Denoising Production Volumetric Rendering

Shilin Zhu
shilin@pixar.com
Pixar Animation Studios

Xianyao Zhang
xianyao.zhang@disneyresearch.com
Disney Research Studios

Gerhard Röthlin
gerhard@disneyresearch.com
Disney Research Studios

Marios Papas
marios.papas@disneyresearch.com
Disney Research Studios

Mark Meyer
mmeyer@pixar.com
Pixar Animation Studios

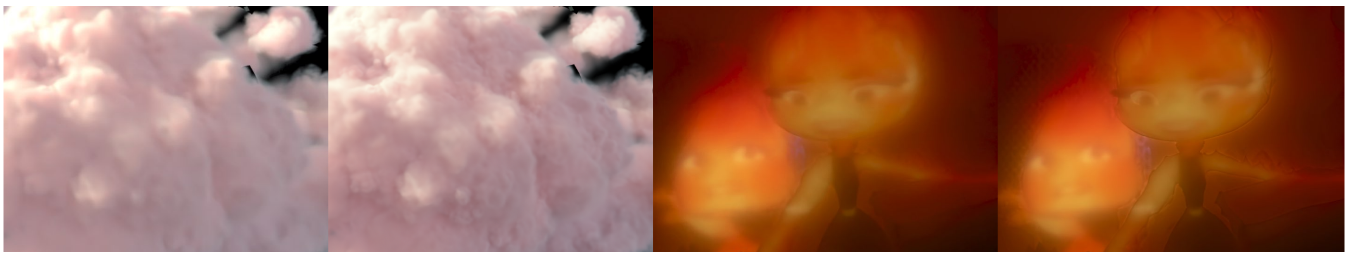

Figure 1: Comparing the current studio denoiser with the improved version proposed by this paper. It is clear that our new volume denoiser output is visibly crisper around edges and results in higher level of details (e.g., enhanced sharpness on the pink cloud boundaries, the fire characters, and the checkered object in the background). We produce these results from volumetric render passes (thick FX volumes and thin fogs) in Disney and Pixar's films Turning Red and Elemental. ©Pixar

ABSTRACT

Denoising is an integral part of production rendering pipelines that use Monte-Carlo (MC) path tracing. Machine learning based denoisers have been proven to effectively remove the residual noise and produce a clean image. However, denoising volumetric rendering remains a problem due to the lack of useful features and large-scale volume datasets. As a result, we have seen issues such as over-blurring and temporal flickering in the denoised sequence. In this work, we modify the production renderer to generate many types of potential volume-specific features that might improve the denoising quality, and then run a state-of-the-art feature selection algorithm to detect the best combination of those features. To train the denoiser for production use, we collect thousands of unique volumetric scenes from our recent films, and augment the inputs to create a large dataset for training. Our evaluation shows a good amount of quality gain compared to the version currently in use.

ACM Reference Format:

Shilin Zhu, Xianyao Zhang, Gerhard Röthlin, Marios Papas, and Mark Meyer. 2023. Denoising Production Volumetric Rendering. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Talks (SIGGRAPH '23 Talks), August 06-10, 2023.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3587421.3595432

1 INTRODUCTION

In the context of MC denoising, deep learning is widely considered as the state-of-the-art approach thanks to its ability to produce sharp and clean image. We choose Kernel-Predicting Convolutional Network (KPCN) proposed by [Bako et al. 2017; Vogels et al. 2018] to be used in the studio since it is more robust than regressing the pixel color directly. In addition, we further adjusted the neural architecture by leveraging U-Nets with scale-level kernel predictors, achieving considerably shallower design and faster inference.

Auxiliary feature buffers (such as albedo, depth, normals, etc.) are important for the neural network denoiser because they are much less noisy and provide necessary information of colors and shapes for objects contained in the rendered image. As a result, these features often improve the denoising quality by retaining more spatial details. For volumetric scenes, we are facing two main challenges: First, we need to find a set of available features for volumes that do not have hard surfaces. Second, keeping a large number of features adds unnecessary cost to the render time and disk storage especially for film production, thus we need to select a few most useful features to minimize such overhead. To reach a solution, we borrow the progressive feature selection method by [Zhang et al. 2022] and find it quite effective to generate a subset of candidate features that can perform well on volume denoising.

2 METHOD

Denoising Architecture. We use an improved KPCN architecture with U-Net as the backbone. The predicted denoising kernels are applied back to the input noisy image at multiple scales and results are combined together to yield the denoised output. We choose 5 scales for single-frame denoising and 4 scales for cross-frame (i.e., temporal) denoising considering there is extra cost of warping neighboring frames to the center frame using motion vectors.

Feature Selection. We generate a large collection of 16 potentially useful features, covering a wide range of lighting and physical properties of volumetric objects in the scene. These additional input layers are rendered simultaneously along with the primary beauty image. It is clear that many input features are mutually correlated, and it is difficult to identify the best combination that we

SIGGRAPH '23 Talks, August 06-10, 2023, Los Angeles, CA, USA
2023. ACM ISBN 979-8-4007-0143-6/23/08.
https://doi.org/10.1145/3587421.3595432

Table 1: List of selected volumetric denoising inputs (color is always selected and excluded from the selection algorithm).

Layer	Description
Color + Variance	Noisy image and sample variance
Single/Multi-Scatter	Lighting decomposition
Alpha	Volume opacity
SampleCount	Samples per pixel
Albedo	Basic volume color
Distance	Distance to scatter location
Density	Volume extinction coefficient
DensityGrad	Spatial extinction gradient
ScatterRatio	Volume scatter probability

should use for the denoiser since the search space grows exponentially with the size of candidate set. We implement the framework presented by [Zhang et al. 2022] on our unique set of candidate features, which only requires training of a single probe denoiser and near-optimal solutions can be found progressively. This method is efficient because it avoids training an ocean of denoisers with different configurations. The full selected input set is presented in Table 1. Furthermore, we can adjust the number of selected features conveniently based on production storage requirement by tuning the size parameter of the feature selection algorithm. For instance, if a show can only afford having 4 auxiliary features due to space limitations, we are able to further narrow down the list.

Loss Function. Following the tradition of KPCN, we supervise the neural network training with a reconstruction loss. More specifically, we use Symmetric Mean Absolute Percentage Error (SMAPE) to compute the difference between the denoised image and the reference image. In addition, a temporal loss on the pixel derivatives is added to penalize the inconsistency between successive frames. These two losses are mixed together through weighted addition.

Training Details. For denoiser training, we implement the KPCN framework using TensorFlow. Input channels that contain High Dynamic Range (HDR) values are pre-processed by the log-modulus transformation for range reduction. The denoiser is trained using gradient descent for 15.2M steps with a learning rate decay from 10^{-4} on a Tesla V100 GPU installed on a NVIDIA DGX cluster.

3 PRODUCTION DATASET

Training Data Collection. Since there is no large-scale volume dataset readily available, we build our own production dataset from the recent Disney and Pixar's films *Turning Red*, *Lightyear*, and *Elemental*. We manually select more than 2,000 unique volumetric renderings from hundreds of shots. These selections contain all kinds of volumes but primarily 3 types: simulated FX volumes, clouds, and fogs. Our production pipeline renders volumes in a separate pass where surfaces are matted out, and we combine elements by final compositing. For each selected camera view, we render both color and all the features listed in Table 1. In addition, we set 3 different target pixel variances on the adaptive sampler to generate inputs with multiple noise levels for training data augmentation. The reference image is rendered in between 8K (minimum) and 16K (maximum) samples per pixel chosen by the adaptive sampler. To support cross-frame temporal denoising, we always render a sequence of 5 consecutive frames for each selected center frame and pre-compute the motion vectors in between using optical flow.

RenderMan Implementation. We modified the volume BxDF shader in RenderMan 24 and 25 by adding the candidate features as Arbitrary Output Variables (AOVs). When path traced ray hits a volume, the BxDF writes the queried values to individual layers in the output EXR image file. We use Light Path Expressions (LPEs) to control the ray depth when recording those features to an output EXR layer. Non-illumination features are obtained from the primary ray hits.

4 RESULTS

With the selected features included as inputs, we successfully increase the quality of the denoised images by recovering more details and removing blotchiness more effectively, as presented in Figure 1. Additionally, re-training the denoiser using our dedicated volumetric dataset further boosts the performance since it does not need to account for hard surface denoising. We implement both 3-frame and 5-frame temporal denoisers with KPCN, which helps reducing flickering between successive frames in a shot sequence by warping the neighboring frames to the center one. The proposed denoiser is now used by the studio to reduce the render time of volumetric elements, including fogs, FX volumes, and clouds.

5 DISCUSSION AND CONCLUSION

We improve the denoising performance on volumetric renderings by analyzing the contribution of various input features. By using an advanced feature selection method and building a large-scale production dataset, we demonstrate a superior volume denoiser with reasonable cost. However, sometimes the extra sharpness from the new denoiser becomes undesirable in some special cases like motion blur or depth of field effects, which needs to be addressed further. Future research should also focus on improving the temporal consistency and combining screen-space with path-space denoising. In addition, we begin to see cases from production where hard surfaces and volumes are mixed together within a single render pass, thus our immediate next step is to find an unified solution combining both worlds instead of having two separate denoisers, while consistently expanding our dataset with volumetric renderings from future shows to cover more diverse styles and looks.

ACKNOWLEDGEMENT

We would like to thank Jeremy Newlin for integrating the proposed denoiser into studio's production pipeline, Stephen Friedman and Per Christensen for helping with RenderMan shader upgrade, and Julian Fong for volume rendering discussion.

REFERENCES

Steve Bako, Thijs Vogels, Brian McWilliams, Mark Meyer, Jan Novák, Alex Harvill, Pradeep Sen, Tony DeRose, and Fabrice Rousselle. 2017. Kernel-predicting convolutional networks for denoising Monte Carlo renderings. *ACM Trans. Graph.* 36, 4 (2017), 97–1.

Thijs Vogels, Fabrice Rousselle, Brian McWilliams, Gerhard Röthlin, Alex Harvill, David Adler, Mark Meyer, and Jan Novák. 2018. Denoising with kernel prediction and asymmetric loss functions. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–15.

Xianyao Zhang, Melvin Ott, Marco Manzi, Markus Gross, and Marios Papas. 2022. Automatic Feature Selection for Denoising Volumetric Renderings. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 63–77.