Deep 3D Mask Volume for View Synthesis of Dynamic Scenes
Supplementary Material

Layer	kernel size	stride	dilation	in	out	activation	input
conv1_1	7	1	1	12	8	ReLU	PSVs
conv1_2	7	2	1	8	16	ReLU	conv1_1
conv2_1	3	1	1	16	16	ReLU	conv1_2
conv2_2	3	2	1	16	32	ReLU	conv2_1
conv3_1	3	1	1	32	32	ReLU	conv2_2
conv3_2	3	2	1	32	64	ReLU	conv3_1
conv4_1	3	1	1	64	64	ReLU	conv3_2
conv4_2	3	1	1	64	64	ReLU	conv4_1
up5		2		128	128		conv3_2 + conv4_2
conv5_1	3	1	1	128	32	ReLU	nnup5
conv5_2	3	1	1	32	32	ReLU	conv5_1
up6		2		64	64		conv2_2 + conv5_2
conv6_1	3	1	1	64	16	ReLU	nnup6
conv6_2	3	1	1	16	16	ReLU	conv6_1
up7		2		32	32		conv1_1 + conv6_2
conv7_1	3	1	1	32	16	ReLU	nnup7
conv7_2	3	1	1	16	8	ReLU	conv7_1
conv7_3	3	1	1	8	1	Sigmoid	conv7_2

Table 1. Details of each layer in our 3D mask network.

1. Network Architecture

Our view synthesis pipeline utilizes two different 3D CNNs to predict the MPI volumes and the 3D mask volume as described in Sec. 4.2 in the main paper. Both networks have similar structures as the one in Mildenhall et al.[3]. However, we made some adjustments to keep the network light for faster training and less memory consumption. We show detailed layers for the mask network in Table 1. The MPI network has the same structure except for some changes in the overall input and output channels to account for different view counts.

2. Ablation Studies on Loss Function

As discussed in Sec. 4.3 and Sec. 5.2 in the main paper, we experiment with different losses to see if we can acquire a 3D mask volume that is more interpretable and possesses physical meaning. Two additional loss functions are described as follows. The first loss is a mask supervision loss $\mathcal{L}_m$, which forces the mask volume to match the shape of the dynamic object in the scene. The second loss is a sparsity loss $\mathcal{L}_s$ applied on the mask volume to encourage the network to reuse $\hat{\mathbf{M}}$ more. To be more specific, for the mask loss, we use the work by Lin et al.[1], which takes the individual frame $\mathbf{I}$ and the background $\hat{\mathbf{I}}$ in the video to generate a dynamic object mask $\mathbf{V}_{gt}$ we later use as supervision. To supervise the mask volume, we directly regularize the over-composited alphas from the warped foreground MPI volume $\mathcal{W}(\mathbf{M} \odot \mathbf{V})$ to be consistent with $\mathbf{V}_{gt}$. We denote the over-composited alpha values as m_1. This mask loss is similar to the mask supervision loss in Lu et al. [2]. We calculate the estimated background mask m_0 by dilating the foreground mask with a kernel of size $(5, 5)$ to produce m_1'. The background mask is then $m_0 = 1 - m_1'$. And the mask supervision loss is:

$$\mathcal{L}_m = \frac{||m_1 \odot (1 - \mathbf{V}_{gt})||_1}{2||m_1||_1} + \frac{||m_0 \odot \mathbf{V}_{gt}||_1}{2||m_0||_1}. \quad (1)$$

Table 2. Effect of different loss functions. Our rendering loss offers better temporal consistency and slightly better visual quality.

Methods	STRRED↓	PSNR↑	SSIM↑
Ours	**0.1683**	**26.22**	0.8390
Ours w/ $\mathcal{L}_s$	0.1745	26.18	**0.8393**
Ours w/ $\mathcal{L}_s, \mathcal{L}_m$	0.1900	26.09	0.8374

Another loss is a L_1 sparsity constraint on the mask volume to ensure it only covers the necessary portions,

$$\mathcal{L}_s = ||\sum_{(x,y,d)} \mathbf{V}(x, y, d)||_1. \quad (2)$$

We use $\mathcal{L} + 0.1\mathcal{L}_s + 0.25\mathcal{L}_m$ for the full combination and $\mathcal{L} + 0.1\mathcal{L}_s$ for the additional sparsity constraint.

As shown in Table 2 (same as Table 4 in the original paper), our rendering loss still offers the most temporally-stable results, whereas the other two losses trade temporal consistency for better interpretability. It is reasonable that the mask supervision loss helps the network to give a sparser and tighter prediction on the dynamic objects. However, it does not take into account the movements of the foliage and the shadows, producing slightly unstable results in those areas. The sparsity constraint is able to achieve marginally better quality than the full $\mathcal{L}_s, \mathcal{L}_m$ combination as it retains some parts of the scene which might cover the slight differences between frames.

Mask visualization can be found in Fig. 1. From the figure, we can observe that our mask volume removes areas around the edges of the dynamic object and the occluded areas behind it. Moreover, the mask softly blends the shadows cast by the moving object. Adding $\mathcal{L}_s$, the mask becomes sparser, ignoring most static areas. However, as shown in Fig. 1, it still contains some areas around the plants on the left and the building in the back. With $\mathcal{L}_s, \mathcal{L}_m$, the mask has more physical meaning and the resulting 3D mask only covers the dynamic object. This might be useful to extract moving objects for other uses such as editing or object insertion.

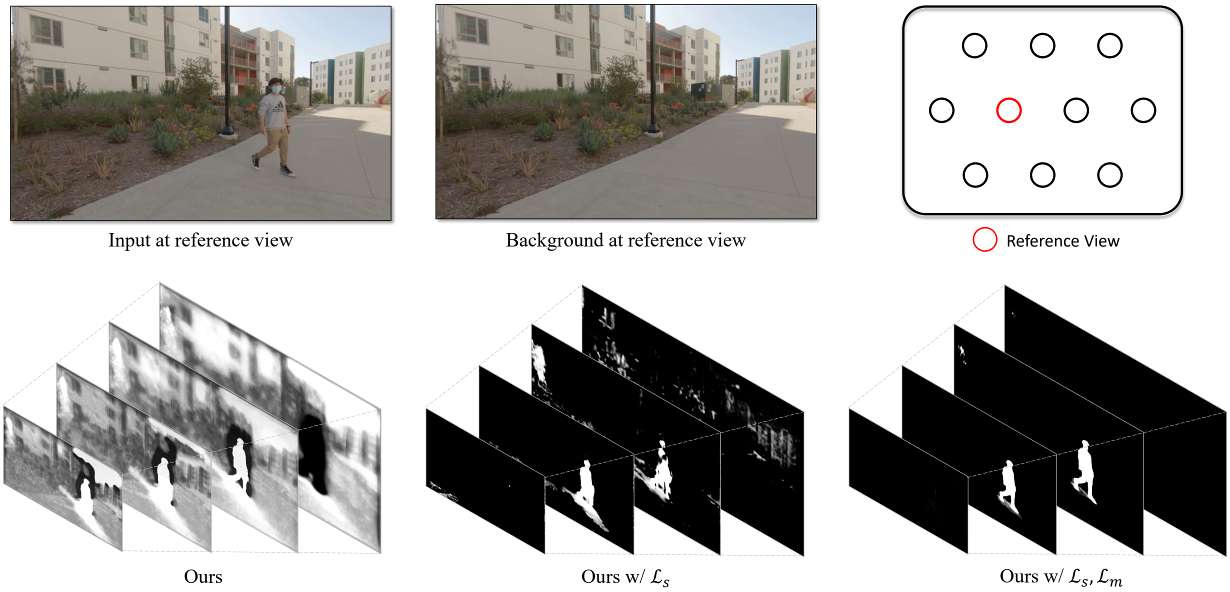

Input at reference view
Background at reference view
Reference View
Ours
Ours w/ $\mathcal{L}_s$
Ours w/ $\mathcal{L}_s, \mathcal{L}_m$

Figure 1. 3D visualization of the masks from different loss functions. With alpha values from the instantaneous MPI, we collapse the mask volumes using over composite to reduce plane count from 32 to 4 for better visualization. (e.g. plane 1∼8 to the furthest plane, ..., plane 25∼32 to the nearest plane.) Note that there is no supervision on static parts in our final loss function, so the values in those parts are unconstrained, resulting in soft blending between instantaneous frames and the background. In general, the 3D mask achieves better temporal consistency by replacing the erroneous disoccluded parts with correct background observations.

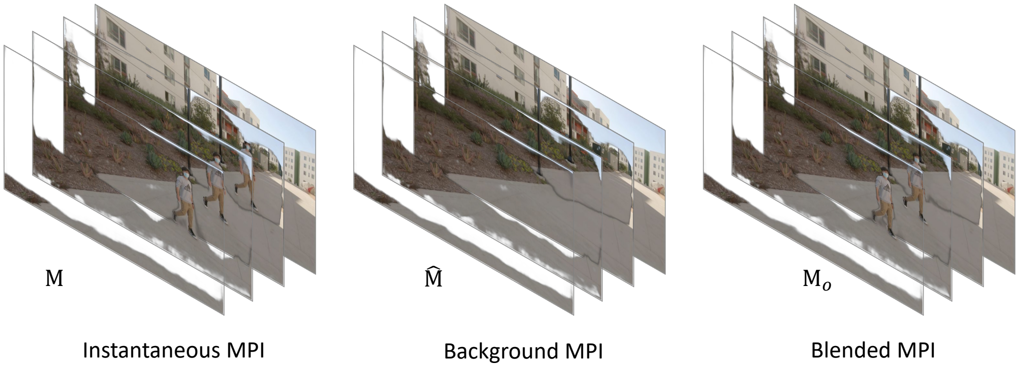


Figure 2. 3D visualization of the MPI volumes using our loss function $\mathcal{L}$. Note that the person on the furthest plane in $\mathbf{M}$ is replaced by the background in $\mathbf{M}_o$.

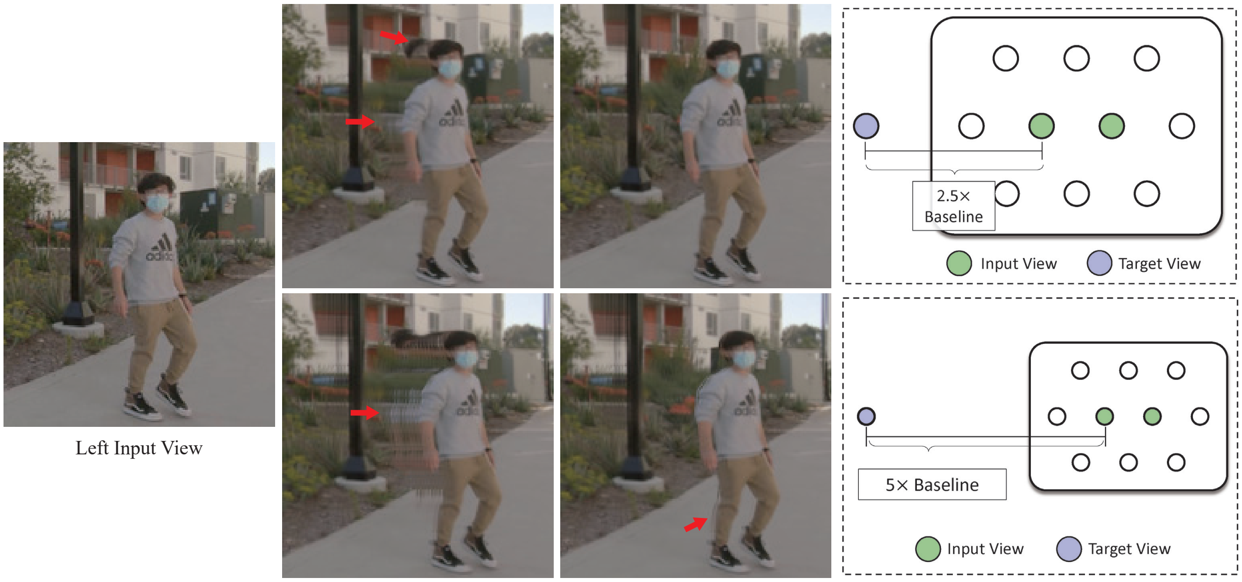
We further examine the 3D visualization of $\mathbf{M}, \hat{\mathbf{M}}$, and $\mathbf{M}_o$ in Fig. 2. Note that in the blended MPI $\mathbf{M}_o$, the occluded area behind the person is filled with actual background information, unlike in $\mathbf{M}$, which has repeated texture of the dynamic object. Since we do not enforce any constraints on the static parts of the scene, our mask has random values in these areas and softly blends them with the background MPI. *This does not affect temporal consistency too much as the difference is minor and some areas are free space which does not contribute any color to the MPI volume as shown in Fig. 2.*

3. Large distance view extrapolation

In Fig. 3, we show results when the target camera is translated far more than the baseline of the input camera pair. When large translational movement is introduced, the conventional method[3] starts to show artifacts in the disoccluded regions. On the contrary, our method still preserves the background details even when the motion is larger, offering a more graceful reduction in quality as the distance is increased.

4. Extension to more input views

Although our proposed method primarily targets binocular view extrapolation, we also demonstrate that it can be

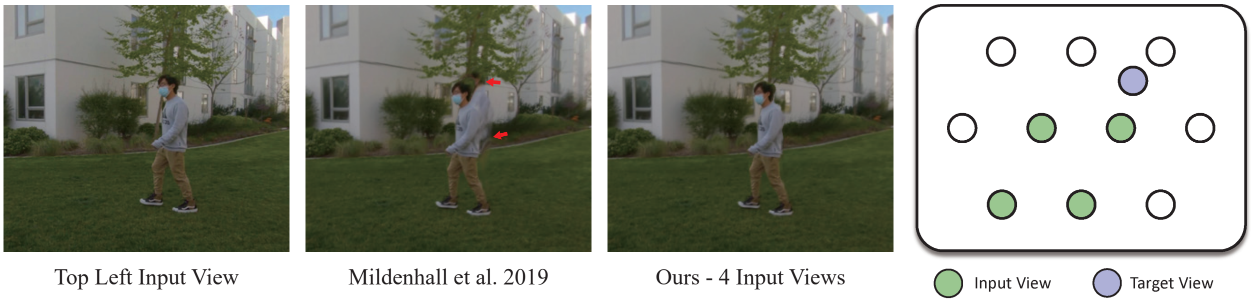


Left Input View

Mildenhall et al. 2019

Ours

Figure 3. Our algorithm is able to provide better visual quality than baseline methods even when the novel viewpoint is far away from the input view. We show results when the baseline is 2.5× and 5× baseline between input views. Note that in the 5× case, our method produces fewer artifacts compared to Mildenhall et al.[3], offering a more graceful degradation.



Top Left Input View

Mildenhall et al. 2019

Ours - 4 Input Views

Figure 4. Our proposed method can also be extended to take 4-view input. We feed 4 input views to both the MPI and mask networks to acquire our result. Here the baseline method is also adjusted to use 4 input views instead of 2. Notice that the artifacts around the person do not appear in our result.

extended to utilize more input views in Fig. 4 and in the supplementary video. With more input views, it can acquire better scene geometry for some cases where there are ambiguities in the plane sweep volume. For example, some ambiguities might occur when there is straight texture-less structure (beams or handrails) parallel to the camera baseline. Using additional cameras can provide more geometric information and avoid similar situations. In Fig. 4, the main difference is that we modify our network to take 4 input views, which convert to 4 instantaneous images and 4 background images as input to the mask network, and output the 3D mask volume as in the pipeline shown in Fig. 4 in the main paper.

References

[1] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sengupta, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Real-time high-resolution background matting. *arXiv*, pages arXiv–2012, 2020. 1

[2] Erika Lu, Forrester Cole, Tali Dekel, Weidi Xie, Andrew Zisserman, David Salesin, William T. Freeman, and Michael Rubinstein. Layered neural rendering for retiming people in video. *ACM Trans. Graph.*, 39(6), Nov. 2020. 1

[3] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 2019. 1, 2, 3