

Supplemental Materials: Inpainting at Modern Camera Resolution by Guided PatchMatch with Auto-Curation

Lingzhi Zhang¹, Connelly Barnes², Kevin Wampler², Sohrab Amirghodsi²,
Eli Shechtman², Zhe Lin², and Jianbo Shi¹

¹ University of Pennsylvania

² Adobe Research

1 Full Resolution Results and Comparisons

Because of our research topic of inpainting at modern camera resolutions, we have prepared a 2 gigabyte supplemental material with an included HTML viewer that shows results of all methods at full resolution, and have uploaded that in an anonymized way to cloud file hosting. We checked with a program committee member and were told we can include a link as long as we are very confident we have anonymized it, which we are. The link is https://drive.google.com/file/d/1Lmar1byASRRreJOSimBfWAE2Dt_-gMhor. To prove that the supplemental is not changed since the time of submission, the MD5 sum of the 2 gigabyte supplemental material is 59eb1591b1601f491bd6962e43e81672.

2 Statistical Hypothesis Testing

For the user preferences in Tables 2, 3, and 4 of the main paper, since users preferred our method, we tested whether this preference is statistically significant. We formed 10 null hypotheses: that ours in Table 2 was preferred equally to each of the four baselines, that ours in Table 3 was preferred equally to each of the four baselines, and that ours in Table 4 was preferred equally to the two baselines. We used a one sample permutation t test with 10^6 simulations and found the p values are all 0.0 except in Table 4 the p value for ours against Random-Guided PatchMatch is $p = 0.000107$. After a Bonferroni correction the preference for our method is significant in all cases at a p threshold of 0.01.

3 Curation Network Details

We generate a dataset for pretraining the curation network as follows: we first collect 48229 diverse photos taken by the authors and collaborators in many countries both indoors and outdoors, where the photos are 2K resolution or above, and resize them to 2K. For each image, we generate 10 synthetic holes using the same process and hole dataset described in ProFill [13]: we generate 5 random stroke holes and 5 object-shaped holes, where each hole is constrained

to fit in a randomly generated 512x512 bounding box. Then, for each of these holes, we compute all 8 possible guided PatchMatch results from our pipeline. In total, this results in more than 3 million inpainted images at 2K resolution. To reduce disk space requirements, we first generate the full inpainted 2K image in memory, and then crop it to the 512x512 hole bounding box before saving it to disk. We also collect the corresponding real photo crops before the synthetic holes were added. Additionally, we associate with each fake inpainted image the hole mask used to generate it, and for the corresponding real inpainted image with the same crop bounding box, we associate it with the hole mask used to produce the fake inpainting. We split the dataset into 80% training, 10% validation, and 10% testing images.

When pretraining and when fine-tuning on human preferences, for data augmentations, we use horizontal and vertical flip, crop and resize, Gaussian blur, color jitter, Gaussian noise, rotation, and JPEG compression. When fine-tuning on human preferences, we use a cross-entropy classification loss and train until after the validation accuracy peaks and declines due to the model overfitting to the training set.

Some key lessons we learned in collecting our human preference data are that one should hire photographers either amateur or professional, train them well with meetings where one explains which of a pair of photos is better and why, and validate the quality of their preference data against a reference ground truth. In our case, the reference ground truth was established by an author who has a passion for photography. In previous variants of our data collection, we tried to use Amazon Mechanical Turk workers, but found that even when averaging opinions among many workers, the results were close to random chance. We also tried working one-on-one with expert human labelers who were not photographers, but we found their accuracy was much worse than if the workers were photographers.

4 Automatic Cropping Details

The automatic cropping mentioned in the main paper Section 3.4 works by placing a crop square around the hole and iteratively expanding the crop square from an initial size s_i by a factor $\gamma = 1.05$ until one of two conditions is hit: (1) the crop square is equal size or larger than the image along either dimension, (2) the number of hole pixels as determined by summation is less than a fraction τ of the crop square’s pixels (we use $\tau = 0.25$). The initial size is $s_i = \max(512, h_w, h_h)$, where h_w, h_h are the width and height of the hole bounding box. Within each iteration, the crop square is first centered at the center of the hole bounding box, but in case the crop square moves outside the image, a translation is applied to each axis independently. This translation is such that for each axis, the crop square is moved back entirely inside the image with as few pixels of translation as possible, or if that is not possible, moves as few pixels as possible so all image pixels are visible in the crop along that axis.

5 Discussion of Inpainting Evaluation Metrics

It is well-known that image inpainting lacks good evaluation metrics. Previous inpainting works often use two types of quantitative evaluation metrics. The first type are the direct image content comparison metrics, such as LPIPS [14], PSNR, and SSIM [9], which measure the similarity between pairs of inpainted images and original images. The other type are metrics that operate over the distributions of features within the dataset, such as FID [5] and P/U-IDS [15]. FID measures distribution similarity on the deep features extracted from Inception network [7] for a set of inpainted images and a set of natural images. Similarly, P/U-IDS trains a linear SVM using both deep features of inpainted images and natural images, and classifies whether a deep Inception feature of an inpainted image is real or fake to show how realistic the inpainted images are.

Unlike super resolution or image restoration tasks, we feel that direct image content comparison between the inpainted and original images does not truly reflect the inpainted image quality, especially when the holes are large. The reason is that the inpainted content could look natural and realistic while simultaneously being very different from the original content. In practice, we indeed observe that in many cases that blurry and unnatural results have better scores for the content comparison metrics than the results that are natural but differ greatly from the original image. Clearly, this departs from human preference. Additionally, humans do not really need to look at the original image as reference to tell whether an inpainted image is realistic, so this once again indicates that content comparison metrics might not be an appropriate metric for image inpainting. Thus, we think that metrics FID and P/U-IDS are relatively more appropriate and closer to human perception. This is consistent with the evidence presented in CoModGAN [15] that FID correlates highly with human preference rate as does P/U-IDS. Nevertheless, we report all metrics for the comparison with other state-of-the-art methods in section 4.4, and also conducted extensive user studies in the main paper. We believe that designing better quantitative metrics that tailor to inpainting task could be very useful for the community in the future works.

6 How Do Different Guides Influence Results?

We show an example of how different guides might influence results in Figure 1. In this example, the depth guide (also chosen by our curation module) helps PatchMatch find the reference patches at the similar depth and defocus to synthesize consistent texture. In general, different guides may be particularly useful in different cases, as we discuss in detail in the main paper.

7 Additional Quantitative and User Study Experiments

As an extension of Table 3 in the paper, we show that our method can also effectively boost the performance with older deep inpainting models, as shown



Fig. 1. The top row shows the input image with hole and the guides, and the middle and bottom rows show the 8 guided PatchMatch results using different guides. \star represents the guided result that was chosen by the curation module. The results that use a depth guide generally avoid copying patches that are defocused or have incorrect texture scale. **Please zoom in to see the details.**

in Table 1. Although we encourage inpainting models to be evaluated on benchmarks such as ours that correspond to modern camera sensors, we also show in Table 2 results on our same benchmark dataset for lower resolutions 1K and 2K, which correspond to camera sensors released approximately *two-and-a-half to two decades ago*, respectively [2,1]. We still find our method is always preferred the most according to the user studies at 1K and 2K, always preferred by the quantitative metrics at 2K, and is usually in first or second place according to quantitative metrics at 1K. Thus, our method helps the most at modern sensor resolutions but also behaves in a graceful way as resolution is lowered.

Methods	LPIPS ↓	FID ↓
MEDFE + SR / Ours	0.05170 / 0.04442	33.97 / 21.81
EdgeConnect + SR / Ours	0.05017 / 0.04350	35.06 / 21.77
Deepfillv2 + SR / Ours	0.05295 / 0.04349	32.87 / 20.50

Table 1. A comparison for older models between results upsampled by Real-ESRGAN and results upsampled by our proposed method.

Methods	LPIPS ↓	FID ↓		P-IDS ↑		U-IDS ↑		User Pref. ↑
		Full	Patch	Full	Patch	Full	Patch	
CoMoD	0.0598/0.0617	20.76/23.10	13.99/19.26	21.57/18.88	19.85/13.89	10.10/8.70	7.55/4.54	22/21
MADF	0.0548/0.0564	20.02/22.24	15.21/21.76	15.96/12.85	14.33/10.75	5.05/3.53	3.92/2.60	13/10
ProFill	0.0631/0.0574	20.64/22.83	16.52/22.37	16.37/13.97	12.80/10.59	5.75/3.94	2.86/2.66	22/14
LaMa	0.0509/0.0537	17.10/18.43	11.82/20.05	23.77/20.72	20.73/13.96	12.04/9.33	8.05/4.13	59/36
Ours	0.0522/0.0525	16.01/17.88	10.23/12.41	23.62/22.88	22.83/20.82	11.95/11.51	9.78/9.87	84/119

Table 2. A comparison study with the state-of-the-art methods at lower resolution of 1K / 2K corresponding to older camera sensors released approximately *two-and-a-half* and *two decades ago*, respectively [2,1]. The top 3 methods are colored: 1, 2, 3.

8 Additional Ablation Studies

8.1 Inpainting on Cropped Images vs. Full Images

Here we study whether it will be helpful if deep inpainting models are run on cropped patches centered around the hole rather than on the full images. The cropped patches are generated by using the auto crop mechanism discussed previously. As shown in Table 3, we observe that the results generated from full image inpainting slightly outperforms the cropped variant. We believe that the main reason for this is that the holes used in our experiments are sufficiently large, which also leads to fairly large cropped patch. Thus, inpaintings on the cropped patches will need to go through a similar amount of subsequent upsampling as inpainting on the full images, however, full images have relatively more context. Due to our observations in this study, we report the inpainting results from deep inpainting models running on the full images in the main manuscript.

Methods	LPIPS ↓	FID ↓	P-IDS ↑	U-IDS ↑
CoModGAN [15] on Crop	0.05244	25.05	16.68	5.46
CoModGAN [15] on Full	0.05099	24.8	14.72	4.47
MADF [16] on Crop	0.04774	28.21	4.52	0.50
MADF [16] on Full	0.04773	23.62	10.48	2.14
ProFill [13] on Crop	0.04925	24.34	10.32	2.72
ProFill [13] on Full	0.04783	24.25	11.35	2.26
LaMa [6] on Crop	0.04600	19.19	17.03	5.43
LaMa [6] on Full	0.04588	19.20	17.24	5.62

Table 3. Quantitative comparison between inpainting methods running on cropped patches vs. full images.

8.2 LaMa Inference on 2K Images

The concurrent work LaMa [6] mentions that their model can run directly on higher-resolution 2K images than the 256x256 images that it was trained on. We found that for our experiments, the best high-resolution inpainting results are obtained by running Lama [6] by resizing down to a maximum size of 512×512 while preserving aspect ratio combined with a larger scale upsampling by Real-ESRGAN [8]. In Table 4, we show a high-resolution quantitative comparison between LaMa [6] computing 2K output with a $2\times$ SR and LaMa [6] computing 512×512 output with a $8\times$ SR. The results show that LaMa [6] computed on 512×512 with a $8\times$ SR outperforms the alternative on most metrics. A typical visual comparison between the two options can be found in Figure 2. In general, we observe that LaMa [6] computed at 512 resolution generates both better texture and more coherent structure reconstruction than the alternative. Therefore, based on our quantitative and qualitative investigations, we report the results from LaMa [6] computing at 512 resolution in the main manuscript.

Methods	LPIPS ↓	FID ↓	P-IDS ↑	U-IDS ↑
LaMa [6] on 2K	0.03982	25.53	5.18	1.18
LaMa [6] on 512	0.04588	19.20	17.24	5.62

Table 4. Comparison between LaMa [6] direct computing on 2K resolution images with $2\times$ super resolution and LaMa [6] computing on 512 images with $8\times$ super resolution. Both cases use the Real-ESRGAN [8] for super resolution step.

8.3 Comparison with Bicubic Upsampling

The most naive way to upsample the inpainted outputs from deep network is bicubic upsampling. In this section, for the LaMa [6] model, we compare bicubic upsampling with Real-ESRGAN [6] both qualitatively and quantitatively. The visual results in Figure 3 shows a trade-off that the bicubic upsampling approach generates very blurry results compared to Real-ESRGAN [6] outputs, while Real-ESRGAN [6] sometimes produces slightly sharper boundary around the hole. Quantitatively, Table 5 shows that Real-ESRGAN [6] outperforms bicubic upsampling. Thus, we chose Real-ESRGAN [6] over bicubic upsampling for upsampling the competing methods.

8.4 Reference-Based Super Resolution

To upsample the outputs of competing methods, we chose Real-ESRGAN [6] since it is the state-of-the-arts super resolution algorithm and is also robust to visual artifacts in the natural image inputs. We also wondered whether reference-based super resolution could do a better job at upsampling the inpainting outputs of baselines. To answer this question, we tried a state-of-the-arts reference-based

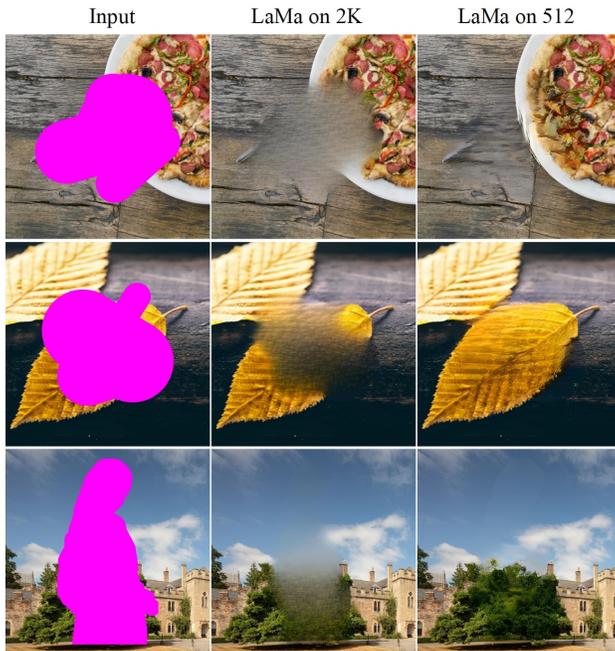


Fig. 2. Visual comparison between LaMa [6] run on 2K images with $2\times$ SR and LaMa [6] run on maximum size of 512×512 images with $8\times$ SR.

Methods	LPIPS ↓	FID ↓	P-IDS ↑	U-IDS ↑
+ Bicubic Upsampling	0.05043	23.66	16.46	5.14
+ Real-ESRGAN [8]	0.04588	19.20	17.24	5.62

Table 5. Quantitative comparison between upsampled inpainting results using bicubic upsampling and Real-ESRGAN [8].

super resolution method named TTSR [10]. In the reference-based SR setting, we feed inpainted images with a maximum size of 512×512 (maintaining aspect ratio) as inputs and the 2K original image with hole as the reference images, generate the $4\times$ upsampled outputs at 2K resolution, and finally upsample the outputs $2\times$ to 4K resolution with Real-ESRGAN [8]. Both methods share the same low resolution inpainted images computed from LaMa [6].

The current reference-based super resolution tasks mainly focus on upsampling images from like 128 to 512, while our starting resolution is 512. When we try to upsample images from 512 to 2048 with TTSR model [10], it requires approximately 118 GB per image, which cannot fit in available GPU memory. This leaves us the only option to run the inference code on CPU. Due to the extremely slow computation time, we did this “TTSR vs. Real-ESRGAN” com-

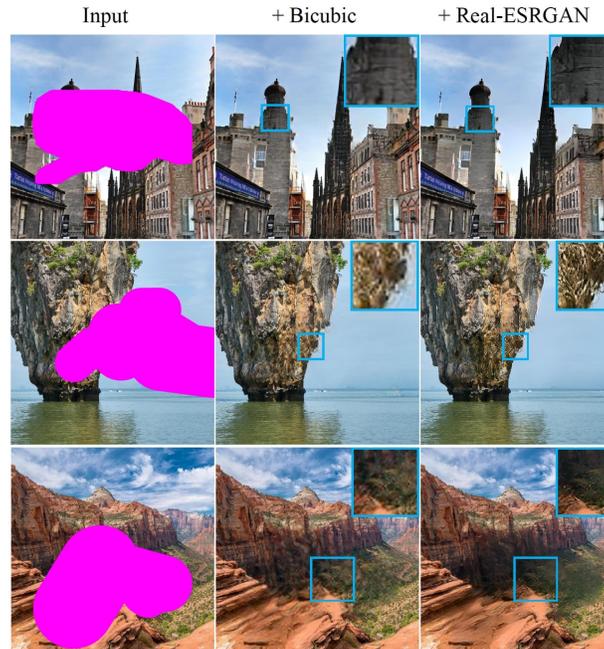


Fig. 3. Qualitative comparison for the LaMa [6] model between upsampled inpainting results using bicubic upsampling and Real-ESRGAN [8].

Methods	LPIPS ↓	FID ↓	P-IDS ↑	U-IDS ↑
+ TTSR [10]	0.04391	25.93	3.09	1.03
+ Real-ESRGAN [8]	0.04571	25.03	6.33	1.66

Table 6. Quantitative comparison between upsampled inpainting results using TTSR [8] and Real-ESRGAN [8].

parison study on a subset of 650 randomly sampled test images, which cost around 3 days computation time on a AMD Ryzen Threadripper 3960X 24-Core CPU. As shown in Table 6, the quantitative results show that Real-ESRGAN [8] is better than TTSR [10] in terms of FID and P/U-IDS and slightly worse in terms of LPIPS. Qualitatively, we observe that TTSR tends to consistently generate bumpy artifacts that are visually more unnatural than the Real-ESRGAN [8] outputs, as shown in Figure 4. Therefore, based on the quality comparison and computation feasibility, we use Real-ESRGAN [8] as the upsampling method for the competing methods in our paper.

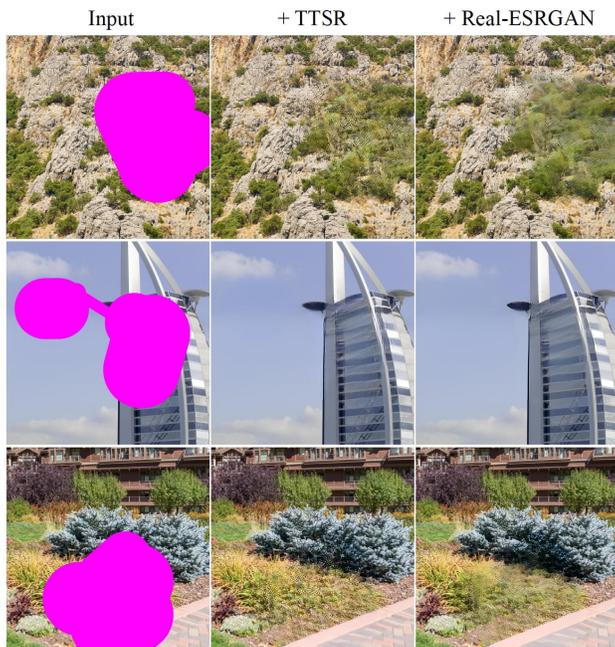


Fig. 4. Qualitative comparison between upsampled inpainting results using TTSR [8] and Real-ESRGAN [8].

8.5 Guided PatchMatch using RGB Images

While our proposed method uses depth, structure map, and panoptic segmentation computed as the guides for PatchMatch, we also evaluate the inpainting quality when using the RGB image produced by the deep inpainting backbone as the guide directly. As shown in Figure 5, we observe that the RGB guided PatchMatch tend to produce very blurry results, and thus make the texture visually unnatural. The quantitative evaluation shown in Table 7 also indicates that the RGB guided PatchMatch outputs do not achieve very good results. Thus, we decided not to incorporate the RGB guided PatchMatch as an option in our curation stage. More discussion of guide choices can be found in section 3.2 in the main manuscript.

Methods	LPIPS ↓	FID ↓	P-IDS ↑	U-IDS ↑
RGB Guide	0.04112	24.65	16.39	5.41
Our Full Model	0.04156	18.74	22.46	10.70

Table 7. Quantitative comparison between using RGB guided PatchMatch and our full model.

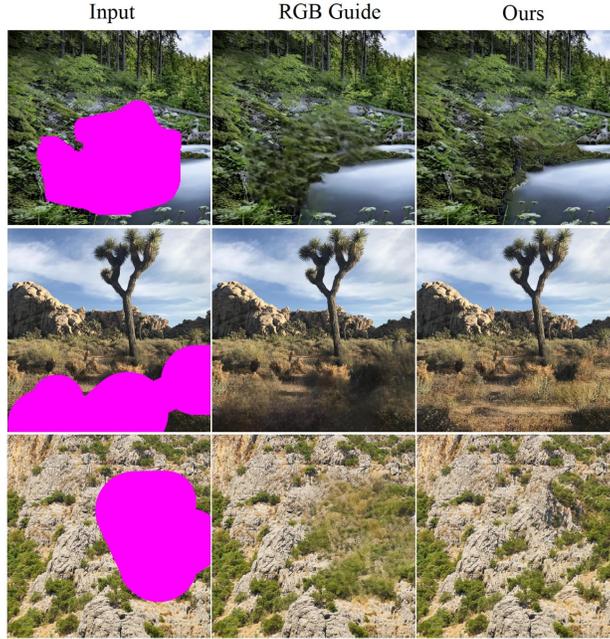


Fig. 5. Qualitative comparison between using RGB guided PatchMatch and our full model.

8.6 All Metrics

As promised in the main manuscript, we show all the evaluation metrics for the comparison with competing methods in Table 8. More discussion of the results can be found in Section 4.3 in the main manuscript, and discussion of evaluation metrics can be found in Section 3 in the supplemental material.

Methods	LPIPS ↓	PSNR ↑	SSIM ↑	FID ↓		P-IDS ↑		U-IDS ↑		User Pref. ↑	
				Full	Patch	Full	Patch	Full	Patch	Full Image	Boundary Patch
EdgeConnect [4]	0.05017	21.54	0.7831	35.06	41.05	0.04	4.56	0.00	0.55	-	-
Deepfillv2 [12]	0.05295	21.46	0.7862	32.87	36.06	5.54	5.47	1.35	0.84	-	-
MEDFE [3]	0.05170	22.03	0.7858	33.97	60.87	0.48	2.23	0.00	0.26	-	-
HiFill [11]	0.05213	20.82	0.7416	34.39	31.74	4.15	5.20	0.75	0.97	-	-
CoModGAN [15]	0.05099	21.63	0.7869	24.81	32.08	14.72	7.01	4.47	1.51	28	17
MADF [16]	0.04773	22.84	0.8047	23.62	33.21	10.48	6.81	2.14	1.48	6	12
ProFill [13]	0.04783	22.59	0.7990	24.25	31.26	11.35	6.89	2.26	1.31	10	16
LaMa [6]	0.04588	23.11	0.8111	19.20	35.95	17.24	6.86	5.62	1.38	28	22
GPFill (Ours)	0.04156	22.73	0.7969	18.74	15.63	22.46	19.77	10.70	10.22	128	133

Table 8. A comparison study with the state-of-the-art inpainting methods. The top 3 methods are colored: 1, 2, 3.

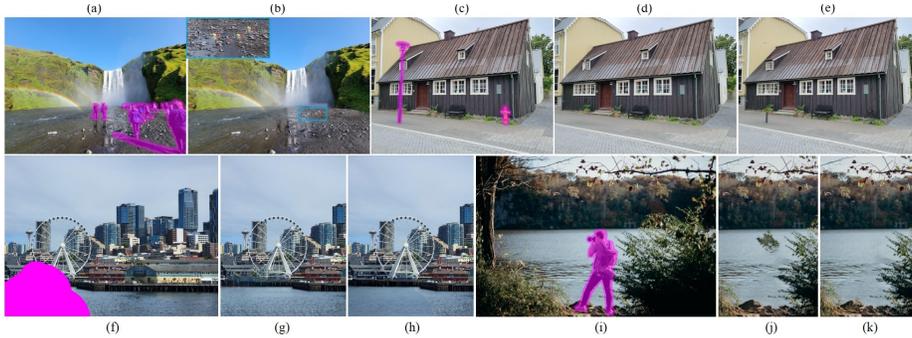


Fig. 6. Limitations of our method. In (a) and (b), undesirable repetitions of the salient yellow and red objects occur in the inpainted region. In (c) and (d), structures under perspective can be broken for the window at the top of the house, while LaMa [6] (e) reconstructs better structure at the window region but has visual artifacts in other regions. In (f) and (g), our patch-based synthesis method fails to inpaint the Ferris wheel, since the image lacks good reference content to directly copy from to fill the hole and requires hallucinating new structure. In this case, LaMa [6] builds some new structure to better fill the Ferris wheel but produce over-smooth pixels in the water region, in (h). In (f) - (k), the curation network sometimes does not pick the best option from eight Guided PatchMatch outputs. In this case, the curation network picks (j) instead of (k), and thus a human manually picking could help for this case. The last section of our main manuscript discusses ways these limitations could potentially be mitigated in future work.

9 Failure Cases

As discussed in the last section of our main manuscript, our method has some limitations. In this section, we provide several visual demonstration in Figure 6 to show the limitations we mentioned in the main paper, and more discussion can be found in the caption of Figure 6.

10 More Qualitative Results

We show more inpainting results at 4K or above resolution in Figure 7, Figure 8, Figure 9, and Figure 10 in the following pages.

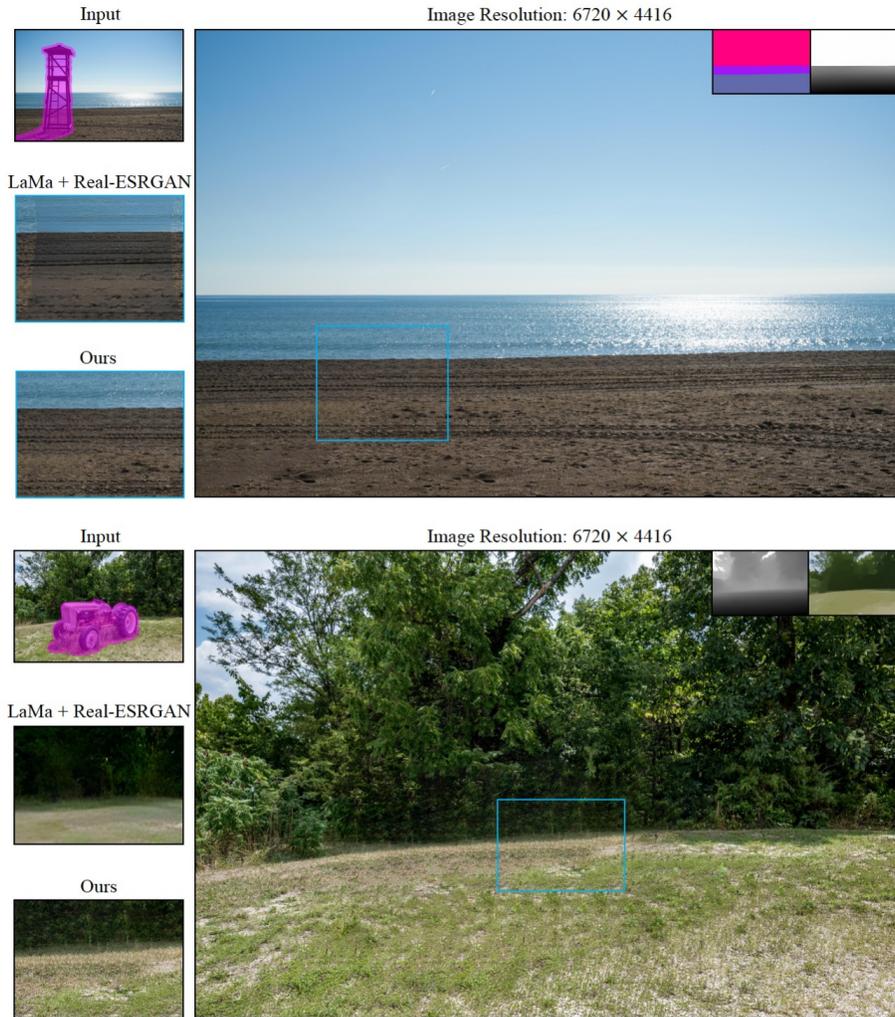


Fig. 7. In each of the two examples, the top left is the input image with a real object mask, the right giant image is the inpainted result from our method, the other two images on the bottom left are the zoom in insets for the closet competing method LaMa [6] upsampled by Real-ESRGAN [8] and ours, where the zoom in location are indicated by the bounding box on the right image. The guides used for each image is shown at the top right corner of the inpainted result.

References

1. Canon: Canon camera museum: Powershot a75 (2022), <https://global.canon/en/c-museum/product/dcc495.html> 4, 5

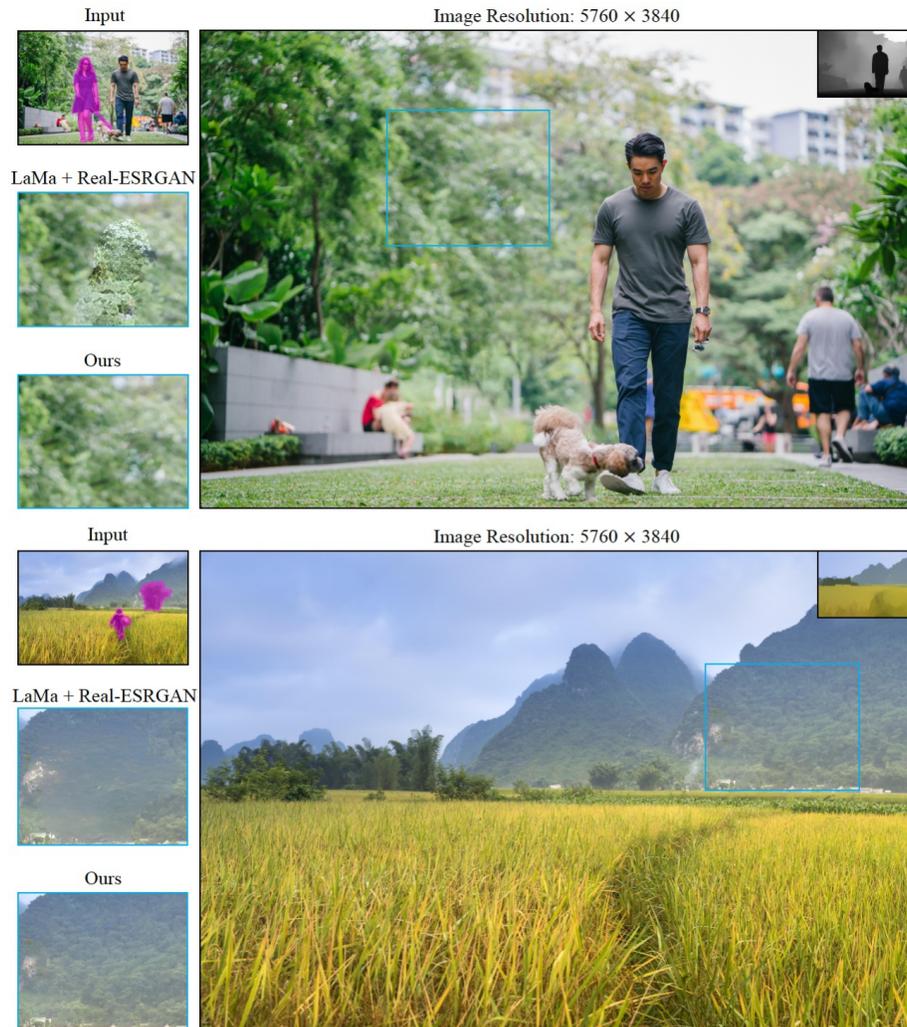


Fig. 8. In each of the two examples, the top left is the input image with a real object mask, the right giant image is the inpainted result from our method, the other two images on the bottom left are the zoom in insets for the closet competing method LaMa [6] upsampled by Real-ESRGAN [8] and ours, where the zoom in location are indicated by the bounding box on the right image. The guides used for each image is shown at the top right corner of the inpainted result.

2. van Hall, D.: Digital kameramuseum: Canon powershot 600n (1997) (2022), <https://www.digitalkameramuseum.de/en/cameras/item/canon-powershot-600n> 4, 5
3. Liu, H., Jiang, B., Song, Y., Huang, W., Yang, C.: Rethinking image inpainting via a mutual encoder-decoder with feature equalizations. arXiv preprint arXiv:2007.06929 (2020) 10

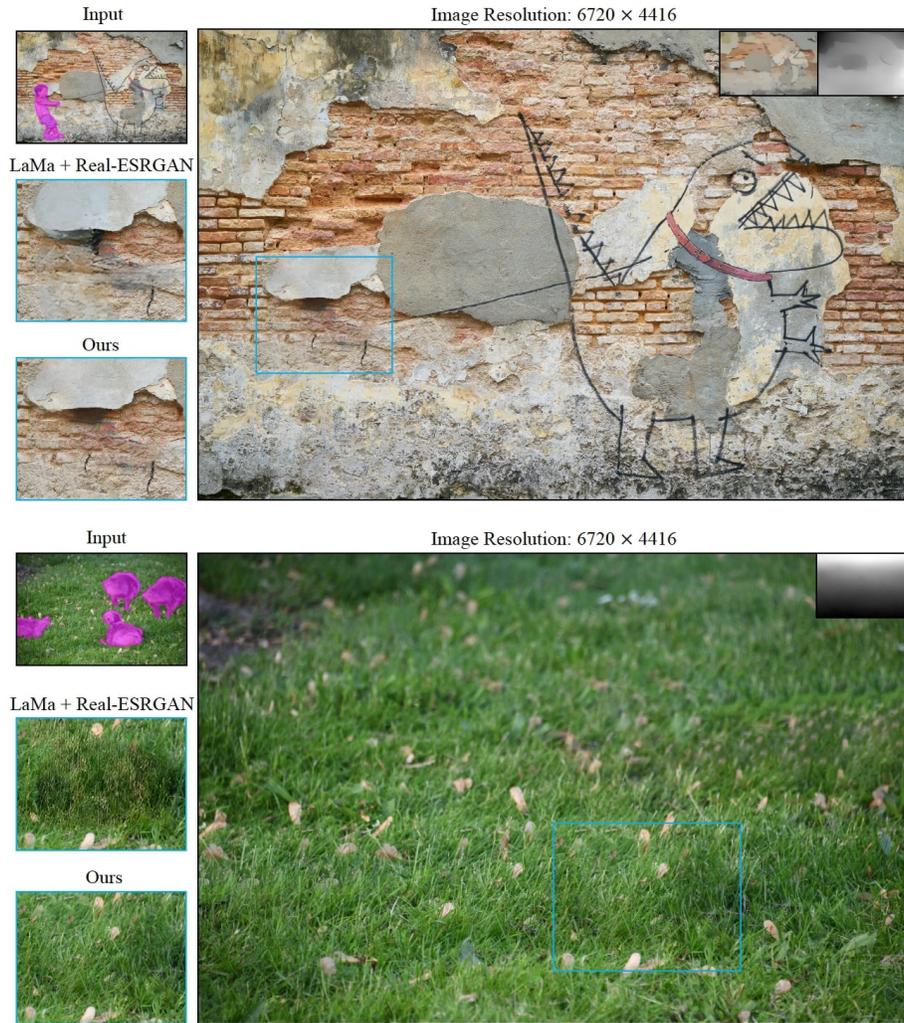


Fig. 9. In each of the two examples, the top left is the input image with a real object mask, the right giant image is the inpainted result from our method, the other two images on the bottom left are the zoom in insets for the closet competing method LaMa [6] upsampled by Real-ESRGAN [8] and ours, where the zoom in location are indicated by the bounding box on the right image. The guides used for each image is shown at the top right corner of the inpainted result.

4. Nazeri, K., Ng, E., Joseph, T., Qureshi, F.Z., Ebrahimi, M.: Edgeconnect: Generative image inpainting with adversarial edge learning. arXiv preprint arXiv:1901.00212 (2019) 10
5. Parmar, G., Zhang, R., Zhu, J.Y.: On buggy resizing libraries and surprising subtleties in fid calculation. arXiv preprint arXiv:2104.11222 (2021) 3



Fig. 10. In each of the two examples, the top left is the input image with a real object mask, the right giant image is the inpainted result from our method, the other two images on the bottom left are the zoom in insets for the closet competing method LaMa [6] upsampled by Real-ESRGAN [8] and ours, where the zoom in location are indicated by the bounding box on the right image. The guides used for each image is shown at the top right corner of the inpainted result.

- Suvorov, R., Logacheva, E., Mashikhin, A., Remizova, A., Ashukha, A., Silvestrov, A., Kong, N., Goka, H., Park, K., Lempitsky, V.: Resolution-robust large mask inpainting with fourier convolutions. WACV: Winter Conference on Applications of Computer Vision (2022) 5, 6, 7, 8, 10, 11, 12, 13, 14, 15

7. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016) [3](#)
8. Wang, X., Xie, L., Dong, C., Shan, Y.: Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1905–1914 (2021) [6](#), [7](#), [8](#), [9](#), [12](#), [13](#), [14](#), [15](#)
9. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **13**(4), 600–612 (2004) [3](#)
10. Yang, F., Yang, H., Fu, J., Lu, H., Guo, B.: Learning texture transformer network for image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5791–5800 (2020) [7](#), [8](#)
11. Yi, Z., Tang, Q., Azizi, S., Jang, D., Xu, Z.: Contextual residual aggregation for ultra high-resolution image inpainting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7508–7517 (2020) [10](#)
12. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Free-form image inpainting with gated convolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4471–4480 (2019) [10](#)
13. Zeng, Y., Lin, Z., Yang, J., Zhang, J., Shechtman, E., Lu, H.: High-resolution image inpainting with iterative confidence feedback and guided upsampling. In: European Conference on Computer Vision. pp. 1–17. Springer (2020) [1](#), [5](#), [10](#)
14. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018) [3](#)
15. Zhao, S., Cui, J., Sheng, Y., Dong, Y., Liang, X., Chang, E.I., Xu, Y.: Large scale image completion via co-modulated generative adversarial networks. In: International Conference on Learning Representations (ICLR) (2021) [3](#), [5](#), [10](#)
16. Zhu, M., He, D., Li, X., Li, C., Li, F., Liu, X., Ding, E., Zhang, Z.: Image inpainting by end-to-end cascaded refinement with mask awareness. *IEEE Transactions on Image Processing* **30**, 4855–4866 (2021) [5](#), [10](#)