Hybrid Image-based Rendering for Free-view Synthesis

SIDDHANT PRAKASH, Université Côte d'Azur and Inria, France
THOMAS LEIMKÜHLER, Université Côte d'Azur and Inria, France
SIMON RODRIGUEZ, Université Côte d'Azur and Inria, France
GEORGE DRETTAKIS, Université Côte d'Azur and Inria, France

CCS Concepts: • **Computing methodologies → Image-based rendering**; *Image processing*; Rasterization; Texturing.

Additional Key Words and Phrases: interactive rendering, image harmonization, uncertainty

ACM Reference Format:
Siddhant Prakash, Thomas Leimkühler, Simon Rodriguez, and George Drettakis. 2021. Hybrid Image-based Rendering for Free-view Synthesis. *Proc. ACM Comput. Graph. Interact. Tech.* 4, 1 (May 2021), 5 pages. https://doi.org/10.1145/3451260

1 COMPUTING THE HARMONIZATION MASK

Akin to [Agarwala et al. 2004] we define a cost function $C(l)$ for label l, where l is diffuse or view-dependent, i.e., $l \in \{\text{diff}, \text{spec}\}$, with a *unary cost* C_u over all pixels p and an *interaction cost* C_i over all pairs of pixels p and q in a 4-neighborhood:

$$C(l) = \sum_{p} C_u(p, l_p) + \sum_{p,q} C_i(p, q, l_p, l_q). \tag{1}$$

This unary cost term C_u identifies view-dependent regions while the interaction cost term C_i tries to find good seams to minimize visible artifacts. We define

$$C_u(p, l_p) = \begin{cases} \infty & \text{if } l_p = \text{spec} \wedge p \notin R \\ \exp(\sigma_c + \lambda \Delta_c) & \text{if } l_p = \text{spec} \wedge p \in R \\ \exp(1 - (\sigma_c + \lambda \Delta_c)) & \text{if } l_p = \text{diff}. \end{cases}$$

The unary cost term considers the global color variance σ_c, the intensity difference between original and diffuse harmonized image Δ_c, and a spatial confidence region R. Ideally, we want the harmonized image to be similar to the original image. High color variance indicates specular regions [Lin et al. 2002]. Similarly, a high intensity difference between the original and diffuse harmonized images indicates highlight suppression. The parameter λ balances the relative importance of these two goals, and we set it to 2 in all our experiments. The spatial confidence margin penalizes specularities near the image edges to avoid re-introducing vignetting artifacts.

We define the interaction cost

Authors' addresses: Siddhant Prakash, Université Côte d'Azur and Inria, Sophia-Antipolis, France, siddhant.prakash@inria.fr; Thomas Leimkühler, Université Côte d'Azur and Inria, Sophia-Antipolis, France, thomas.leimkuhler@inria.fr; Simon Rodriguez, Université Côte d'Azur and Inria, Sophia-Antipolis, France, simon.rodriguez@inria.fr; George Drettakis, Université Côte d'Azur and Inria, Sophia-Antipolis, France, george.drettakis@inria.fr.

$$C_i(p, q, l_p, l_q) = \begin{cases} \exp\left(\|c_p - c_q\|^2\right) & \text{if } l_p = l_q \\ \exp\left(1 - \|c_p - c_q\|^2\right) & \text{if } l_p \neq l_q, \end{cases}$$

where c denote the colors of the original image. If the color difference of neighboring pixels is high we penalize assigning same labels to both pixels since the pixels may constitute a true edge that we want to preserve.

2 COMPARISON CODE

We used published implementations for previous work. Specifically, for [Hedman et al. 2018]: https://sibr.gitlabpages.inria.fr, for [Mildenhall et al. 2020]: https://github.com/bmild/nerf, for [Riegler and Koltun 2020] https://github.com/intel-isl/FreeViewSynthesis, and for the perceptual error metric E-LPIPS [Kettunen et al. 2019]: https://github.com/mkettune/elpips.

For NeRF, we had to select a subset of cameras, since if we used all of them the results were unusable. We tried different combinations of cameras, and kept the one with the best visual result.

3 ADDITIONAL RESULTS AND PREPROCESSING STATISTICS

3.1 Runtime Comparisons

We compare the total runtime of different algorithms on the 2 machines described in the main paper for a given dataset in Table 1.

Table 1. Comparison of total runtime of different algorithms over a sequence of 100 frames on 2 different machines with 2 datasets for a rendering resolution of 1280x720.

IBR Algorithms	Desktop		Laptop	
	Hugo	Dr Johnson	Hugo	Dr Johnson
ULR	0.06 *ms*	7.32 *ms*	0.01 *ms*	14.45 *ms*
InsideOut	8.37 *ms*	12.93 *ms*	10.63 *ms*	18.24 *ms*
Deep Blending	64 *ms*	68.78 *ms*	79.89 *ms*	98.88 *ms*
Ours	21.11 *ms*	31.59 *ms*	27.12 *ms*	52.03 *ms*

3.2 Test Image Set

We show all images used for testing with ground truth from the Ponche dataset in Figure 1 and from the Synthetic Attic dataset in Figure 2.

3.3 Pre-processing Time

We also report the pre-processing time with a breakdown of each component in Table 2. The pre-processing time varies based on number of images and resolution of original images. Hence, we provide the numbers for 3 different datasets to give a better insight into pre-processing time.

3.4 Dataset Statistics

We provide additional information on our datasets reporting total number of images, input image resolutions, and total processing time in Table 3. All input images are Low Dynamic Range (LDR) captured using commercial hand-held devices (typical SLR Cameras) in an unconstrained or casual acquisition process. We resize the input images to 1920px in the dominant resolution maintaining the aspect ratio to fit all data on the GPU VRAM. We do not support datasets which scale above the GPU VRAM limit but a future extension by using streaming architectures is an interesting follow-up to our method.

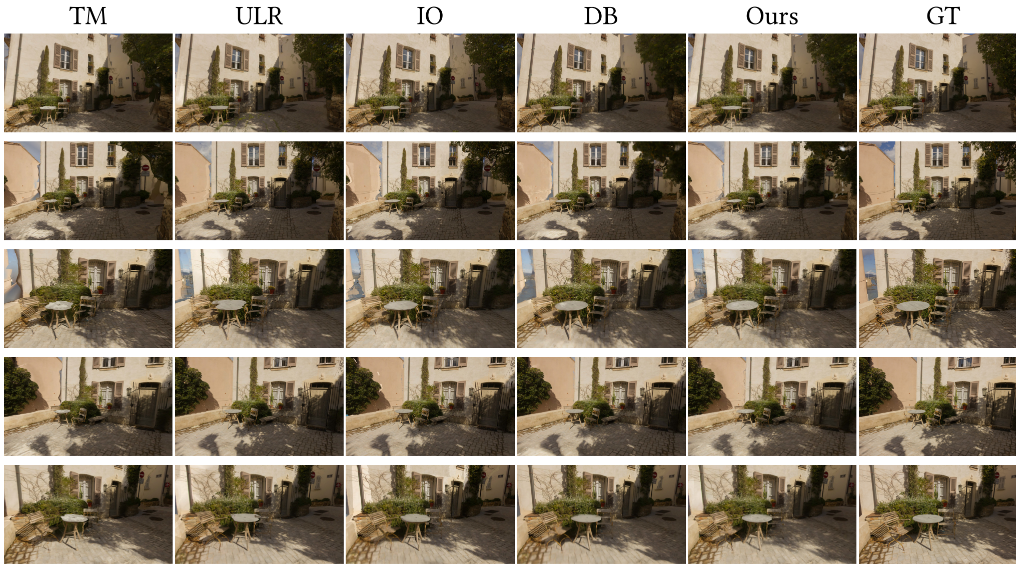

Fig. 1. Held-out test images with their ground truth input image from Ponche. TM: Textured Mesh [Reality 2018]; ULR: Unstructured Lumigraph [Buehler et al. 2001]; IO: Inside Out [Hedman et al. 2016]; DB: Deep Blending [Hedman et al. 2018]; GT: Ground Truth.

Table 2. Pre-processing time for Hugo (24 images, 3216X2136), Creepy Attic (246 images, 1228X816), and Library (222 images, 3552X2000) on a system with Intel Xeon Gold 5218 2.30GHz Processor and Quadro RTX 5000 GPU.

Component	Hugo	Creepy Attic	Library
Harmonization Step 1	4m	6m	38m
Harmonization Step 2	9m	14m	1h 58m
Spec Mask Texturing	26s	1m 13s	1m 21s
Tiling and Storing	16s	1m 33s	1m 37s
Total	14m	26m	2h 40m

Table 3. Statistics for all datasets presented in the paper.

Dataset	#Images	Image Res.	Preprocess Time (Total)
Hugo	24	$3216X2136$	14 min
Ponche	50	$2016X1344$	19 min
Library	222	$3552X2000$	2h 40 min
Playroom	226	$1264X832$	25 min
Creepy Attic	246	$1228X816$	26 min
Dr Johnson	264	$1296X864$	31 min
Syn. Attic	283	$1920X1080$	57 min
Train	301	$1920X1080$	2h 8 min
Salon	344	$3000X2000$	3h 45 min

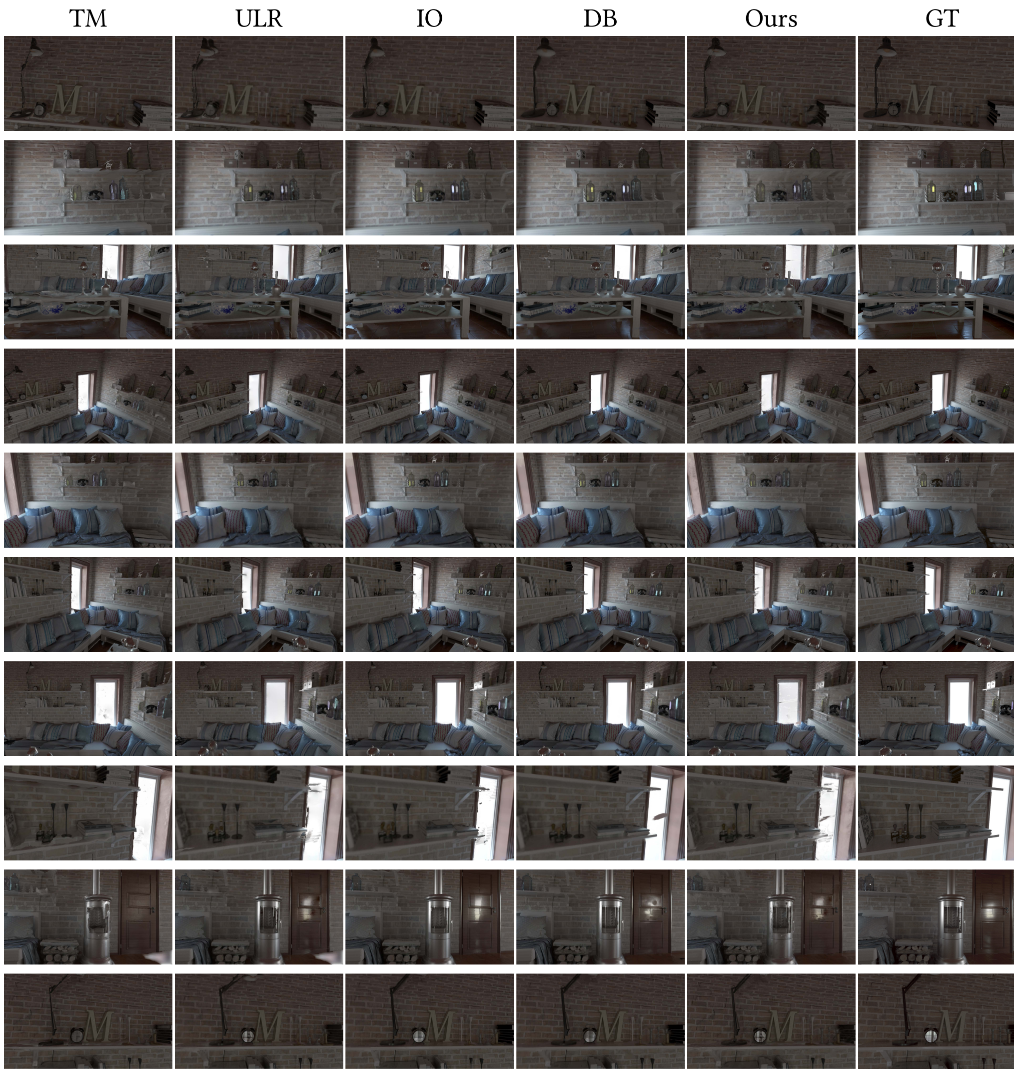

Fig. 2. Novel view test images with their path traced ground truth from Synthetic Attic. TM: Textured Mesh [Reality 2018]; ULR: Unstructured Lumigraph [Buehler et al. 2001]; IO: Inside Out [Hedman et al. 2016]; DB: Deep Blending [Hedman et al. 2018]; GT: Ground Truth.

REFERENCES

Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael Cohen. 2004. Interactive Digital Photomontage. In *ACM SIGGRAPH 2004 Papers* (Los Angeles, California) *(SIGGRAPH '04)*. Association for Computing Machinery, New York, NY, USA, 294–302. https://doi.org/10.1145/1186562.1015718

Chris Buehler, Michael Bosse, Leonard McMillan, Steven Gortler, and Michael Cohen. 2001. Unstructured Lumigraph Rendering. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*. Association for Computing Machinery, New York, NY, USA, 425–432. https://doi.org/10.1145/383259.383309

Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. 2018. Deep Blending for Free-Viewpoint Image-Based Rendering. *ACM Trans. Graph.* 37, 6, Article 257 (Dec. 2018), 15 pages. https://doi.org/

10.1145/3272127.3275084

Peter Hedman, Tobias Ritschel, George Drettakis, and Gabriel Brostow. 2016. Scalable inside-out image-based rendering. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 231.

Markus Kettunen, Erik Härkönen, and Jaakko Lehtinen. 2019. E-LPIPS: Robust Perceptual Image Similarity via Random Transformation Ensembles. arXiv:1906.03973 [cs.CV]

Stephen Lin, Yuanzhen Li, Sing Bing Kang, Xin Tong, and Heung-Yeung Shum. 2002. Diffuse-Specular Separation and Depth Recovery from Image Sequences. In *Computer Vision — ECCV 2002*, Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 210–224.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 405–421.

Capturing Reality. 2018. RealityCapture reconstruction software. https://www.capturingreality.com/Product.

Gernot Riegler and Vladlen Koltun. 2020. Free View Synthesis. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 623–640.