Spectral Monte Carlo Denoiser

Mathieu Noizet, Robin Rouphael, Stéphanie Prévost, Hervé Deleau, Luiz Angelo Steffenel, Laurent Lucas

Spectral Monte Carlo Denoiser

M. NOIZET[1], R. ROUPHAEL[2], S. PRÉVOST[1], H. DELEAU[1], L.A. STEFFENEL[1], AND L. LUCAS[1]

[1] UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE, LICIIS, LRC DIGIT, France
[2] UNITED VISUAL RESEARCHERS, PARIS, FRANCE

SIGGRAPH 2024
DENVER+ 28 JUL — 1 AUG

Problem

Spectral Monte-Carlo rendering allows the creation of physically accurate virtual prototypes by:

- Computing global illumination in the spectral domain
- Taking into account wavelength-dependent phenomenons

But comes at the cost of significant computing time.

Efficient denoising methods could reduce this cost but:

- State of the art RGB denoising [1,2] is unable to reconstruct spectral renders (see Fig. 1)
- Spectral domain denoising remains unexplored

Figure 1: Visual comparison of different denoisers on Kitchen scene. From left to right: Noisy, OptiX, ours (λ+PCA+DS), Ground truth, and crops: OptiX (top), ours (bottom).

Related Work

Three relevant architectures in state-of-the-art ML denoising :

- KPCN [3]: Kernel prediction
  - ✅ Good high-frequencies reconstruction
  - ❌ Sensitive to fireflies in references
- U-net [4]: Direct prediction
  - ✅ Suited for high-variance inputs
  - ❌ Loss of high-frequencies
- DEMC [5]: Direct prediction dual-encoder
  - ✅ Separately encoded noisy and auxiliary features to reduce redundancy
  - ❌ Loss of informations not represented in auxiliary features (such as specular reflections)

Loss function and metrics:

- Pixel-per-pixel (MSE, RMSE, NRMSE, etc.)
  - ❌ Sensitive to fireflies in training
- Structural (SSIM, MS-SSIM, etc.)
  - ❌ Prone to color shifts in training
- High-dynamic range pixel-per-pixel (SMAPE)
  - ✅ More suited for spectral values
  - ✅ Good color conservation in training
- Perceptual (LPIPS)
  - ❌ Limited to RGB space
  - ✅ Closer to human perception

Proposed solution

Our **Deep Learning** based **denoising method** operates the reconstruction **in the spectral domain** with the assumption that corrected radiometric values result in corrected photometric values.

A discretization of the light spectrum into n parts, called *n*-bins, is obtained from the renderer. These *n*-bins (where *n*=16) are then individually denoised before reconstructing the final RGB output.

Training used the SMAPE loss function and 4554 pairs of noisy/ground truth with 1024 (noisy) and 4M (Ground truth) samples-per-pixel.

Our evaluation compares our results with established off-the-shelf denoisers, both visually and using metrics.

References

[1] NVidia. 2017. *OptiX denoiser.* https://developer.nvidia.com/rtx/ray-tracing/optix

[2] Intel. 2021. *Intel Open Image Denoise.* https://www.openimagedenoise.org/

[3] Steve Bako et al. Kernel-predicting convolutional networks for denoising Monte Carlo renderings. ACM Transactions on Graphics 36, 4 (July 2017).

[4] Chakravarty R. et al. Interactive reconstruction of Monte Carlo image sequences using a recurrent denoising autoencoder. ACM Transactions on Graphics 36, 4 (July 2017).

[5] Xin Yang et al. DEMC: A Deep Dual-Encoder Network for Denoising Monte Carlo Rendering. Journal of Computer Science and Technology 34, 5 (Sept. 2019).

[6] Jonghee Back et al. 2020. Deep combiner for independent and correlated pixel estimates. ACM Transactions on Graphics 39, 6 (Dec. 2020), 1–12

Our method

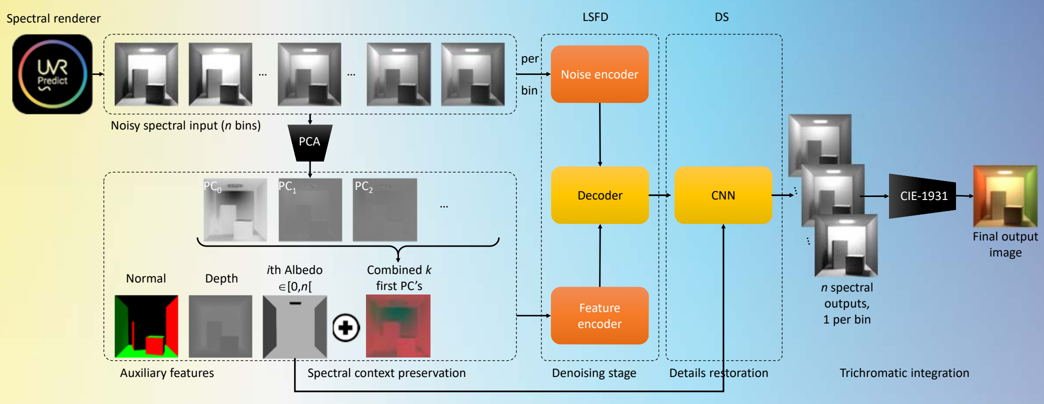

Figure 2: Overview of our spectral denoiser

Pipeline in Fig. 2 structured into 6 parts:

- Output of the spectral renderer (noisy discretized spectral bins, normals, depth and spectral albedo).
- Principal Component Analysis (PCA) on the spectral bins to compute a low-dimensional representation for context preservation. We combined the k-first PC (i.e. ≈95% of the variance with k=3).
- Auxiliary features concatenation (normal, depth, albedo, *k*-PC).
- Dual encoding denoising network inspired by the DEMC [5], our **Light Spectrum Features Denoiser** (LSFD) reconstructs each bin separately with the help of auxiliary features.
- Second estimator **Details Sharpener** (DS), derived from the DC [6], which reinjects high-frequency details from the albedo.
- Riemann integration to reconstruct a displayable RGB output.

Results

Dataset: 4554 set of 1024x768 images (noisy, GT, auxiliary features). Generation time: 3 months, size: 1.8 TB

Training: ≈3 days on NVIDIA A100 for 5,000 epochs. For each epoch, inputs are randomly cropped into 64x64 images. Training took approximately 40 seconds per epoch, divided into 30 seconds to create the crop dataset and 10 seconds for inference, giving a total over 5000 epochs of 1.7 days for crop dataset creation and 1.3 days for inference.

Evaluations: Fig. 1 and 3 show the improvement of our pipeline compared to OptiX. We are better able to reconstruct low-frequency areas, remove the color noise and predict the correct hue of the ground truth (GT). Metrics also support these results. Our final method (λ+PCA+DS) yields lower NRMSE values for all scenes, with each step of the pipeline decreasing the value in most cases. We get an average gain of about 0.0842 overall. Let's note that RGB solutions are less performant (red number).

Furthermore, Tab. 1 displays the same observation for the LPIPS value, except for the Bedroom scene, which can be explained by the absence of color shift between noisy and GT and the predominance of high-frequency details, better handled by OptiX.

The inference time of all these methods is ≈0.02 ms.

Method	Bathroom	Bedroom	Kitchen
OptiX	0.0231	0.0053	0.0679
λ	0.0084	0.0209	0.0328
λ+PCA	0.0074	0.0175	0.0312
λ+PCA+DS	0.0164	0.0091	0.0265

Table 1: LPIPS metric to measure perceptual gain.

Figure 3: Crops taken from three scenes denoised with various methods. From top to bottom : Bathroom, Bedroom and Kitchen. From left to right: noisy, OptiX, RGB domain (RGB), Spectral domain : LSFD without PCA (λ), with PCA (λ+PCA), and with DS (λ+PCA+DS). NRMSE was used as the evaluation metric.

Limitations

While our solution is able to reconstruct highly-noised inputs, complex scenes such as displayed here still require too many samples to denoise at an interactive frame-rate. Moreover, our solution suffers from a bigger loss in high-frequency details compared to OptiX. In order to explore its limitations, we plan to integrate more complex scenes (iridescence, volume rendering, etc.) into our tests.

Acknowledgment

This work is part of the research program ``LUCE'' (*Light-transport SimUlation and MaChine Learning*), which is funded by the French national agency for scientific research (ANR). We thank the ROMEO computing center and the Image Center at URCA for their material contributions in high-performance computing and visualization resources.

anr
project number
ANR-21-CE33-0010