SSNeRF: Sparse View Semi-supervised Neural Radiance Fields with Augmentation

Xiao Cao[1] Beibei Lin[1] Bo Wang[2] Robby T. Tan[1] Zhiyong Huang[1]

National University of Singapore University of Mississippi

Code: https://github.com/massyzs/SSNeRF.git
Email: xiaocao@u.nus.edu
Wechat:c1265199717

Problem Definition and Contributions

Goal: Augment Sparse-view NeRF performance without using supplementary information

Motivation:

- Sparse-view NeRF is challenging due to the under-constrained optimization.
- Some existing methods address this issue by relying on supplementary information (e.g., depth).
- Additional-information-free methods rely on strong assumption and limit real-world application.

Contribution:

- A semi-supervised NeRF framework (i.e., SSNeRF).
- Three sparse-view-specific augmentations are introduced, focusing on (1) sample-level densities, (2) layer robustness, and (3) sparse-view degradation.
- Efficiently address the floating points and resolve flicking pixels in rendered videos.

Unbiased High-confidence Label Generation

Monte-Carlo Epistemic Confidence Map:

- Epistemic confidence map evaluates uncertainty caused by under-optimized model and can be achieved by comparing the same pixels generated by model perturbed with different ratio (e.g., dropout).

- To eliminate instableness caused by randomness, we use ensembling strategy that a series of small, varied dropout ratios (e.g., 0%, 5%, 15%, 20%) and calculate the variance of each pixels.

$$Conf = 1 - \frac{\sum var(image'_{(0\%,\ldots,20\%)})}{4}$$

Biase correction:

Black and white regions are more stable. To address this issue, HSV filter is applied to filter out those extreme regions which may leads to over/under-saturated results.

Sparse-view-specific Augmentation

Sample-level Augmentation:

- Floaters are caused by pointxs with small density value and non-zero RGB.
- By perturbing density values with τ -noise, NeRF adjusts sample-level RGBs according to wrong density.

$$\varepsilon \sim \mathbf{P}(x) = \frac{e^{e^{-x^2}}}{e^{e^{x^2}} + e^{e^{-x^2}}}$$

$$\mathbf{c}_i = f^n(r, \omega \mathbb{1}(i = n)\varepsilon)$$

$$\hat{c}(\mathbf{r}) = \sum_{i}^{N}(T_i(1 - e^{-(\sigma_i)\delta_i}) + \omega\varepsilon)\mathbf{c}_i$$

$$\hat{C}(\mathbf{r}) = \odot\hat{c}(\mathbf{r_m}, \mathbb{1}(\mathbf{r} \in \Re))$$

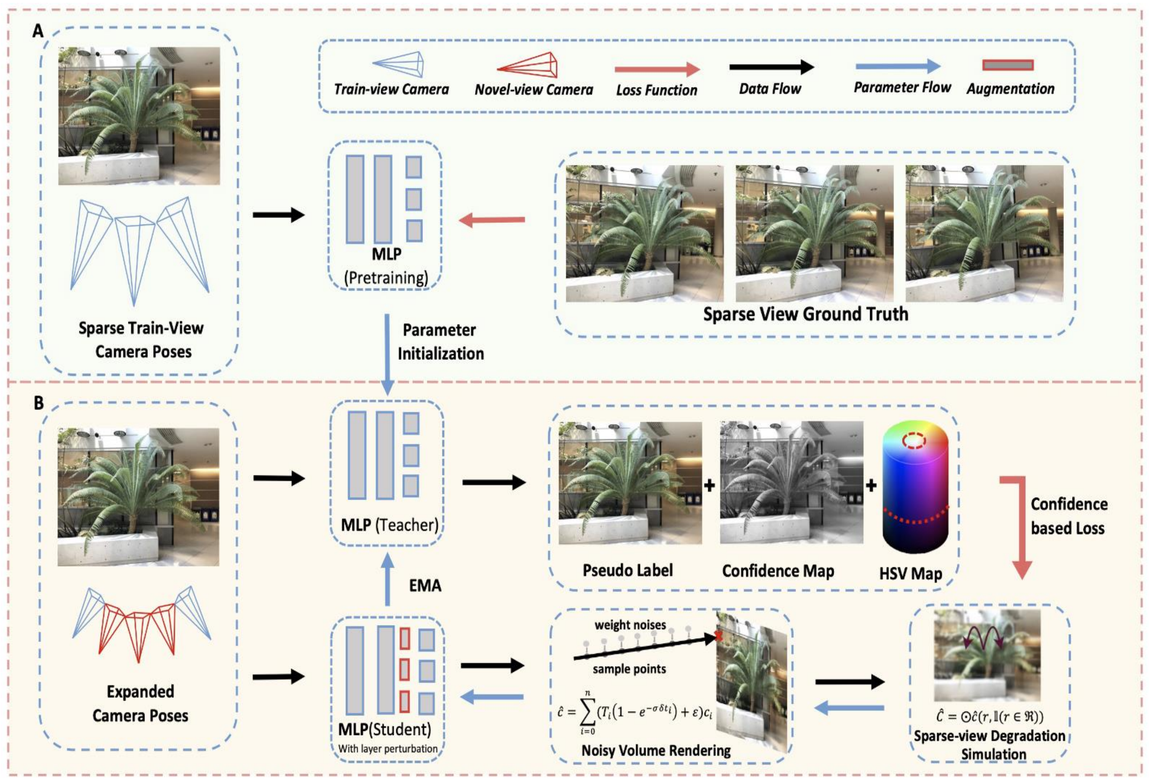
Layer Augementation:

Layers showing large parameter variance across NeRFs trained with different view counts were treated as vulnerable layers and we augment those layers.

Sparse-view Degradation Simulation:

To augment NeRF from result aspect, we simulate the blurriness by assigning the brightest color to nearby pixels, collaborating with a patch sampling strategy.

Pipeline

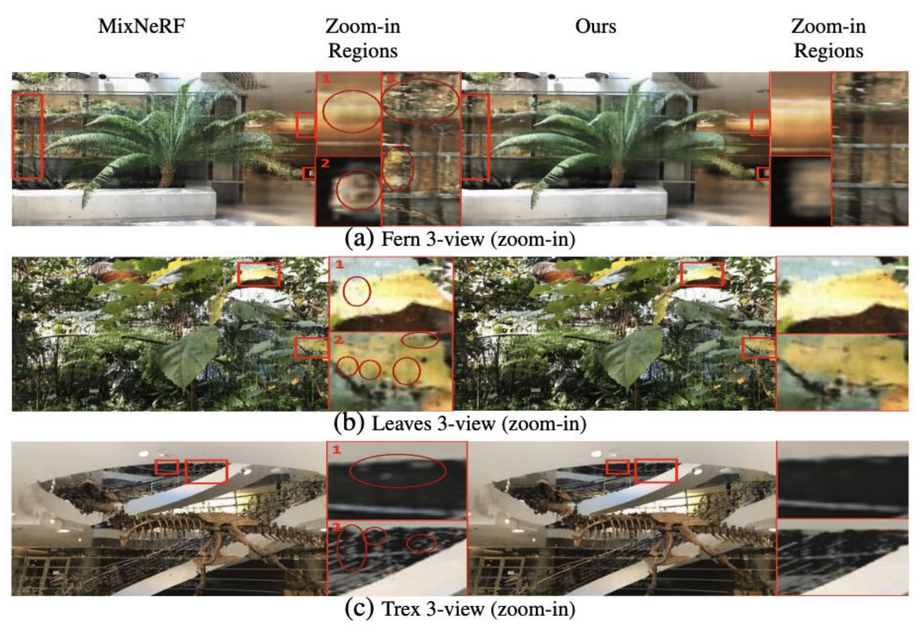


Experimental Results

Quantitative Results

Approach	PSNR ↑ 3-view	SSIM ↑ 3-view	LPIPS ↓ 3-view	PSNR ↑ 4-view	SSIM ↑ 4-view	LPIPS ↓ 4-view	PSNR ↑ 6-view	SSIM ↑ 6-view	LPIPS ↓ 6-view
LLFF dataset									
MipNeRF	17.377	0.714	0.335	19.142	0.7852	0.214	22.122	0.876	0.150
RegNeRF	16.005	0.359	0.363	18.030	0.518	0.255	21.155	0.647	0.190
MixNeRF	18.011	0.731	0.243	18.928	0.789	0.187	22.902	0.891	0.132
Ours	18.113	0.748	0.250	19.400	0.796	0.190	22.990	0.892	0.110
Blender dataset									
MipNeRF	10.316	0.445	0.385	9.812	0.397	0.494	18.844	0.875	0.166
RegNeRF	18.791	0.763	0.235	20.759	0.781	0.277	22.531	0.817	0.227
MixNeRF	18.420	0.781	0.250	21.821	0.880	0.209	25.289	0.960	0.151
Ours	18.955	0.893	0.213	21.357	0.925	0.153	25.391	0.974	0.096

Qualitative Results



(a) Fern 3-view (zoom-in)

(b) Leaves 3-view (zoom-in)

(c) Trex 3-view (zoom-in)

Ablation Study

Augmentation	PSNR ↑	SSIM ↑	LPIPS ↓
Vanilla MixNeRF	19.076	0.738	0.345
Layer augmentation	19.194	0.744	0.365
Sparse-view Degradation	19.282	0.0.745	0.365
Weight Perturbation	19.233	0.746	0.364
Density Perturbation	19.208	0.745	0.365
Ours(student)	17.289	0.676	0.394
Ours	19.275	0.746	0.364

Student NeRF overfits on specific noise pattern. EMA transfers only denoising ability.

Denoising Ability

Augmentation	PSNR ↑	SSIM ↑	LPIPS ↓
MipNeRF	18.571	0.703	0.493
RegNeRF	17.417	0.395	0.775
MixNeRF	18.727	0.712	0.412
Ours(student)	17.569	0.671	0.409
Ours	18.793	0.713	0.408

We inject uniform noise to densities, and ours (teacher NeRF) shows the best denoising ability and achieve the best performance.