FAST LOCAL NEURAL REGRESSION FOR LOW-COST GLOBAL ILLUMINATION DENOISING

Arturo Salmi arturo.salmi@imgtec.com **Szabolcs Cséfalvay** szabolcs.csefalvay@imgtec.com **James Imber** james.imber@imgtec.com

Abstract

Recent path tracing denoisers are benefitting from a greater availability of AI acceleration hardware by using neural networks to achieve excellent visual quality at low sample per pixel (spp) budgets. Although some of these "neural" denoisers can already achieve real-time performance on commodity hardware, neural networks are often less computationally efficient than traditional hand-crafted methods exploiting domain knowledge. We show that a robust, high-quality hand-engineered spatial denoiser, when augmented with a neural network, can achieve state-of-the-art results at a fraction of the execution time of conventional neural network-based methods.

Fast Local Regression for Denoising

Local linear models fit clean, reference signals x to noisy signals y, to obtain a denoised output I using a closed-form least-squares solver. Figure 1 shows how multiple input signals (guides) can be combined to denoise an input signal [1]. We simplify the mathematics of the local linear model [2] and use a neural network to generate improved guides that handle complex global lighting.

We construct signal $\boldsymbol{Y}_k \in \mathbb{R}^{(N\times 3)}$ and guide $\boldsymbol{X}_k \in \mathbb{R}^{(N\times(Q+1))}$ matrices for each window T_k centred around pixel k. Each row corresponds to a pixel location in T_k, while each column represents a different channel of the respective tensors. The first column in $\boldsymbol{X}_k$ always consists of 1s to model offsets.

$$\boldsymbol{X}_k = \begin{bmatrix} 1 & \boldsymbol{X}_{0,1} & \cdots & \boldsymbol{X}_{0,Q} \\ 1 & \boldsymbol{X}_{1,1} & \cdots & \boldsymbol{X}_{1,Q} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \boldsymbol{X}_{N-1,1} & \cdots & \boldsymbol{X}_{N-1,Q} \end{bmatrix}, \quad \boldsymbol{Y}_k = \begin{bmatrix} \boldsymbol{R}_0 & \boldsymbol{G}_0 & \boldsymbol{B}_0 \\ \boldsymbol{R}_1 & \boldsymbol{G}_1 & \boldsymbol{B}_1 \\ \vdots & \vdots & \vdots \\ \boldsymbol{R}_{N-1} & \boldsymbol{G}_{N-1} & \boldsymbol{B}_{N-1} \end{bmatrix}$$

We fit the parameter matrix $\boldsymbol{A}_k \in \mathbb{R}^{((Q+1)x3)}$ such that $\boldsymbol{Y}_k \approx \boldsymbol{X}_k\boldsymbol{A}_k$:

$$\boldsymbol{A}_k^* = \arg\min_{\boldsymbol{A}_k} \|\boldsymbol{X}_k\boldsymbol{A}_k - \boldsymbol{Y}_k\|_2^2 = \left(\boldsymbol{X}_k^T\boldsymbol{X}_k\right)^{-1}\boldsymbol{X}_k^T\boldsymbol{Y}_k$$

The solution $\boldsymbol{A}_k^*$ determines the denoised pixel value $\boldsymbol{I}_k$ as:

$$\boldsymbol{I}_k = x_k\boldsymbol{A}_k^*$$

x_k is the row of the guidance matrix $\boldsymbol{X}_k$ corresponding to the pixel k. Note that fitting multiple guide channels to multiple output channels is straightforward in our method. We make the following further improvements:

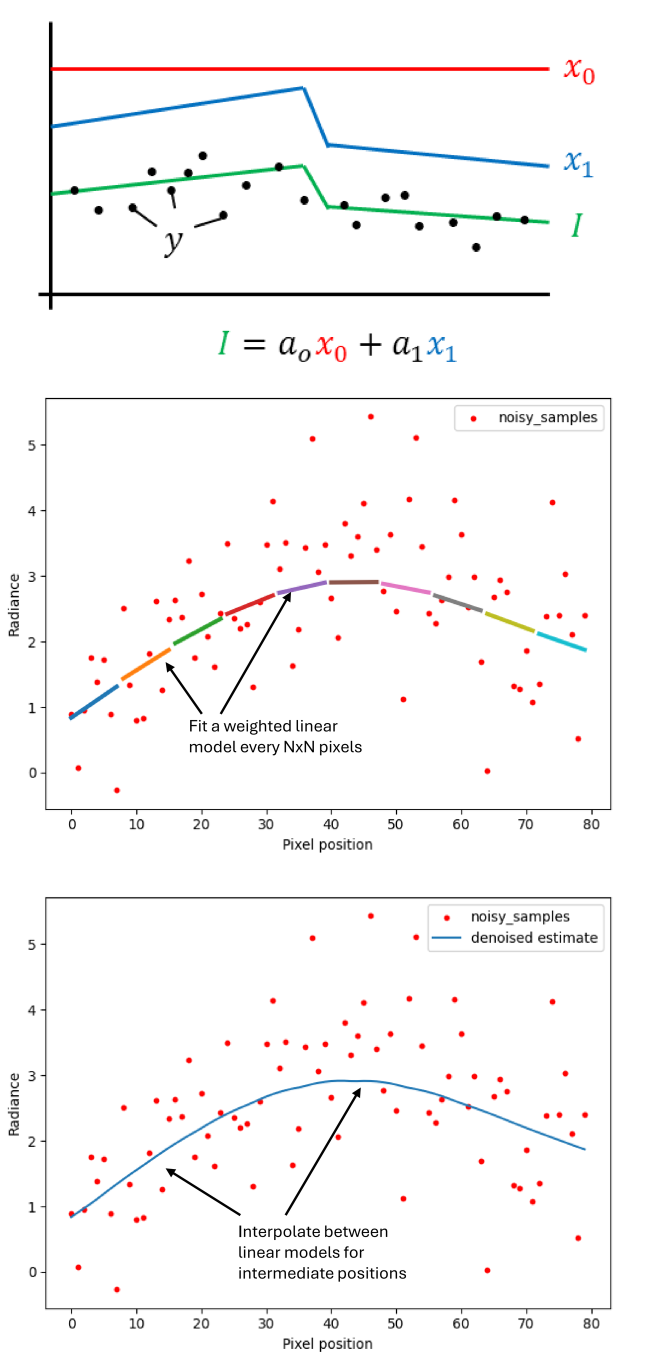

Figure 1: We fit multiple guide channels to noisy input lighting using a windowed local linear regressor [2].

- We exploit redundant computation between overlapping windows [2]. Blurring the outer product tensors with an efficient box filter yields exactly the per-pixel matrices $\boldsymbol{X}_k^T\boldsymbol{X}_k$ and $\boldsymbol{X}_k^T\boldsymbol{Y}_k$.
- We use a separable Gaussian blur to increase the contribution of pixels near the centre of the window (corresponds to weighted least squares [3]).
- We decompose the blur into downsampling the outer product tensor and Gaussian filtering. This reduces the kernel size significantly.

Our proposed Fast Local Regression (FLR) method can denoise a 1080p frame in just 0.64 ms when executed on an Nvidia RTX 2080Ti.

Enhancing Local Regression with Neural Networks

Regression-based filters can only adapt to structure present in the guides. Therefore, cast shadows and direct lighting are generally handled poorly by such methods.

To address this limitation, our Fast Local Neural Regression (FLNR) extends FLR by using a neural network to transform the input guides. For a given frame, the neural network is trained to produce enhanced guide images $\mathcal{X}'$ from the input guide channels $\mathcal{X}$ and the noisy input radiance $\mathcal{Y}$. Feeding input noisy light as an additional input allows the network to adapt the enhanced guides to maximise quality for given light conditions.

The network architecture is a simple U-Net with 4 encoder/decoder blocks and skip connections. Each block comprises two 3x3 convolution layers followed by a Leaky ReLU activation function. As the complexity of the task performed by the network is limited to enhancing the guides, the number of channels required in each layer is low, keeping the network lightweight.

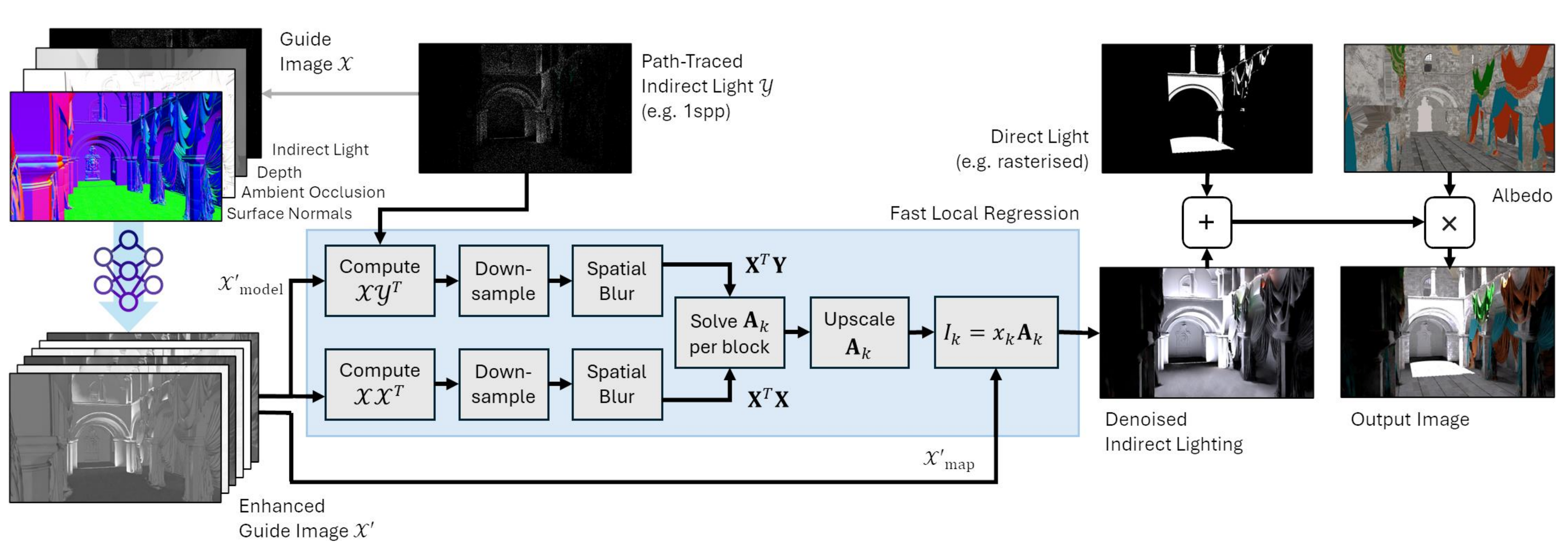

Figure 2: FLNR builds upon FLR to improve handling of complex lighting.

Choosing Input Guides

In common with previous literature, we also found surface normals and depth to be useful auxiliary inputs.

Furthermore, we found ambient occlusion to be a valuable input, as it provides the model with cast shadows information that might not be readily available in other guides

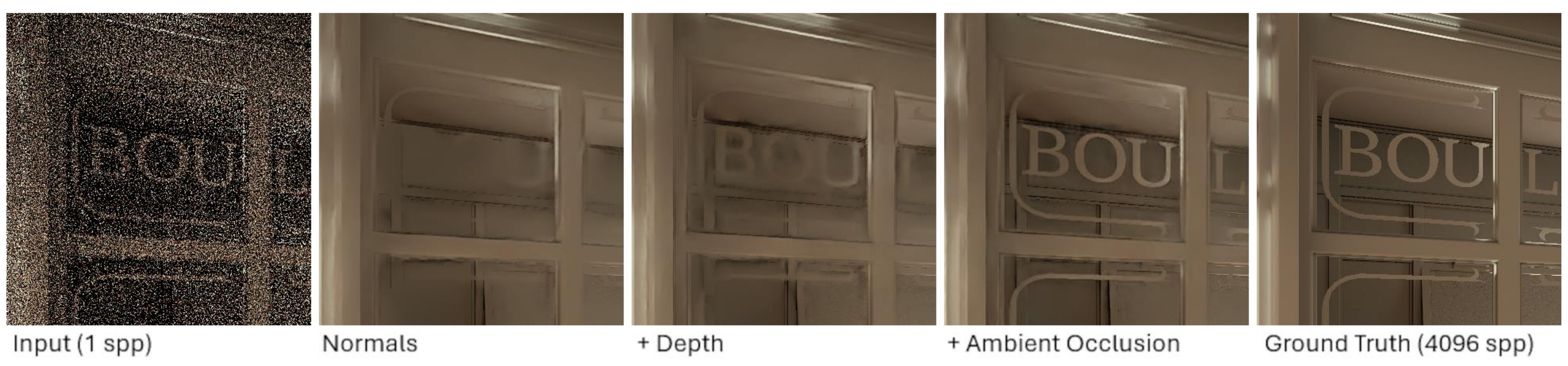

Figure 3: Ambient occlusion helps the filter restore the lost structural information while also improving the quality of denoised shadows.

Results

We compared our model against other benchmark methods on the task of denoising indirect diffuse light rendered at 1 spp. FLNR outperforms all the other benchmark methods across all the metrics used for evaluation. FLR also achieves competitive results.

Method	Normals	Geometry	AO	PSNR ↑	SSIM ↑	FLIP ↓	rMSE ↓	Runtime (ms) ↓
Guided Filter [1]	✓	✓		34.925	0.890	0.061	0.024	11.1*
SVGF [4]	✓	✓		25.924	0.540	0.153	0.070	10.950
BMFR [5]	✓	✓		33.194	0.834	0.075	0.028	1.731
NBG [6]	✓	✓		33.458	0.828	0.074	0.029	196.614
MKPCN [6]	✓	✓		26.933	0.603	0.125	0.057	170.401
OIDN [7]	✓			36.316	0.885	0.055	0.021	37.370
ODDN [8]	✓			34.418	0.813	0.076	0.026	7.313
FLR (Ours) w/o AO	✓	✓		35.334	0.898	0.055	0.023	-
FLR (Ours)	✓	✓	✓	35.863	0.897	0.052	0.021	**0.636**
FLNR (Ours)	✓	✓	✓	**37.238**	**0.904**	**0.046**	**0.018**	11.542

Unlike other neural-based methods, FLNR is capable of reconstructing accurate cast shadows without introducing chromatic noise or overblurring the output.

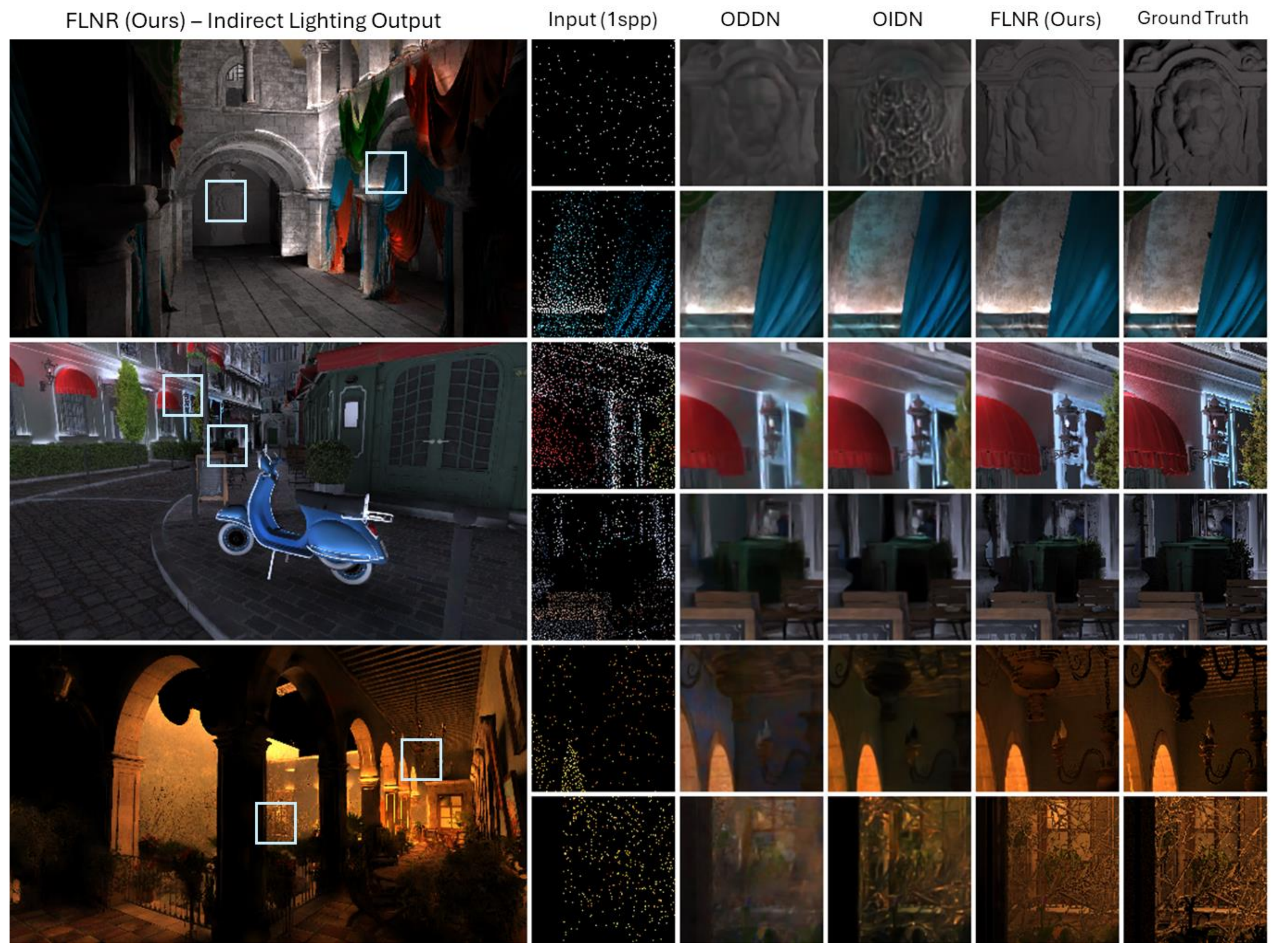

Figure 4: Visual comparison of albedo-modulated results for denoising of 1spp indirect lighting. Locations of crops are indicated on left.

Furthermore, our method is also capable of handling noisy direct light rendered using environment maps, area lights and multiple light sources.

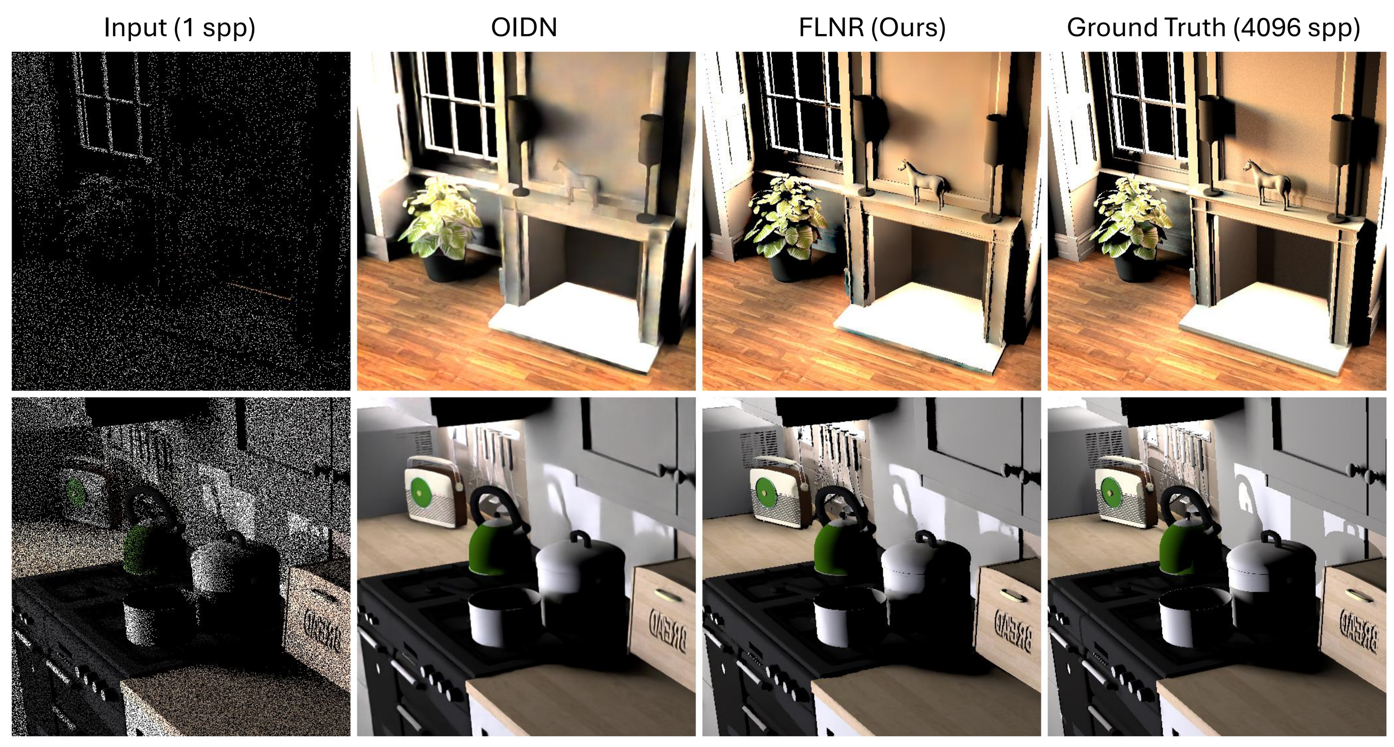

Figure 5: Our method is capable of handling noisy direct light from environment maps, area lights and multiple light sources.

FLNR visual quality scales well with increasing sampling rates, although for higher budgets (>16 spp) methods like OIDN [7] and ODDN [8] tend to perform better.

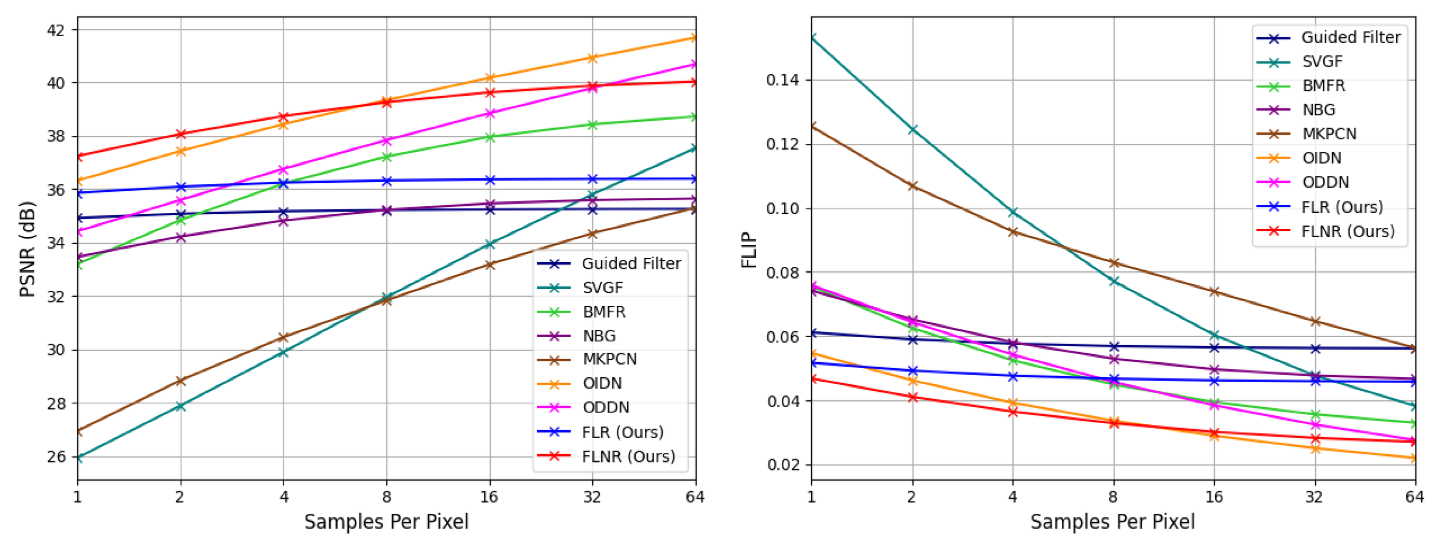

Figure 6: Evaluation of quality metrics for increasing spp budgets.

We design FLNR primarily as a spatial denoiser. Extending to temporal sequences can be done simply by accumulating temporally reprojected samples from previous frames (scan QR code below for an example).

Future Work

- Extend our method to handle non-diffuse surfaces.
- Improve temporal results by training the model on frame sequences.
- Further optimise the efficiency of our model by using a static compiler and quantising to lower bit depths.

References

[1] Pablo Bauszat, Martin Eisemann, and Marcus Magnor. 2011. Guided Image Filtering for Interactive High-quality Global Illumination. Computer Graphics Forum 30, 4 (2011), 1361–1368.

[2] Kaiming He and Jian Sun. 2015. Fast Guided Filter. (2015). arXiv:1505.00996.

[3] William S Cleveland. 1979. Robust locally weighted regression and smoothing scatterplots. Journal of the American statistical association 74, 368 (1979), 829–836.

[4] Christoph Schied, Anton Kaplanyan, Chris Wyman, Anjul Patney, Chakravarty R. Alla Chaitanya, John Burgess, Shiqiu Liu, Carsten Dachsbacher, Aaron Lefohn, and Marco Salvi. 2017. Spatiotemporal variance-guided filtering: real-time reconstruction for path-traced global illumination. In Proceedings of High Performance Graphics (Los Angeles, California) (HPG '17). Association for Computing Machinery, New York, NY, USA, Article 2, 12 pages.

[5] Matias Koskela, Kalle Immonen, Markku Mäkitalo, Alessandro Foi, Timo Viitanen, Pekka Jääskeläinen, Heikki Kultala, and Jarmo Takala. 2019. Blockwise Multi-Order Feature Regression for Real-Time Path-Tracing Reconstruction. ACM Trans. Graph.38, 5, Article 138 (jun 2019).

[6] Xiaoxu Meng, Quan Zheng, Amitabh Varshney, Gurprit Singh, and Matthias Zwicker. 2020. Real-time Monte Carlo Denoising with the Neural Bilateral Grid. In Eurographics Symposium on Rendering - DL-only Track, Carsten Dachsbacher and Matt Pharr (Eds.). The Eurographics Association.

[7] Attila T. Áfra. 2024. Intel®Open Image Denoise. https://www.openimagedenoise.org.

[8] Chakravarty R. Alla Chaitanya, Anton S. Kaplanyan, Christoph Schied, Marco Salvi, Aaron Lefohn, Derek Nowrouzezahrai, and Timo Aila. 2017. Interactive reconstruction of Monte Carlo image sequences using a recurrent denoising autoencoder. ACM Trans. Graph. 36, 4, Article 98 (jul 2017)

Video QR Code