A User Study.

The user study can be found on the link: https://eyetoeye-anonym.github.io/eye2eye-sm/vr-viewer-userstudy. Use the VR browser to view it.

B Additional Details

B.1 Training details

We fine-tune Lumiere on a dataset of 100K clips from Stereo4D as mentioned in Section 4.3 of the main paper. We temporally subsample the videos into 80 frames at 16 fps to match Lumiere's pre-training temporal resolution. We train the model for 120K steps with batch size 32 using 32 Tpu V5 chips. We employ the ada factor optimizer [Shazeer and Stern 2018] with its default configuration and a constant learning rate of $2 \cdot 10^{-5}$.

The original clips resolution is 512×512 pixels. To train the Eye2Eye base model, we additionally downsample the frames spatially to 128×128 pixels. For the Eye2Eye refiner, we randomly sample crops of 128 pixels.

B.2 Sampling hyper-parameters for our method

B.2.1 Base Eye2Eye sampling

We sample with 50 diffusion timesteps and without classifier-free guidance. We sample from this model at a resolution of 256 pixels, as we found that this resolution best mitigates visual quality and 3D effect.

B.2.2 Eye2Eye refiner

We upsample the output of the base Eye2Eye model to 512×512 pixels resolution and noise it to diffusion timestep $t = 0.9$. We then denoise it with 48 diffusion timesteps and without classifier-free guidance

C Baselines

C.1 Warp-and-inpaint implementation

For a fair comparison with the warp-and-inpaint approach, we implement and train this baseline using the same pretrained model as in our method. We use the same dataset described in 4.3 to fine tune the base Lumiere inpainting model to inpaint left-right disocclusion masks. We use [46] to estimate disparity of each pair of stereo frames, $V^{\text{left}}, V^{\text{right}}$ and obtain the disocclusion mask by computing left-right consistency of the disparity prediciton. At training, the model is conditioned on the right video warped according to the estimated disparity, $V^{\text{right}}_{\text{warped}}$, and the corresponding disocclusion mask M, to denoise the left frame, with the standard diffusion objective:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t, V^{\text{right}}_{\downarrow\text{warped}}, M, c)\|_2^2\right] \quad (3)$$

Here c is the text caption, $x_t = \sqrt{\alpha_t} V^{\text{left}} + \sqrt{1-\alpha_t}\epsilon$, and $\epsilon \sim \mathcal{N}(0, I)$. Denote by $\theta(x_t, t, V^{\text{right}}_{\text{warped}}, M, c)$ this model after training. At inference time, given a video V^{right}, we use SOTA monocular disparity estimation [30] to estimate video disparity D^V. As this estimation is scale and shift invariant, we fit a scale and shift parameter to the disparity map to align it with the disparity of our outputs (we first estimate the disparity of our outputs using [46]). We then forward-warp the frames using depth ordered softmax splatting [61] and downsample the warped frames to obtain $V^{\text{right}}_{\text{warped}\downarrow}$. The inpainting mask here are the pixel locations that were not mapped onto by D^V. We open and dilate the mask to reduce temporal inconsistencies before feeding it along with the downsampled right eye video to θ model, to obtain a low resolution inpainted video:

$$\theta(x_T, T, V^{\text{right}}_{\text{warped}\downarrow}, M, c) = V^{\text{inpainted}}_{base}$$

For spatial super resolution, we use the pretrained Lumiere SSR model and take a blended diffusion approach for maintaining faithfulness to the original video. Specifically, the input to the SSR model is the low resolution base inpainting model output $V_{base}^{\text{inpainted}}$, and at each timestep t, we blend the predicted clean super-resolved output

$$\hat{x_0^t}(x_t, t, V_{base}^{\text{inpainted}})$$

with the high resolution warped right video

$$V_{\text{warped}}^{\text{right}} = \text{softmax\_z\_splatting}(V, D^v)$$

according to the dissocclusion mask M:

$$\hat{x_0^t} \leftarrow M \cdot \hat{x_0}^t(x_t, t, V_{base}^{\text{inpainted}}) + (1 - M) \cdot V_{\text{warped}}^{\text{right}}$$

This blending ensures that details in areas that appear in the input right video are preserved in the super-resolved left view. We use a the standard lumiere sampling of 256 and 32 diffusion timesteps for the base model and the SSR model, respectively, and a classifier free guidance of 8.

C.2 Stereo-Crafter

We use the official Stereo-Crafter repository `ttps://github.com/TencentARC/StereoCrafter`. For the depth splatting stage, we scale and shift the predicted disparity in the same way described in C.1

C.3 Deep3D

As the original paper implementation uses a deprecated codebase, we turn to a more recent implementation found in the link: `https://github.com/HypoX64/Deep3D`. Their training data consists of 3D movies, which are typically processed in a different manner then our data—the zero disparity plane is usually shited to increase human comfort, making the RGB comparison difficult. We thus encourage the viewer to use anaglyph glasses for these results.

C.4 Dynamic Gaussian marbles

We optimize the Dynamic Gaussian Marbles using the official paper implementation `https://github.com/coltonstearns/dynamic-gaussian-marbles`, using their default real-world videos configuration. We observed the optimizing the representation for the full number of steps (100K) in this configuration diverges, and thus synthesize stereo views from it after 40K steps.