Supplementary Material
TransFill: Reference-guided Image Inpainting by Merging Multiple Color and Spatial Transformations

Yuqian Zhou[1], Connelly Barnes[2], Eli Shechtman[2], Sohrab Amirghodsi[2]
[1]IFP, UIUC, [2]Adobe Research

1. Ablation Study

1.1. Color-Spatial Transformer

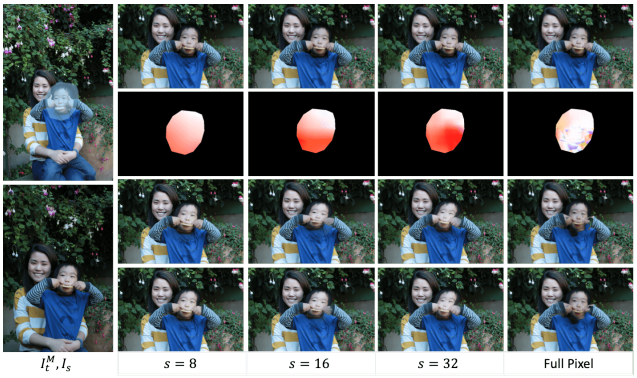

Figure 1: Ablation study on the resolution setting in the Color-Spatial Transformer. First row, direct composition of target image I_t^M and one of the single-homography transformed source images I_s^i. Second row, the learned pixel-wise warping field A_s^i visualized using color wheel in [1]. Third row, the color-spatial transformed image $\hat{I}_s^i$. Last row, the final merging result I_o.

Recall that while introducing the Color-Spatial Transformer, we intend to preserve the texture details and the rigidity of the source image contents. Therefore, given $A_c^i = [K_c^i \quad b_c^i] \in \mathbb{R}^{W\times H\times 3\times 4}$, and $\bar{A}_s^i = B_s(u_s^i) \in \mathbb{R}^{s\times s\times 2}$, we fix $s = 8$ and $d = 8$ in our experiments. We find d does not influence the performance a lot, and the guidance map is automatically learned to uniformly span the necessary bins like in the HDRNet[2]. Figure 1 shows the comparison when we set different s values. It suggests that increasing s gives more degrees of freedom to the learned warping field A_s^i. However, while encountering larger holes like in Figure 1, better flexibility does not better align the contents as expected, but distorts the contents inside the hole. The transformed color field also becomes less smooth as s increases. In an extreme case, suppose we replace the deep bilateral grid and directly learn a full-resolution pixel-wise color-warping field with total variance constraints as in the last column, the model struggles to infer a reasonable color-warping operation within a large hole.

We conclude that CST with smaller s value like $s = 8$ generalizes better to inference images with varying spatial resolutions. It is mainly due to the ill-posedness of image completion. Unlike conventional image registration tasks where all the pixels of the matched regions are available, hole regions are missing in the inpainting task. Less freedom in the hole area preserves better content integrity and semantics.

1.2. Network Components

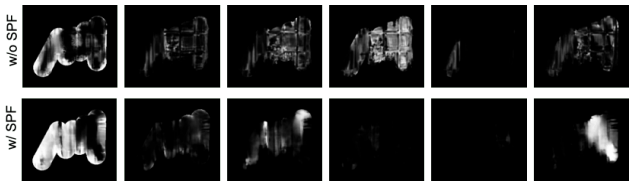

Figure 2: Final fusion masks $\bar{c}_i$ learned by the model with or without the Single-Proposal Fusion (SPF) module. By using SPF outputs as guidance to learn the MPF, the final weights learned tend to be more sparse.

Importance of Single-Proposal Fusion (SPF) Our experiments exhibit that the proposed Single-Proposal Fusion (SPF) module before the Multi-Proposal Fusion (MPF) is necessary for effectively learning the final merging weights of all the proposals. We find directly learning the weights to fuse all the proposals is very challenging. The learned weights have a hard time becoming sparse even though the same total variance loss is imposed. A comparison of the merging mask $\bar{c}_i$ between the model with and without SPF is shown in Figure 2. Using SPF outputs c_i as a structure guidance for learning the fusion of multiple proposals works better in practice.

Correlation between c_i and $\bar{c}_i$ In our experiments, the

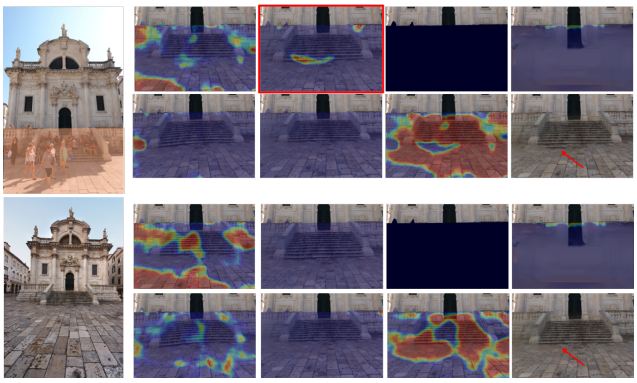

Figure 3: One example of user-involved interactive editing. For the given target and source images on the left, we generate seven proposals including one I_g from ProFill. For the upper group of images, we visualize the regions selected by our model to synthesize the final results. The image with red bounding box yields an unexpected artifacts of stairs. By zeroing out its corresponding c_2, we can correspondingly obtain zero-valued $\bar{c}_2$ as shown in the lower image group. Other maps are also correspondingly redistributed. The final result on the lower-right position is then generated by merging the other selected proposals with nonzero weighted masks. As we can see, the artifact disappears.

learned single-proposal fusion mask c_i and multi-proposal fusion mask $\bar{c}_i$ demonstrate strong correlation. Specifically, by zeroing out one of the c_i, the values in $\bar{c}_i$ will also vanish. This shows the MPF constructs the correspondence to make $\bar{c}_i$ be conditioned on c_i. This provides more flexibility for our model to incorporate user interactions. Suppose users want to eliminate the elements in some proposals, one can simply zero them out and the final results will only be merged from other selected proposals. Such a process is demonstrated in Figure 3.

1.3. APAP with Poisson Blending

We experimented with using Poisson blending [4] combined with APAP. The testing result on the *Small Set* of images with only few non-existing regions is increased from 31.94dB / 0.9738 to 32.56dB / 0.9754 in terms of PSNR / SSIM. However, we did not incorporate Poisson blending in the baseline because we found in some cases there could be significant color bleeding artifacts due to strong color mismatches and non-existing regions especially along the boundary of the hole. Some visual comparisons are shown in Figure 4.

1.4. Using ProFill with Partial Masks

As we stated that single image techniques don't work well for larger holes, while in our work, the single-image inpainting is computed over the full mask area. We also

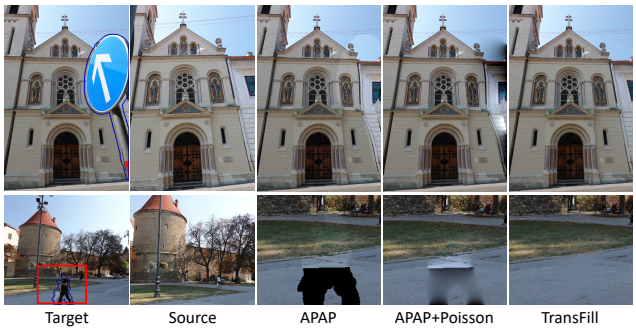

Figure 4: Ablation study on APAP with Poisson blending post-processing. The color bleeding artifacts are significant in some cases when there are strong color mismatches or non-existing regions. APAP does not include image inpainting, so regions that are outside the source image appear as black.

Figure 5: Post-hoc refilling results using ProFill. The TransFill columns show a zoom of the original output, the post-hoc filling result (TransFill + Post), and the region to be refilled from the confidence c_g (Binary Mask). The re-filling with a partial mask may introduce additional artifacts like broken door frames.

thought about using ProFill or other single-image inpainting method with partial mask, but could not find a principled and an end-to-end way to do this. However, we analyzed an approximation of this approach where we used the confidence map c_g estimated by our method, and binarized it to do a post-hoc fill (with ProFill) of each hole region of the target that corresponds to single image inpainting (where the content is not visible in the source image or not well reused). Comparisons are shown in Figure 5. This reveals that since the mask was learned for merging purposes, a post-hoc filling using the mask may introduce other artifacts like broken door frames. The average testing results on RealEstate10K decreased from 37.58dB / 0.9879 / 0.0164 to 37.13dB / 0.9871 / 0.0173 in terms of PSNR / SSIM / LPIPS. However, using partial masks to fill only non-existing regions might work better for images with larger non-existing regions, and become more robust if another approach of learning is taken.

2. User Study Details

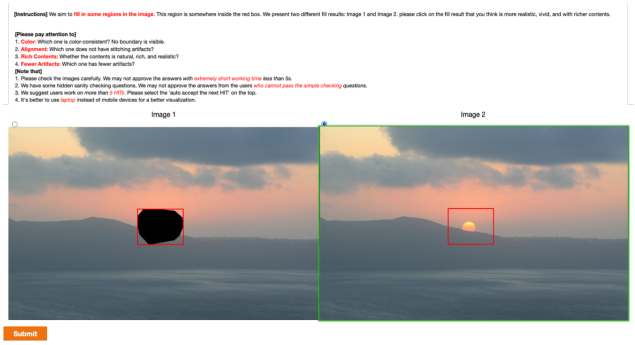

Figure 6: User study interface and check questions. We set up 10 trivial questions to let the users choose which one is more completed.

The GUI of our user study at AMT is shown in Figure 6. To guarantee the reliability of the users' feedback, we require the users to take a qualification test before they evaluate. The test presents users with the 10 trivial pairs I_t and I_t^M and users who answer correctly more than 8 questions are approved to take the official test. We also mix 10 random sanity check questions with the real questions. No users had to be disqualified due to failing the initial test, and only very few users (4 users) got check questions later in the study wrong (5.7% of total opinions), so we conclude that the user responses are reliable.

3. Failure Cases

Figure 7: Failure cases. These demonstrate limitations with large changes in viewing angle, outpainting artifacts from the off-the-shelf single image inpainting module ProFill, and challenges in handling dramatically different lighting environments.

Figure 7 shows some examples of failure cases when the viewing angle changes are large. The color matching module may struggle if there are extreme lighting differences. We may also encounter outpainting artifact issues caused by ProFill.

4. More Visual Results Comparison

Visual Results on RealEstate10K We present more visual results in Figure 8 on the RealEstate10K dataset. Compared with the baselines, our proposed TransFill achieves better spatial alignment and content faithfulness.

Visual Results on Synthetic Adobe-5K In Figure 9, we show more results on the synthetic Adobe-5K dataset to evaluate the performance of our color transformation. As stated in the paper, we synthesize misaligned and color inconsistent images from Adobe-5K dataset. The spatial transformation is a simple homography-based warping, so the CST module works well to align the images and match the color. More challenging cases can be visualized in user-provided images.

More Results on User-provided Images Additional higher-resolution results can be found at the following link: Additional Results.

5. Unfolding the Model: Intermediate Results

In Figure 10 and 11, we unfold the whole pipeline of TransFill to visualize the intermediate results of each proposed module. We demonstrate the process of image completion in a more intuitive way. After proposing different homography-warped images, the CST effectively adjusts the misalignment and color mismatching. Then the proposed TransFill fills in the holes by selectively merging the well-aligned and color-consistent regions from different proposals. Imperfect regions are finally filled with the output from ProFill.

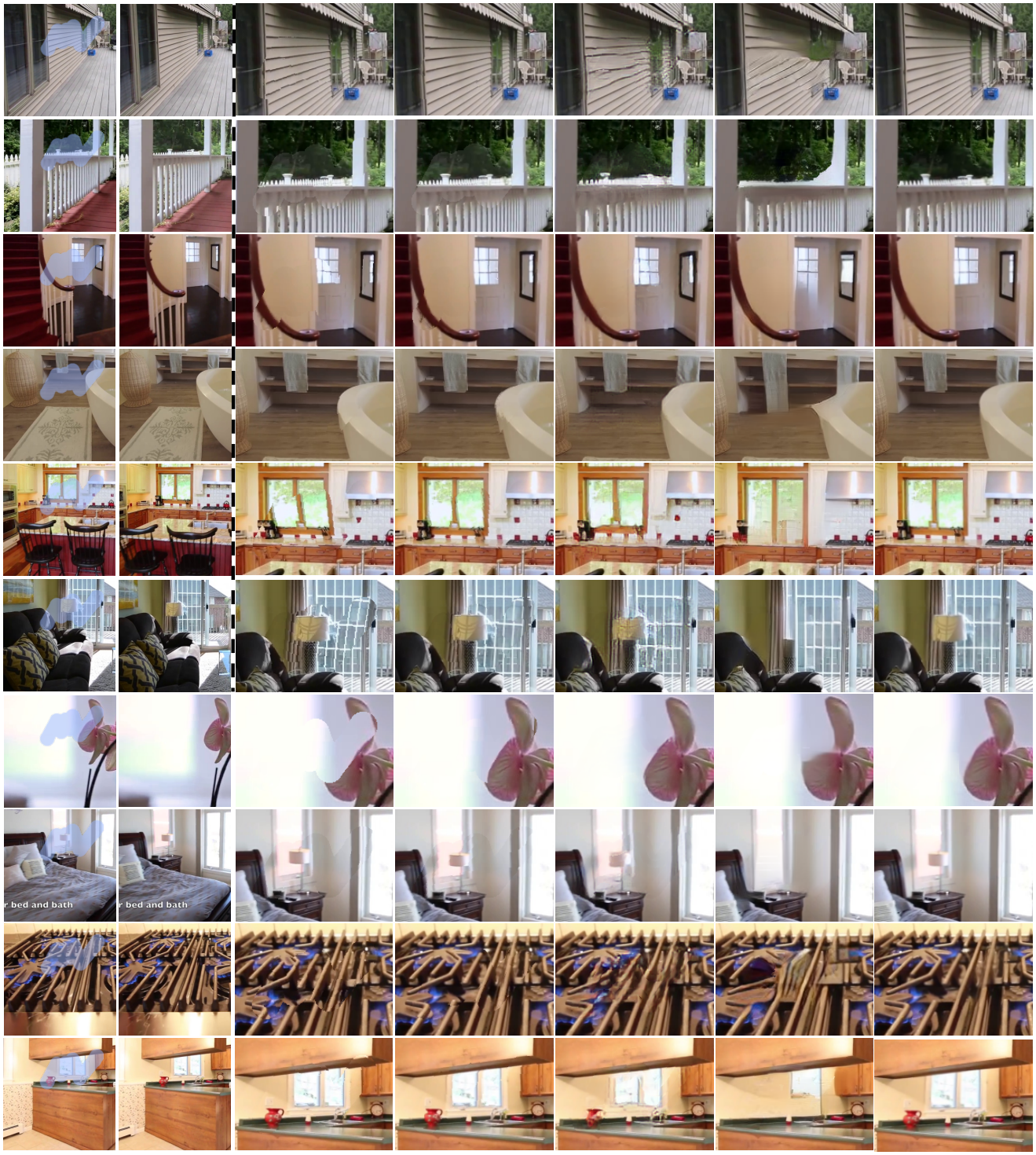

Figure 8: Visual results comparison on the RealEstate10K dataset with FD=10. These have been cropped. **Please zoom in so that there are about 3-4 images across the width of the screen to reveal the significant differences in fine details.** Compared with the baselines, our proposed TransFill achieves better spatial alignment and faithfulness to the source image content.

Figure 9: Visual results on the synthetic Adobe-5K dataset. For each group of photos, the left one is the composition of I_t^M and I_s. We transform the color and warp I_s to make it consistent with I_t^M and composite them as the right image. Our CST module resolves the color mismatches and spatial misalignment problems.

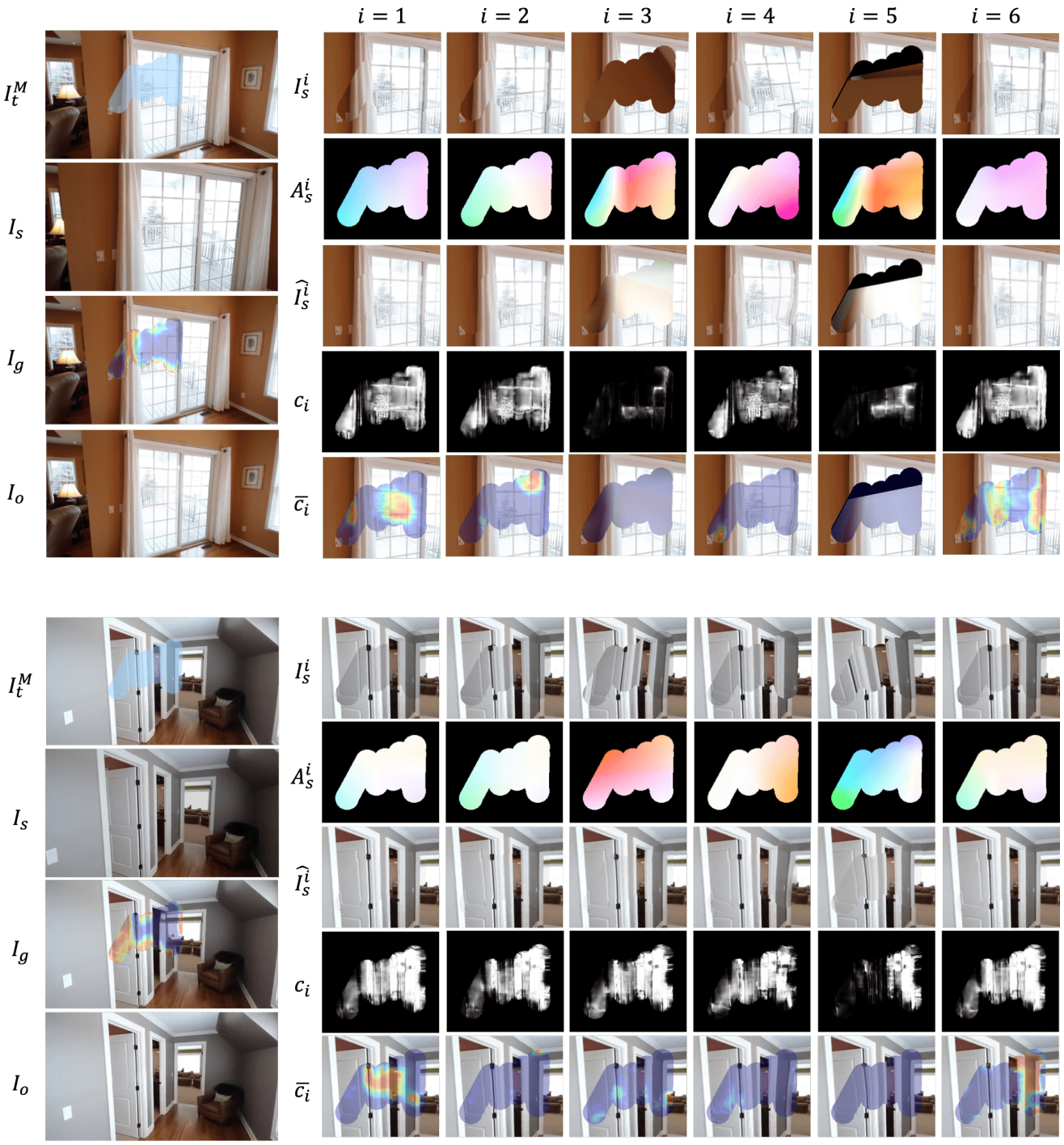

Figure 10: Unfolding the whole pipeline to visualize the intermediate results of each module.

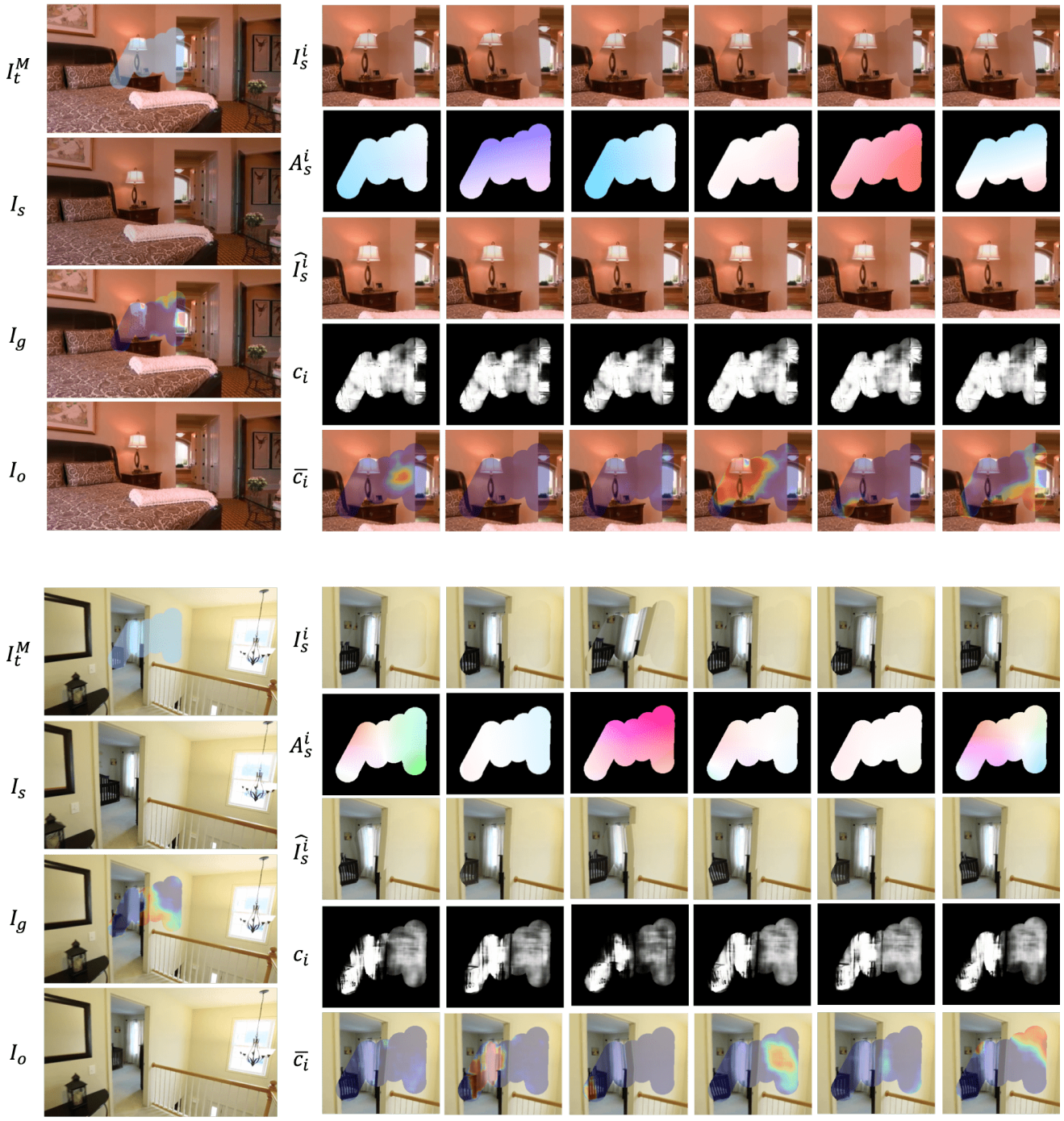

Figure 11: Unfolding the whole pipeline to visualize the intermediate results of each module. For some challenging cases when the line alignment is hard, our model can also leverage the outstanding performance of line generation of ProFill to synthesize the door frame.

6. Network Structures

The network structures are summarized in Table 1, 2 and 3. The structures follow a UNet structure with minor modifications such as some shared structures and some parameter-free components like the tri-linear interpolation layer in the CST. SPF is implemented by a shallower UNet than MPF.

Input	Output	# Out	Type
I_t^M, I_s^i, M	EncConv1_1	32	Conv 3×3
EncConv1_1	EncConv1_2	32	Conv 3×3
EncConv1_2	Pool1	32	Maxpool 2×2
Pool1	EncConv2_1	64	Conv 3×3
EncConv2_1	EncConv2_2	64	Conv 3×3
EncConv2_2	Pool2	64	Maxpool 2×2
Pool2	EncConv3_1	128	Conv 3×3
EncConv3_1	EncConv3_2	128	Conv 3×3
EncConv3_2	Pool3	128	Maxpool 2×2
Pool3	EncConv4_1	256	Conv 3×3
EncConv4_1	EncConv4_2	256	Conv 3×3
EncConv4_2	Pool4	256	Maxpool 2×2
Pool4	EncConv5_1	512	Conv 3×3
EncConv5_1	EncConv5_2	512	Conv 3×3
EncConv5_2	Pool5	512	Maxpool 2×2
EncConv5_2	EncConv6_1	512	Conv 3×3
EncConv6_1	EncConv6_2	512	Conv 3×3
EncConv6_2	ColorCoeff_1	96	Conv 3×3
ColorCoeff_1	ColorCoeff_2	96	Conv 3×3
EncConv6_2	WarpCoeff_1	2	Conv 3×3
WarpCoeff_1	WarpCoeff_2 +Tanh	2	Conv 3×3
I_s^i	GuideConv1_1	16	Conv 1×1
GuideConv1_1	GuideConv1_2 +Tanh	1	Conv 1×1

Table 1: Network structure of CST. Prior to each convolution except EncConv1_1, a PReLU [3] is applied as a pre-activation.

Name	# Out	Type	
I_t^M, M, I_s^i	EncConv1_1	32	Conv 3×3
EncConv1_1	EncConv1_2	32	Conv 3×3
EncConv1_2	Pool1	32	Maxpool 2×2
Pool1	EncConv2_1	64	Conv 3×3
EncConv2_1	EncConv2_2	64	Conv 3×3
EncConv2_2	Pool2	64	Maxpool 2×2
Pool2	EncConv3_1	128	Conv 3×3
EncConv3_1	EncConv3_2	128	Conv 3×3
EncConv3_2	Deconv2	64	Deconv 3×3
Deconv2	Concatenate2	64	Deconv2 , EncConv2_2
Concatenate2	DecConv2_1	64	Conv 3×3
DecConv2_1	DecConv2_2	64	Conv 3×3
DecConv2_2	Deconv1	32	Deconv 3×3
Deconv1	Concatenate1	32	Deconv1 , EncConv1_2
Concatenate1	DecConv1_1	32	Conv 3×3
DecConv1_1	DecConv1_2	32	Conv 3×3
DecConv1_2	DecConv1_3 + Sigmoid	1	Conv 3×3
DecConv1_3	Concatenate_feature	4	DecConv1_3, I_t^M
Concatenate_feature	FeatureConv1_1	3	Conv 3×3
FeatureConv1_1	FeatureConv1_2	3	Conv 3×3

Table 2: Network structure of SPF.

Input	Output	# Out	Type
I_t^M, M, f_s^i, f_g	EncConv1_1	32	Conv 3×3
EncConv1_1	EncConv1_2	32	Conv 3×3
EncConv1_2	Pool1	32	Maxpool 2×2
Pool1	EncConv2_1	64	Conv 3×3
EncConv2_1	EncConv2_2	64	Conv 3×3
EncConv2_2	Pool2	64	Maxpool 2×2
Pool2	EncConv3_1	128	Conv 3×3
EncConv3_1	EncConv3_2	128	Conv 3×3
EncConv3_2	Pool3	128	Maxpool 2×2
Pool3	EncConv4_1	256	Conv 3×3
EncConv4_1	EncConv4_2	256	Conv 3×3
EncConv4_2	Pool4	256	Maxpool 2×2
Pool4	EncConv5_1	512	Conv 3×3
EncConv5_1	EncConv5_2	512	Conv 3×3
EncConv5_2	Deconv4	256	Deconv 3×3
Deconv4	Concatenate4	256	Deconv4 , EncConv4_2
Concatenate4	DecConv4_1	256	Conv 3×3
DecConv4_1	DecConv4_2	256	Conv 3×3
DecConv4_2	Deconv3	128	Deconv 3×3
Deconv3	Concatenate3	128	Deconv3 , EncConv3_2
Concatenate3	DecConv3_1	128	Conv 3×3
DecConv3_1	DecConv3_2	128	Conv 3×3
DecConv3_2	Deconv2	64	Deconv 3×3
Deconv2	Concatenate2	64	Deconv2 , EncConv2_2
Concatenate2	DecConv2_1	64	Conv 3×3
DecConv2_1	DecConv2_2	64	Conv 3×3
DecConv2_2	Deconv1	32	Deconv 3×3
Deconv1	Concatenate1	32	Deconv1 , EncConv1_2
Concatenate1	DecConv1_1	32	Conv 3×3
DecConv1_1	DecConv1_2	32	Conv 3×3
DecConv1_2	DecConv1_3+Softmax	$N+2$	Conv 3×3

Table 3: Network structure of MPF.

References

[1] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International journal of computer vision*, 92(1):1–31, 2011. 1

[2] Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017. 1

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 8

[4] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*, pages 313–318. 2003. 2