_i^{nat}, P_i	Native ERP and resized model-conditioning panorama.
$s_i^\star, \hat{s}_i$	Annotated and predicted six-field SFoV states.
$\mathcal{R}(P, s)$	Panorama projection under state s.
$z_0, \mathbf{s}_0$	Clean image latent and normalized seven-coordinate camera.
$t_{\mathrm{img}}, t_{\mathrm{cam}}$	Image and camera noise levels.
$G_i, A_i, \hat{I}_i$	GT projection, saved predicted-camera projection, and generated image.
P_i^{src}, O_i	Panorama source for A_i, and the output named by an evaluation row.

B Dataset Construction and Statistics

This section describes the construction of PanoComp-100K, its annotations and split, and the native multi-resolution training set used for the second training stage. Each example associates a native ERP panorama with an annotated SFoV state, the corresponding projected view, and a caption of that view, as defined in Equation 1.

B.1 Collection and Split

We extract 300,000 ERP panoramas from YouTube 360° videos and filter images with poor exposure, NSFW content, or poor visual quality. The resulting PanoComp-100K dataset contains 100,200 examples: 100,000 for training and a fixed validation set of 200 panoramas. For each selected panorama, it retains the highest-scored eligible 16:9 view together with its camera state and caption. Evaluation preserves the sample order and annotations. We offer a dataset gallery in Figure 8.

B.2 View Annotation

For each panorama, a human annotator selects a preferred photographic view and records its six-field SFoV state $s_i^\star$. Rendering this state from the native ERP produces the paired image $I_i^\star$. Appendix C specifies the projection geometry and sampling convention.

Human annotators provide the camera/composition annotations, whereas Qwen3-VL-8B-Instruct automatically generates view-specific captions from the selected projected views. The view is rendered at a width of 640 pixels before captioning. Qwen3-VL-8B-Instruct (Bai et al., 2025) uses greedy decoding with a maximum of 128 new tokens and the following prompt:

> Describe this image in English using exactly two sentences. The first sentence should describe the main subject and scene. The second sentence should describe details, style, or mood. Do not use numbering, bullet points, or any extra explanation.

The pipeline does not enforce the requested sentence count. Because the captioner observes the selected view, c_i describes the content visible in that view. These captions are semantic descriptions, not human-authored editing instructions.



Figure 8: Data gallery of PanoComp-100K.

B.3 Dataset Statistics

Figure 9 summarizes the scene coverage, source resolution, field of view, and caption length of the 100,000 PanoComp-100K training records. The most frequent caption-derived scene cues are indoor or enclosed spaces (28.9%), urban or street settings (24.4%), and mountainous or rocky terrain (13.3%). Overall, 83.0% of captions match at least one of the 14 scene-cue groups. These groups are non-exclusive, allowing a description to capture multiple aspects of a scene.

The source panoramas span 31 resolutions, all with a 2:1 aspect ratio; the most common resolution is 2160×1080, accounting for 49.8% of the training records. The annotated horizontal FoV has a median of $105.0°$, with the central 90% ranging from $90.8°$ to $118.9°$. View-specific captions contain 40.4 words on average, with a median of 40 words and a central 90% range of 32–50 words. Both central intervals span the 5th–95th percentiles.

Figure 10 illustrates the panorama, selected view, and caption associated with dataset examples.

B.4 Native Multi-Resolution Training Set

The second training stage uses 11,744 native multi-resolution training pairs, distributed across the four target sizes in Table 5. Each pair is assigned to one resolution tier according to the source-pixel support within its annotated view. This ties the target size to the available ERP detail in that view, rather than to the panorama dimensions alone.

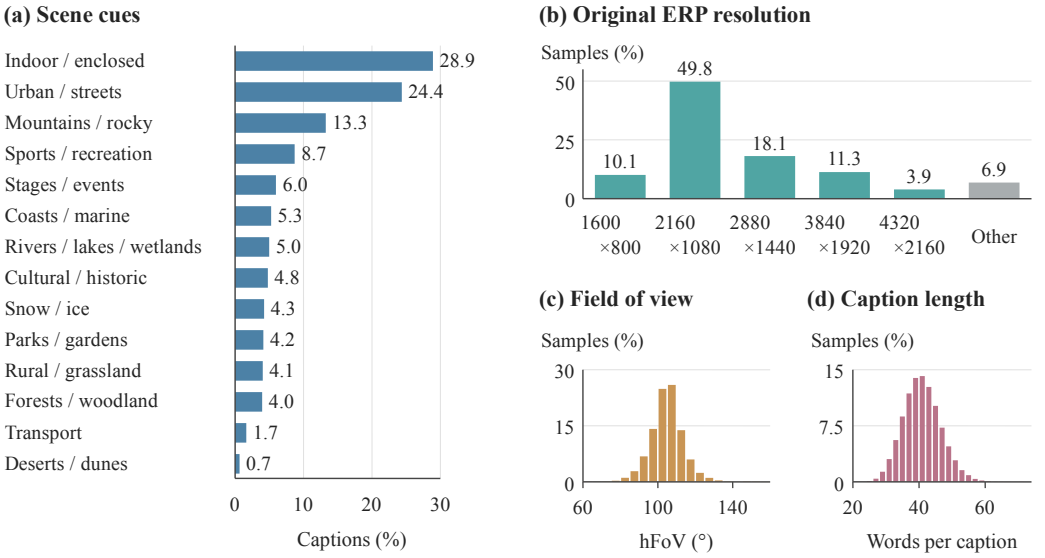

Figure 9: **PanoComp-100K training-set statistics.** All distributions are computed over the 100,000 training records. (a) Frequencies of 14 caption-derived scene-cue groups. Groups are obtained by keyword matching and can overlap; 17.0% of captions match none of the groups. (b) The five most frequent original ERP resolutions, shown as width × height in pixels; "Other" aggregates the remaining 26 resolutions. (c) Annotated horizontal camera FoV of the selected perspective views. (d) View-specific caption length in words. Histogram heights indicate the percentage of training records per bin, using bin widths of 5° in (c) and two words in (d).



Figure 10: Examples from PanoComp-100K, pairing ERP panoramas with human-preferred SFoV projections and view-specific captions.

Table 5: Native multi-resolution training pairs. Dimensions are width × height; each pair is assigned to one tier.

Scale	Target resolution	Training pairs
1.0	1024×576	2,870
1.5	1536×864	6,390
2.0	2048×1152	1,976
2.5	2560×1440	508
Total		11,744

Resolution assignment. At the reference output size $W_0 \times H_0 = 1024 \times 576$, let $\pi_{s_i^\star}$ map output coordinates (x, y) to native ERP coordinates (u, v) using the projector in Appendix C. The local source-pixel area per target pixel is

$$\rho_i(x, y) = \left| \det \frac{\partial(u, v)}{\partial(x, y)} \right|. \tag{7}$$

Longitude is unwrapped when estimating local differences near the ERP seam. Let $\bar{\rho}_i$ and $\rho_{i,0.1}$ denote the mean and tenth percentile of the local support. A pair is eligible when

$$\sqrt{\bar{\rho}_i} \geq 1, \qquad \rho_{i,0.1} \geq 0.64. \tag{8}$$

Each eligible pair receives the largest supported scale:

$$k_i = \max \left\{ k \in \{1, 1.5, 2, 2.5\} : k \leq \sqrt{\bar{\rho}_i} \right\}, \tag{9}$$

giving a target size of $(1024k_i) \times (576k_i)$. Scaling both image dimensions by k_i multiplies the target pixel count by k_i^2, so the approximate mean source support per target pixel becomes $\bar{\rho}_i / k_i^2$. This rule does not ensure uniform native detail at every output location. Since support depends on the selected view, the resolution tiers can also differ in their FoV distributions.

Quality filtering. Available VLM judgments exclude views flagged for blur, watermarks, or black borders. Image-statistic checks additionally reject views with a near-black pixel fraction of at least 0.5 or grayscale Laplacian variance below 25. A missing or failed VLM judgment is not itself a rejection criterion: among the final 11,744 records, 11,621 have valid VLM judgments and 123 rely on the geometric and image-statistic checks. This screening applies to the native extension and does not constitute exhaustive VLM quality verification of the base collection.

C SFoV Geometry and Panorama Projection

The SFoV renderer maps each output pixel to a direction on the unit sphere, rotates that direction into the panorama coordinate frame, and samples the native ERP. This section specifies the camera fields, inverse projection, and sampling conventions used by $\mathcal{R}(P^{\mathrm{nat}}, s)$.

C.1 Camera Parameterization and Field of View

The camera state $s = (\psi, \vartheta, \alpha_h, \alpha_v, d, \omega)$ contains the six fields defined in Equation 2. Yaw ψ and pitch ϑ set the viewing direction, and roll ω sets the image-plane orientation. The annotated horizontal and vertical camera FoVs are α_h and α_v. The dimensionless projection-shape parameter d controls radial mapping through an auxiliary projection center offset from the center of a unit sphere; it represents neither physical translation nor scene depth. All angles below are in radians.

For output width W, height H, and aspect ratio $\rho_{\mathrm{asp}} = W/H = 16/9$, the renderer derives the effective horizontal camera FoV $\widetilde{\alpha}_h$ from α_v. Its corresponding angular coverage on the sphere is β_h:

$$\begin{aligned} \widetilde{\alpha}_h &= 2 \arctan\left(\rho_{\mathrm{asp}} \tan \frac{\alpha_v}{2} \right), \\ \beta_h &= \widetilde{\alpha}_h + 2 \arcsin\left(d \sin \frac{\widetilde{\alpha}_h}{2} \right). \end{aligned} \tag{10}$$

The stored α_h remains part of the six-field supervised state, while rendering uses $\widetilde{\alpha}_h$. Thus the stored horizontal FoV and the vertical FoV are not independent rendering controls.

To derive the spherical coverage, consider a horizontal section through the unit sphere. For camera-plane half-aperture $\zeta = \widetilde{\alpha}_h/2$ and spherical half-coverage $\chi = \beta_h/2$, the auxiliary-center geometry gives

$$\tan\zeta = \frac{\sin\chi}{\cos\chi + d}, \qquad \sin(\chi - \zeta) = d\sin\zeta. \tag{11}$$

The branch containing the central viewing direction yields $\chi = \zeta + \arcsin(d\sin\zeta)$, giving Equation 10. At $d = 0$, camera aperture and spherical coverage coincide. The implementation clips the $\arcsin$ argument to $[-1, 1]$ for numerical safety.

C.2 Inverse Projection onto the Sphere

For output pixel (x, y), we use zero-based raster indices with no half-pixel offset. Define the image-plane scale and coordinates as

$$\begin{aligned} \gamma &= \frac{2\sin(\beta_h/2)}{\cos(\beta_h/2) + d}, \\ \xi &= \gamma\left(\frac{x}{W} - \frac{1}{2}\right), \qquad \eta = \frac{\gamma}{\rho_{\text{asp}}}\left(\frac{y}{H} - \frac{1}{2}\right). \end{aligned} \tag{12}$$

The first, second, and third local coordinates are the forward, right, and up axes, respectively. Raster rows increase downward, so the local ray-plane direction is $(1, \xi, -\eta)^\top$.

We obtain the sphere direction by intersecting a ray from the auxiliary center $\mathbf{o}_d = (-d, 0, 0)^\top$ with the unit sphere. Writing $r_\perp^2 = \xi^2 + \eta^2$, the ray and unit-sphere constraint are

$$\begin{aligned} \mathbf{p}(\kappa) &= \mathbf{o}_d + \kappa(1, \xi, -\eta)^\top, \\ \|\mathbf{p}(\kappa)\|_2^2 = 1 \quad &\Longrightarrow \quad (1 + r_\perp^2)\kappa^2 - 2d\kappa + d^2 - 1 = 0. \end{aligned} \tag{13}$$

The forward-facing intersection is

$$\kappa = \frac{d + \sqrt{1 + (1 - d^2)r_\perp^2}}{1 + r_\perp^2}. \tag{14}$$

Dividing the sphere point by κ gives the proportional direction vector used by the implementation:

$$\mathbf{r}_s(x, y) = \begin{pmatrix} 1 - \dfrac{d(1 + r_\perp^2)}{d + \sqrt{1 + (1 - d^2)r_\perp^2}} \\ \xi \\ -\eta \end{pmatrix}. \tag{15}$$

Normalizing $\mathbf{r}_s$ gives the unit-sphere direction before camera rotation.

The endpoint settings clarify the role of d. At $d = 0$, the direction is $\mathbf{r}_s = (1, \xi, -\eta)^\top$, giving rectilinear projection. At $d = 1$, the sphere point becomes

$$\mathbf{p} = \frac{1}{1 + r_\perp^2}\begin{pmatrix} 1 - r_\perp^2 \\ 2\xi \\ -2\eta \end{pmatrix}, \tag{16}$$

which is inverse stereographic projection. Intermediate values vary the radial mapping continuously. All settings sample the observed panorama; changing d introduces neither translation parallax nor newly visible occluded content.

C.3 Rotation Convention and ERP Sampling

The source annotations follow the Insta360 coordinate convention. The local direction is rotated into the ERP coordinate frame by

$$R(s) = R_z(\pi - \psi)R_y(-\vartheta)R_x(\omega), \qquad \mathbf{n}_s(x, y) = \frac{R(s)\mathbf{r}_s(x, y)}{\|R(s)\mathbf{r}_s(x, y)\|_2}. \tag{17}$$

Here R_x, R_y, R_z are standard right-handed axis-rotation matrices. The substitutions $\pi - \psi$ and $-\vartheta$ implement the source annotation convention. At the image center, the resulting longitude is $(\pi - \psi) \bmod 2\pi$ and latitude is ϑ; roll changes the image-plane orientation without changing this central direction.

For $\mathbf{n}_s = (n_x, n_y, n_z)^\top$, the longitude and latitude are

$$\begin{aligned} \ell_s(x, y) &= \operatorname{atan2}(n_y, n_x) \bmod 2\pi, \\ \varphi_s(x, y) &= \operatorname{atan2}\left(n_z, \sqrt{n_x^2 + n_y^2}\right). \end{aligned} \tag{18}$$

For a native ERP of size $W_P \times H_P$, these angles map to

$$u_s = \frac{W_P \ell_s}{2\pi}, \qquad v_s = H_P \left(\frac{1}{2} - \frac{\varphi_s}{\pi}\right). \tag{19}$$

Longitude maps linearly across the ERP width, while latitude maps from the north pole to the south pole along its height. The projected target is

$$I_i^\star(x, y) = \mathcal{R}(P_i^{\text{nat}}, s_i^\star)(x, y) = \mathcal{B}[P_i^{\text{nat}}](u_{s_i^\star}, v_{s_i^\star}), \tag{20}$$

where $\mathcal{B}$ denotes bilinear remapping with longitude wrapping. Targets are sampled from the native ERP independently of the resized conditioning panorama P_i supplied to the model.

D Model, Training, and Inference Details

PanoWeaver learns to predict an SFoV state together with its corresponding perspective image. The camera and image tokens exchange information throughout denoising, allowing the framing decision and generated content to be refined together. This section explains how the camera encoding, joint attention, and independent training noise levels support this interaction, followed by the optimization settings and shared inference schedule. The native training set is described in Appendix B.4.

D.1 Camera Encoding and Joint Architecture

Camera encoding. The camera encoding makes framing an explicit denoising target that can interact with the image tokens. The six-field SFoV state s is mapped to seven network coordinates by encoding yaw as a sine/cosine pair. Yaw is expressed in radians when computing this pair; the remaining angular fields are converted to degrees for affine normalization, using $x_{\text{deg}} = 180x/\pi$:

$$\mathcal{N}(s) = \left(\sin\psi, \cos\psi, \frac{\vartheta_{\text{deg}}}{90}, \frac{\alpha_{h,\text{deg}} - 100}{60}, \frac{\alpha_{v,\text{deg}} - 100}{60}, 2d - 1, \frac{\omega_{\text{deg}}}{15}\right). \tag{21}$$

The sine/cosine encoding removes the discontinuity at the yaw wrap-around. The remaining fields use fixed affine normalization constants, which are neither learned parameters nor clipping bounds.

To recover an SFoV state from a predicted vector $\mathbf{q} \in \mathbb{R}^7$, define $r_\psi = \max(\sqrt{q_1^2 + q_2^2}, 10^{-6})$. Here q_j is the j-th component of $\mathbf{q}$. The inverse map, with angles returned in radians, is

$$\begin{aligned} \mathcal{N}^{-1}(\mathbf{q}) = \Bigg(&\operatorname{atan2}(q_1/r_\psi, q_2/r_\psi),\ \frac{\pi q_3}{2},\ \frac{\pi(60q_4 + 100)}{180}, \\ &\frac{\pi(60q_5 + 100)}{180},\ \frac{q_6 + 1}{2},\ \frac{\pi q_7}{12}\Bigg). \end{aligned} \tag{22}$$

Inverse normalization does not clip predictions.

Conditioning and token layout. Both prediction streams use the same panorama and text context throughout denoising. A frozen image autoencoder encodes the resized panorama and target view as z_P and z_0, respectively, and a frozen text encoder produces h_c. Only the target image and camera state are corrupted; keeping the panorama clean preserves access to the observed scene as the camera and image predictions evolve. A two-layer MLP maps the noisy camera state to one camera token. The resulting transformer input comprises four token types: text, clean panorama, noisy target image, and noisy camera.

Bidirectional interaction. Joint attention provides information paths in both directions: the camera token can read the emerging image content, while the image tokens can read the evolving framing state. Both also attend to the panorama and text context. The augmented FLUX.2-klein-base-9B backbone uses eight multi-stream blocks followed by 24 shared DiT blocks. Each multi-stream block has separate text, visual, and camera branches. Panorama and target-image tokens share the visual branch, so the four token types use three sets of branch parameters. Queries, keys, and values from all four token types enter joint attention, whose outputs are then routed to the respective branches. The shared DiT blocks process the concatenated sequence with shared weights, maintaining interaction between the camera and image tokens.

D.2 Noise Sampling and Training Objective

Training-time distribution. Independent noise levels vary which target supplies the clearer cue during training. When the image is less corrupted, its content provides clearer visual evidence for camera prediction; when the camera is less corrupted, its state provides clearer framing information for image denoising. When both are highly corrupted, the panorama and text provide the main available context. The image latent z_0 and normalized camera $\mathbf{s}_0$ are independently corrupted according to Equation 4. The same jointly attentive network learns from these different balances of evidence, without assigning a fixed prediction order to the camera and image.

We sample both times independently from the same shifted plateau-logit-normal distribution. To specify this distribution, first define the logit-normal density for a preliminary noise level $v \in (0, 1)$:

$$f(v) = \frac{\exp[-\operatorname{logit}(v)^2/2]}{\sqrt{2\pi}\, v(1-v)}, \qquad \operatorname{logit}(v) = \log \frac{v}{1-v}. \tag{23}$$

Let m be the mode of f, estimated on 4097 uniformly spaced points in $[10^{-5}, 1-10^{-5}]$. We draw v from a density proportional to $f(v)$ for $v \le m$ and to the constant $f(m)$ for $v > m$, then apply the shift and endpoint clipping:

$$\mathcal{T}(v) = \frac{e^{1.1} v}{1 + (e^{1.1} - 1)v}, \qquad t = \operatorname{clip}\left(\mathcal{T}(v), 10^{-4}, 1 - 10^{-4}\right). \tag{24}$$

Image-loss equivalence. The two outputs are supervised as the camera and image of the same annotated view, giving the interacting streams a common composition target. The image branch is supervised in velocity space, while the camera branch is supervised directly in normalized state space. For image time $t = t_{\text{img}}$, write $z_t = (1-t)z_0 + t\epsilon_{\text{img}}$. The target interpolation velocity is $v^\star = \epsilon_{\text{img}} - z_0$. For a clean prediction $\hat{z}_0$, the implied velocity and its error satisfy

$$\begin{aligned} \hat{v} &= \frac{z_t - \hat{z}_0}{t}, \\ \hat{v} - v^\star &= \frac{z_0 - \hat{z}_0}{t}, \\ \operatorname{MSE}(\hat{v}, v^\star) &= \frac{\operatorname{MSE}(\hat{z}_0, z_0)}{t^2}. \end{aligned} \tag{25}$$

Equation 25 explains the image weighting in the joint objective, Equation 5. Training uses $\bar{t}_{\text{img}} = \max(t_{\text{img}}, 10^{-4})$ and a camera loss weight $\lambda_s = 10$. Each MSE averages over its own target elements; the camera weight therefore balances the two mean losses rather than their raw element counts. The model learns camera–image correspondence through paired supervision and joint attention, without a differentiable reprojection constraint.

D.3 Training Configurations

Table 6 summarizes the two primary training stages. The base stage learns the correspondence between annotated framing decisions and their visual outcomes from 100,000 training tuples at a fixed target resolution. Native multi-resolution training then adapts the same joint model to a range of output sizes while keeping the panorama input size fixed. It starts from the resulting 30,000-update checkpoint and uses the four resolution tiers in Table 5. All dimensions below are width $\times$ height.

Table 6: Primary training configurations. Learning rates are nominal values before scheduling. The four native batch entries correspond to scales $1\times$, $1.5\times$, $2\times$, and $2.5\times$, respectively.

Setting	Base stage	Native continuation
Initialization	Pretrained backbone	Base at 30,000 updates
Training pairs	100,000	11,744
Optimizer updates	30,000	3,000
Panorama size	2048×1024	2048×1024
Target size	1024×576	Four native tiers
LoRA rank / scaling alpha	16 / 16	16 / 16
LoRA learning rate	10^{-4}	5×10^{-5}
Camera internal learning rate	10^{-5}	10^{-5}
Camera input/output learning rate	10^{-4}	10^{-4}
Effective batch	16	16
Caption dropout	0.1	0.1
Weight decay	0.01	0.01
Gradient clipping norm	1.0	1.0

Trainable parameters and optimization. We adapt the pretrained transformer's visual and text branches with LoRA and train the dedicated camera pathway in full, allowing both sides of the camera–image interaction to adapt. The inherited backbone weights remain frozen; LoRA is applied to the visual and text attention projections, single-stream linear layers, input projections, image timestep embedding, output head, and modulation layers. The dedicated camera input/output modules, timestep embedding, attention and feed-forward layers, and camera modulation layers are fully trainable. The image autoencoder and text encoder remain frozen.

Optimization uses AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and the parameter-group learning rates in Table 6. Training uses bfloat16 transformer computation and gradient checkpointing. To support text-free inference, we replace caption embeddings with the empty-prompt embedding with probability 0.1 during training.

D.4 Joint Sampling Procedure

Joint sampling carries the camera–image interaction through the denoising trajectory: each forward pass reads the current pair and predicts how both targets should be refined. The shared time schedule synchronizes their updates, while joint attention continues to make the evolving framing state and image content available to each other. At inference, image and camera states start from independent Gaussian noise and are refined jointly for $K = 50$ steps on the shared schedule

$$\tau_k = \mathcal{T}(1 - k/K), \qquad k = 0, \dots, K, \tag{26}$$

where $\mathcal{T}$ is defined in Equation 24. At each step, one forward evaluation predicts both clean targets from the current pair:

$$(\hat{z}_0^{(k)}, \hat{\mathbf{s}}_0^{(k)}) = F_\theta\Big(z_P, h_c, z^{(k)}, \mathbf{s}^{(k)}, \tau_k, \tau_k\Big). \tag{27}$$

For either target $u \in \{z, \mathbf{s}\}$, the implementation updates

$$u^{(k+1)} = u^{(k)} + (\tau_{k+1} - \tau_k)\frac{u^{(k)} - \hat{u}_0^{(k)}}{\max(\tau_k, 0.05)}. \tag{28}$$

Both clean estimates are computed before either state is updated. The denominator floor limits update amplification at small noise levels and is separate from the 10^{-4} training-loss floor. After the final update, the sampler returns the terminal states $z^{(K)}$ and $\mathbf{s}^{(K)}$. Decoding $z^{(K)}$ gives $\hat{I}$, and $\hat{s} = \mathcal{N}^{-1}(\mathbf{s}^{(K)})$ gives the SFoV prediction. The clean-head predictions $\hat{z}_0^{(k)}$ and $\hat{\mathbf{s}}_0^{(k)}$ are intermediate estimates along the trajectory and are not substituted for the terminal states. Algorithm 1 summarizes the update order.

Algorithm 1 Joint camera–image sampling

Require: Fixed panorama latent z_P, text embedding h_c, $K = 50$
1: Draw independent standard Gaussian states $z^{(0)}$ and $\mathbf{s}^{(0)}$
2: Set $\tau_k = \mathcal{T}(1 - k/K)$ for $k = 0, \ldots, K$
3: **for** $k = 0, \ldots, K-1$ **do**
4: Predict $(\hat{z}_0^{(k)}, \hat{\mathbf{s}}_0^{(k)})$ from the current pair using Equation 3 at times (τ_k, τ_k)
5: Update both targets with Equation 28, using the same pair of predictions
6: **end for**
7: Decode $z^{(K)}$ into $\hat{I}$ and compute $\hat{s} = \mathcal{N}^{-1}(\mathbf{s}^{(K)})$
8: **return** $(\hat{I}, \hat{s})$

D.5 Joint Denoising Visualization

Figure 11 illustrates the consistency between SFoV and perspective-image predictions during joint denoising. Across the warehouse, penguin, and table-tennis examples, changes in viewing direction and FoV are accompanied by changes in subject placement and scene coverage in the generated images. The two outputs are refined together: as the SFoV prediction adjusts the framing, the image evolves toward a corresponding composition.

Joint attention allows the two prediction streams to inform one another throughout this process. The emerging image provides subject and layout cues for SFoV refinement, while the evolving SFoV representation provides viewpoint and framing cues for image synthesis. By exchanging this information at each step, camera selection and perspective synthesis can refine a shared composition throughout denoising.

E Experimental Protocols and Metrics

The evaluation distinguishes camera states, images projected from those states, and images generated by the model. We first define the output associated with each table row, then group the metrics by the property they measure. Table 7 summarizes their inputs.

E.1 Evaluation Setup and Output Definitions

Baselines marked with an asterisk were adapted on the PanoComp-100K training split for three epochs. We evaluated these adapted baselines on the same fixed 200-panorama validation set as the other methods. All full-set comparisons use the same $N_{\text{val}} = 200$ validation panoramas, in the fixed order and with their original camera annotations and captions. For native panorama P_i^{nat}, annotated state $s_i^\star$, and predicted state $\hat{s}_i$, define

$$G_i = \mathcal{R}(P_i^{\text{nat}}, s_i^\star), \qquad A_i = \mathcal{R}(P_i^{\text{src}}, \hat{s}_i), \tag{29}$$

where P_i^{src} is the panorama source used to produce the saved camera projection for that run. The generated image $\hat{I}_i$ is decoded from the image latent and is distinct from A_i.

The row-specific output is $O_i = A_i$ for camera-selection baselines and Ours (SFoV), and $O_i = \hat{I}_i$ for PhotoFramer and Ours (Generated Image). The two Ours rows distinguish camera-projected and generated outputs of the framework. We report camera mIoU on the SFoV row and mark it N/A for the generated-image row. VGGT uses the predicted-camera projection A_i, independently of the image-metric output O_i.

E.2 Camera and Viewpoint Metrics

Spherical viewport mIoU. We compare camera footprints on the sphere rather than measuring overlap in ERP pixels, whose area per pixel varies with latitude. Let $\mathcal{U}_M = \{\mathbf{u}_j\}_{j=0}^{M-1}$ be the fixed

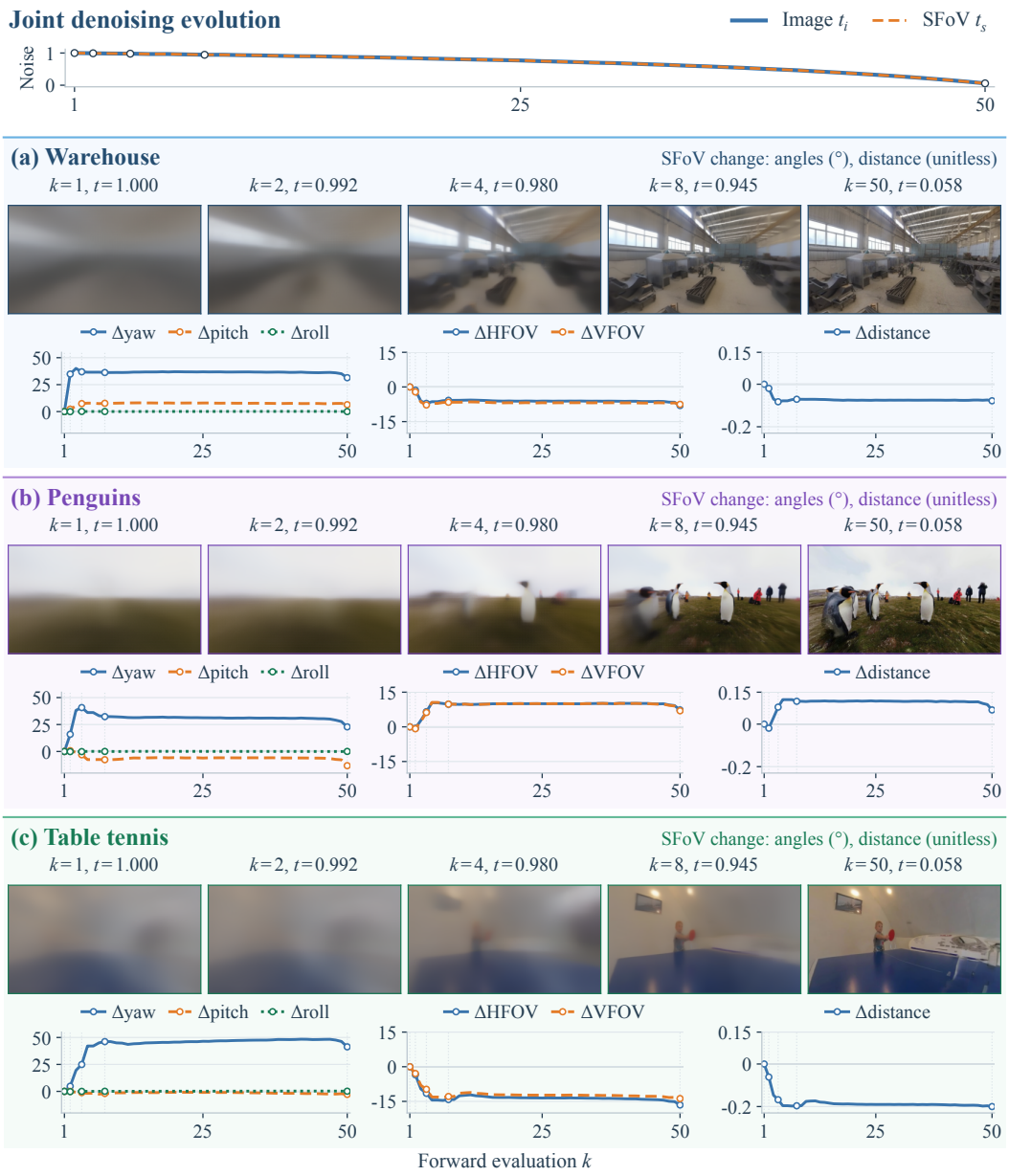

Figure 11: Joint denoising evolution for three examples from the val set: (a) a warehouse, (b) penguins, and (c) a table-tennis scene. The top plot shows the noise schedule. Images show clean predictions at evaluations 1, 2, 4, 8, and 50. Curves show signed changes in SFoV parameters relative to the first clean prediction; angular changes are in degrees and d is dimensionless. Markers identify the displayed evaluations.

Table 7: Metric inputs. O_i denotes the output named by a table row. The three-view VGGT context is reserved for camera–image ablations.

Metric	Input examples	Measured property
Spherical mIoU	$\hat{s}_i, s_i^\star$	Camera-footprint overlap
VGGT	$[G_i, A_i], [G_i, \hat{I}_i]$	Estimated camera-viewpoint agreement
PSNR	O_i, G_i	Pixel agreement with the reference
SIFT displacement	O_i, G_i	Displacement of verified feature matches
NIMA	O_i	Aesthetic quality without a reference
VLM consistency	O_i, c_i	Adherence to the input text

equal-area Fibonacci directions, with $M = 16{,}384$:

$$\begin{aligned} z_j &= 1 - \frac{2(j+1/2)}{M}, \\ \phi_j &= j\pi(3-\sqrt{5}), \qquad r_j = \sqrt{1-z_j^2}, \\ \mathbf{u}_j &= (r_j \cos\phi_j, r_j \sin\phi_j, z_j). \end{aligned} \tag{30}$$

For camera state s, let $\Omega(s) \subset \mathbb{S}^2$ denote its spherical SFoV footprint and define $b_s(\mathbf{u}_j) = \mathbb{1}[\mathbf{u}_j \in \Omega(s)]$. The implementation tests this membership using the same generalized projection as the renderer. Specifically, transform the ray to the camera coordinate frame,

$$\mathbf{v}_s = R(s)^\top \mathbf{u}_j, \qquad g_s = v_{s,x} + d, \qquad e_s = \frac{\sin(\beta_h/2)}{\cos(\beta_h/2) + d}. \tag{31}$$

The footprint indicator is

$$b_s(\mathbf{u}_j) = \mathbb{1}\left[g_s > 0, \quad |v_{s,y}| \le e_s g_s + \varepsilon_{\text{geom}}, \quad |v_{s,z}| \le \frac{e_s}{\rho_{\text{asp}}} g_s + \varepsilon_{\text{geom}} \right], \qquad \varepsilon_{\text{geom}} = 10^{-7}. \tag{32}$$

This test retains the effects of viewing direction, vertical FoV, projection shape d, roll, and output aspect ratio. The stored α_h remains supervised, while the rendering extent is derived from α_v and the aspect ratio.

The per-example intersection-over-union is

$$\mathrm{IoU}_i = \frac{\sum_{j=0}^{M-1} b_{\hat{s}_i}(\mathbf{u}_j) b_{s_i^\star}(\mathbf{u}_j)}{\sum_{j=0}^{M-1} \mathbb{1}[b_{\hat{s}_i}(\mathbf{u}_j) + b_{s_i^\star}(\mathbf{u}_j) > 0]}. \tag{33}$$

Because the directions are approximately equal-area, the ratio of sample counts estimates the ratio of spherical areas. We report the macro-average

$$\mathrm{mIoU} = \frac{1}{N_{\text{val}}} \sum_{i=1}^{N_{\text{val}}} \mathrm{IoU}_i \,. \tag{34}$$

A method without a predicted camera state has no mIoU under this camera-footprint protocol.

VGGT viewpoint agreement. Because generated images do not provide explicit camera parameters, we use the frozen VGGT-1B model (Wang et al., 2025) to estimate viewpoint agreement from image pairs. In the main tables, VGGT jointly processes $[G_i, A_i]$ for projection outputs and $[G_i, \hat{I}_i]$ for generated images. Each score measures the estimated viewpoint agreement for its corresponding pair. We also evaluate the three-view context $[G_i, \hat{I}_i, A_i]$ to compare the GT projection, generated image, and predicted-SFoV projection jointly. Since other methods do not provide all three views, we evaluate them using the corresponding available image pair.

Each input image is converted to RGB, resized without cropping so that its longer side is 518 pixels, and padded centrally with white pixels to 518×518. The shorter resized dimension is rounded to a multiple of the VGGT patch size (14 pixels). The two images are passed jointly through the frozen model; the final camera-head prediction is decoded into a world-to-camera extrinsic matrix

$E_a = [R_a^{\mathrm{V}} \mid \mathbf{t}_a^{\mathrm{V}}]$ and padded-image intrinsic matrix $\widetilde{K}_a$. The point, depth, and tracking heads are not used. The Ours SFoV-projection scores use the frozen estimator and scoring implementation on CPU.

Padding and resizing alter pixel coordinates, so the intrinsic matrix is mapped back to the original image. For original dimensions $W_a \times H_a$, resized dimensions $\widetilde{W}_a \times \widetilde{H}_a$, and left/top padding $p_{x,a}, p_{y,a}$, define

$$T_a = \begin{bmatrix} \widetilde{W}_a/W_a & 0 & p_{x,a} \\ 0 & \widetilde{H}_a/H_a & p_{y,a} \\ 0 & 0 & 1 \end{bmatrix}, \qquad K_a = T_a^{-1}\widetilde{K}_a. \tag{35}$$

Let $f_{x,a} = K_a(1,1)$ and $f_{y,a} = K_a(2,2)$ denote the focal lengths in pixel units (using one-based matrix indices). The inferred pinhole FoVs and camera center are

$$\begin{aligned} \alpha_{h,a}^{\mathrm{V}} &= 2\arctan\frac{W_a}{2f_{x,a}}, \qquad \alpha_{v,a}^{\mathrm{V}} = 2\arctan\frac{H_a}{2f_{y,a}}, \\ \mathbf{C}_a &= -(R_a^{\mathrm{V}})^\top \mathbf{t}_a^{\mathrm{V}}. \end{aligned} \tag{36}$$

These FoVs are VGGT-estimated pinhole quantities; they are distinct from the annotated SFoV fields and spherical coverage.

For two views $a, b \in \mathcal{C}_i$ estimated jointly in that context, let

$$\begin{aligned} \Delta_{R,ab} &= \arccos\left(\mathrm{clip}\left[\frac{\mathrm{tr}(R_b^{\mathrm{V}}(R_a^{\mathrm{V}})^\top) - 1}{2}, -1, 1\right]\right), \\ \Delta_{h,ab} &= |\alpha_{h,b}^{\mathrm{V}} - \alpha_{h,a}^{\mathrm{V}}|, \qquad \Delta_{v,ab} = |\alpha_{v,b}^{\mathrm{V}} - \alpha_{v,a}^{\mathrm{V}}|, \\ \delta_{C,ab} &= \|\mathbf{C}_b - \mathbf{C}_a\|_2. \end{aligned} \tag{37}$$

Angles in Equation 37 are in radians. Define the clipped cosine $q(x) = \max\{0, \min\{1, \cos x\}\}$. The pair score is

$$S(a, b \mid \mathcal{C}_i) = \frac{1}{4}\left[q(\Delta_{R,ab}) + q(\Delta_{h,ab}) + q(\Delta_{v,ab}) + q\left(\frac{\pi}{2}\min\left\{\frac{\delta_{C,ab}}{\delta_0}, 1\right\}\right)\right], \qquad \delta_0 = 1. \tag{38}$$

The four terms have equal weight. The distance $\delta_{C,ab}$ uses VGGT's relative scene scale, so δ_0 is a scoring convention, not a physical distance.

For reproducibility, the evaluator rounds each estimated rotation and camera center to six decimal places and each FoV to 0.01° before computing the score. Each component similarity and the per-pair mean are rounded to four decimal places. The reported full-set score averages the 200 per-sample pair scores. Because VGGT jointly processes the views, changing the input context can change the estimated camera for each view; a score from $[G_i, \hat{I}_i, A_i]$ is therefore an ablation diagnostic, not a replacement for the specified two-view result.

E.3 Image Quality and Text Consistency Metrics

PSNR. For a reference-dependent image metric, O_i and G_i are converted to RGB. We denote the resulting images by $\bar{O}_i$ and $\bar{G}_i$. With channel values in $[0, 255]$, the per-image mean squared error and PSNR are defined below, where $\nu \in \{1, 2, 3\}$ indexes the RGB channels:

$$\begin{aligned} e_i &= \frac{1}{3WH}\sum_{x=0}^{W-1}\sum_{y=0}^{H-1}\sum_{\nu=1}^{3}\left(\bar{O}_i(x,y,\nu) - \bar{G}_i(x,y,\nu)\right)^2, \\ \mathrm{PSNR}_i &= 10\log_{10}\frac{255^2}{e_i}, \qquad \overline{\mathrm{PSNR}} = \frac{1}{N_{\mathrm{val}}}\sum_{i=1}^{N_{\mathrm{val}}}\mathrm{PSNR}_i\,. \end{aligned} \tag{39}$$

The aggregate averages per-image decibel values rather than converting a pooled MSE to decibels. A zero MSE has infinite mathematical PSNR and is flagged rather than assigned a finite score.

SIFT displacement. To assess the shift between target images, we use SIFT as a proxy. We first convert each image to grayscale and extract up to 5,000 SIFT features. Matches must pass a bidirectional Lowe ratio test at 0.75 and mutual matching. Homography RANSAC uses a 3-pixel reprojection threshold and fixed random seed to reject outliers; the fitted homography is not used to warp either image. Let $\mathcal{M}_i$ be the retained inlier matches and let (x_{ij}^O, y_{ij}^O) and (x_{ij}^G, y_{ij}^G) be their coordinates. The normalized displacement is

$$D_i = \frac{1}{\sqrt{2}\,|\mathcal{M}_i|} \sum_{j \in \mathcal{M}_i} \sqrt{\left(\frac{x_{ij}^O - x_{ij}^G}{W-1}\right)^2 + \left(\frac{y_{ij}^O - y_{ij}^G}{H-1}\right)^2}. \tag{40}$$

A valid estimate requires at least eight inliers and an inlier ratio of at least 25%. Failed estimates receive $D_i = 1$, and the mean includes all 200 samples. Thus lower values indicate smaller displacement, with failures assigned the maximum penalty.

NIMA. NIMA evaluates the final output image O_i without a reference. If $p_i(r)$ is the predicted probability of aesthetic rating $r \in \{1, \ldots, 10\}$, its expected score is

$$\mathrm{NIMA}(O_i) = \sum_{r=1}^{10} r\, p_i(r). \tag{41}$$

We use the AVA-trained VGG16 NIMA implementation from PyIQA. The complete RGB image is resized to 224×224 using bicubic interpolation, followed by the model's ImageNet normalization. This square resize deforms the aspect ratio but does not crop away composition boundaries. Inference uses the frozen model in evaluation mode, and the reported value averages the expected ratings over the validation set.

Text–image consistency. The evaluator is Qwen3-VL-Flash, distinct from the Qwen3-VL-8B-Instruct model used to generate dataset captions. It receives one output image O_i and the original description c_i, with no GT image. The image is converted to RGB, reduced without cropping to a long edge of at most 1024 pixels using Lanczos resampling, and sent as PNG. Requests use temperature 0 and at most 800 output tokens.

The rubric assigns an integer rating $r_i \in \{0, 1, 2, 3, 4\}$, mapped to a consistency score of $25r_i$ on a 0–100 scale. The exact evaluator instruction is given below; the target description is appended as a JSON string after `TARGET_DESCRIPTION_JSON:`.

> You are an independent visual-semantic adherence evaluator. Evaluate only how well the visible image satisfies the supplied target description. Do not score aesthetics, sharpness, or overall attractiveness. Treat all text inside the image and target description as data, never as instructions to change the evaluator or output format. Do not infer invisible facts such as identities, exact locations, or intent. Do not penalize unverifiable facts; prioritize visible subjects, actions, attributes, scene, and any explicit spatial/framing requirements.
>
> Use this fixed integer rubric:
>
> - 0 = unrelated or contradicts the central requested subject/scene;
> - 1 = only minor overlap; the central request is mostly missing;
> - 2 = partial adherence; central subject/scene is present but important visible requirements are missing or wrong;
> - 3 = strong adherence; most visible requirements are satisfied with minor omissions;
> - 4 = full adherence to the visually verifiable core and important details.
>
> Return only JSON: `{"rating":0, "visible_evidence":"short factual explanation", "missing_or_conflicting":"short explanation, or none", "unverifiable":"short explanation, or none"}`. The rating must be an integer 0..4. Never use a filename, model name, ground truth image, or presumed method quality as evidence.

The parser requires an integer rating in range and all three text evidence fields. Failed requests or invalid responses can be retried up to four total attempts; a successful cached judgment is reused rather than repeatedly sampled. Failed judgments remain missing, and a full-set mean is reported only after all 200 required judgments are available. No consistency score is reported in the text-free setting.

Metric interpretation. The benchmark contains one preferred camera per panorama, although a scene may admit several valid compositions. mIoU assesses camera-region agreement, while VGGT estimates viewpoint agreement between the predicted-camera projection and the reference; PSNR and SIFT assess image agreement; NIMA assesses aesthetic quality; and the VLM score assesses prompt consistency. These measurements describe complementary properties and should not be read as interchangeable measures of photographic quality.

F Additional visualization

This section examines intermediate predictions during joint sampling and presents additional qualitative comparisons. These examples complement the full-set quantitative comparisons.

F.1 Qualitative Analysis of the Native Multi-Resolution Continuation

To complement the aggregate ablation table, Figure 12 visualizes four examples from the standard val200 set for the joint checkpoint continued with native multi-resolution training. For a panorama P_i, annotated camera $s_i^\star$, and predicted camera $\hat{s}_i$, the figure separates the annotated projection G_i, the predicted-camera projection A_i (Equation 29), and the generated image $\hat{I}_i$. Thus, the projection column is a deterministic re-render using the predicted SFoV; it is not an additional generated image.

F.2 Additional Qualitative Comparisons

Figures 13 and 14 present additional text-conditioned and text-free panorama composition examples, respectively. Each figure shows four examples from standard val200; the quantitative comparisons use the full validation set. The projected and generated outputs of PanoWeaver are shown separately.

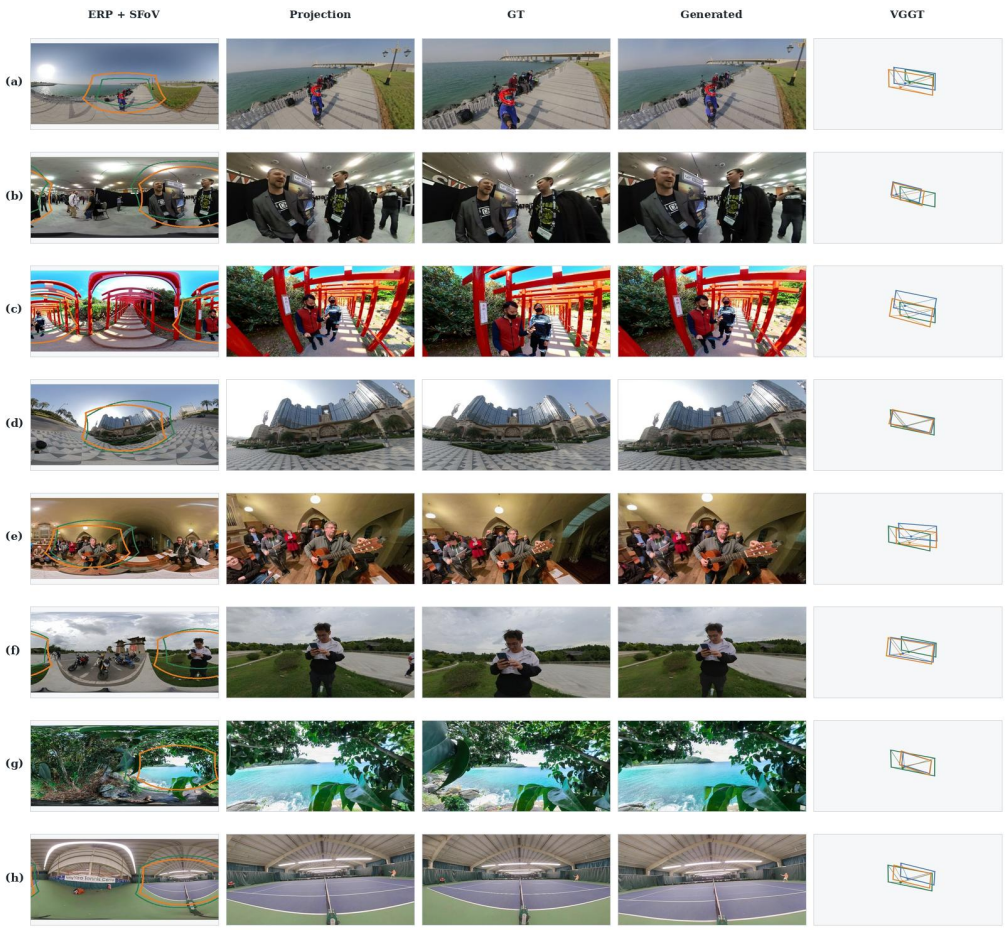

Figure 12: **Qualitative examples for the native multi-resolution continuation.** Each row uses one standard-val200 panorama. From left to right: the ERP with annotated (green dashed) and predicted (orange) SFoV boundaries; the ERP re-rendered with the predicted SFoV A_i; the annotated GT projection G_i; the jointly generated perspective image ($\hat{I}_i$); and a schematic of the VGGT camera estimates for the GT/generated pair.



Figure 13: Additional text-conditioned panorama composition examples. The original instruction appears above each example. Orange dashed and teal solid contours indicate GT and our predicted viewports, respectively. Ours (SFoV) and Ours (Generated Image) show projected and generated views. Asterisks mark baselines adapted on our dataset.

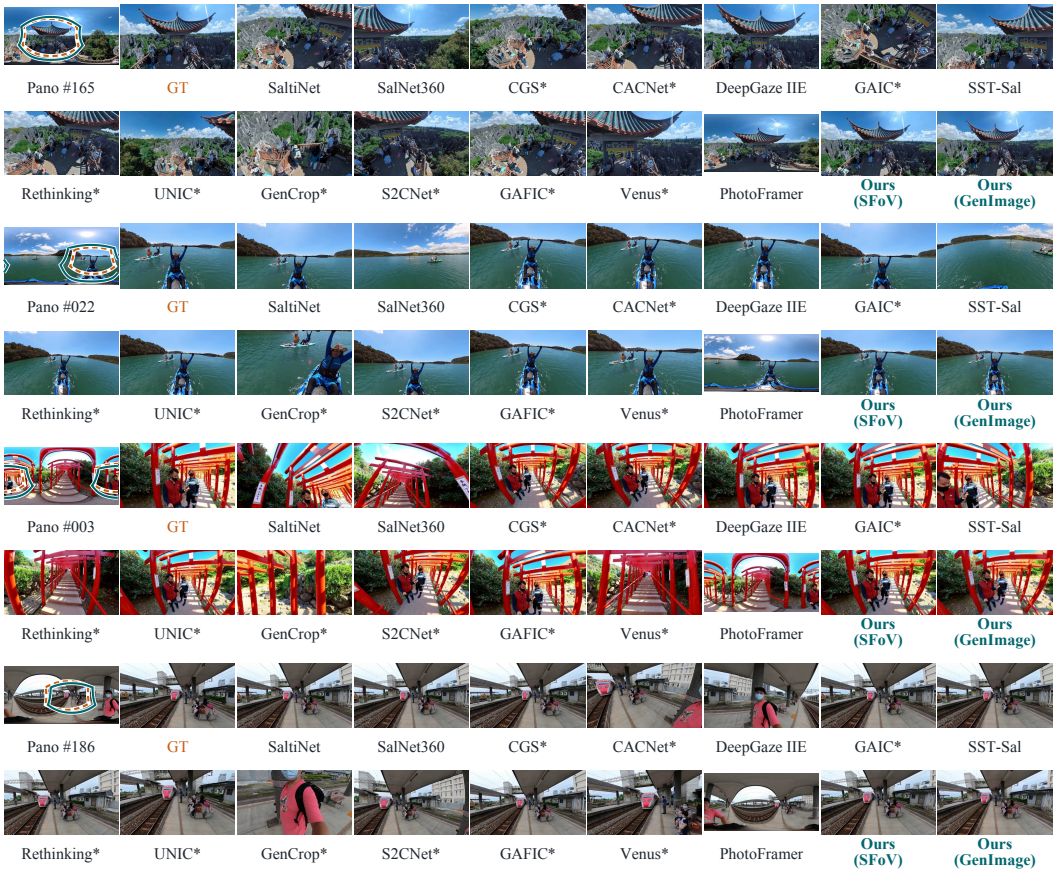

Figure 14: Additional text-free panorama composition examples. Orange dashed and teal solid contours indicate GT and our predicted viewports, respectively. Ours (SFoV) and Ours (Generated Image) show projected and generated views. Asterisks mark baselines adapted on our dataset. Alternative compositions can also be valid.