NEX$^+$: NOVEL VIEW SYNTHESIS WITH NEURAL REGULARISATION OVER MULTI-PLANE IMAGES

Wenpeng Xing and Jie Chen

Department of Computer Science
Hong Kong Baptist University

ABSTRACT

We propose Nex$^+$, a neural Multi-Plane Image (MPI) representation with alpha denoising for the task of novel view synthesis (NVS). Overfitting to training data is a common challenge for all learning-based models. We propose a novel solution for resolving such issue in the context of NVS with signal denoising-motivated operations over the alpha coefficients of the MPI, without any additional requirements for supervision. Nex$^+$ contains a novel 5D Alpha Neural Regulariser (ANR), which favors low-frequency components in the angular domain, i.e., the alpha coefficients' signal subspace indicating various viewing directions. ANR's angular low-frequency property derives from its small number of angular encoding levels and output basis. The regularised alpha in Nex$^+$ can model the scene geometry more accurately than Nex, and outperforms other state-of-the-art methods on public datasets for the task of NVS.

***Index Terms*—** Novel view synthesis, multi-plane image, image denoising, neural basis learning.

1. INTRODUCTION

Novel view synthesis (NVS) has attracted great research attention recently for its potentials on virtual reality and other immersive applications. The performance of a NVS pipeline should be evaluated in two aspects, i.e., accurate estimation of scene geometry and the rendering of photo-realistic view-dependent effects.

A number of scene representations that jointly model geometry and appearances were proposed recently, such as Layered Depth Image [1], Multi-Plane Image (MPI) [2, 3], Neural Radiance Field (NeRF) [4], Deep Voxels [5] and Neural Volumes (NV) [6]. Volumetric rendering technique [7] are widely adopted, which accumulates color density values weighted with opacity sampled on 3D points along a ray. The ray-casting mechanism adopted in [3, 4], which queries a continuous function in the form of an MLP, faces a common issue – the geometrical ambiguity of the neural radiance field. NeRF^{++} [8] represented a recent attempt to address such ambiguity existed in NeRF. Such a bottleneck also exists for the MPI. Though the geometrical ambiguities in the MPI during alpha compositing do not affect much the direct rendering quality, however, when it is employed by a recent framework, Nex, which tries to extend the conventional MPI with neural angular basis, these geometrical artefacts start to cause trouble. In this paper, we refer to such ambiguity as Alpha Ambiguity in the context of MPI.

We propose a novel 5D Alpha Neural Regulariser (ANR) to denoise the alpha value along rays. The rendered depth from Nex$^+$ is more accurate and complete than that from Nex, and achieves state-of-the-art performance on rendering quality. We show that when the number of training images decreases, the rendered RGB images and estimated depth from Nex$^+$ contain much less observable artifacts than those from Nex.

2. RELATED WORK

MPI was first proposed by Zhou et al. [2] for the task of NVS. Conventional methods that generate MPI [2, 9, 10] require Plane-Sweep-Volume (PSV) as input to detect pixel's correspondences between two images in a set of discrete depth planes. With advancements made in recent studies, MPI can be generated from either a single image [11] or a set of images [9, 3]. To push the rendering boundary of MPI, Srinivasan et al. [10] theoretically analysed the positive correlations between the rendering baseline and the number of planes in MPI. To further enlarge the rendering baseline of MPI, several neighbouring MPIs can be combined to render large-baseline novel views [9]. However, MPI based methods still face challenges of modeling extreme specular components and view-dependent effects. So Nex [3] were proposed to expand the conventional MPI with neural expansions that can reproduce photo-realistic specular and reflective surfaces.

The ANR herein proposed is directly related to the task of high-dimensional signal modelling and denoising. Specifically, the task of image denoising can be achieved by modeling image priors, such as non-local self-similarity (NSS) [12], sparse coding [13], gradient models [14] and Markov random field (MRF) models [15]. With the rise of deep learning, Convolutional Neural Networks (CNNs) were used in [16] to denoise images. But in addition to CNNs, multi-layer perceptron (MLP) networks were also applied in [17, 18] to tackle

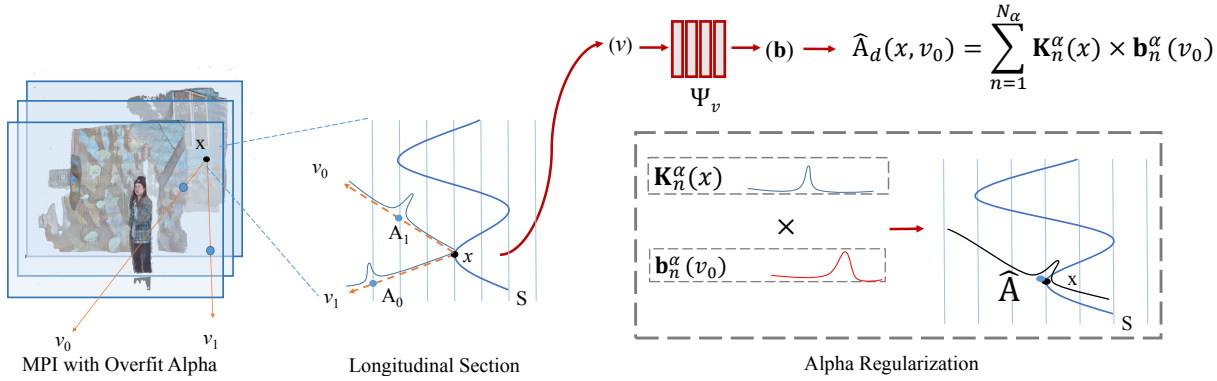


Fig. 1. The overall pipeline. Nex$^+$ learns a MPI representation with alpha neural regularization (ANR) that can render photo-realistic novel views even under extreme sparse multi-view images. The proposed ANR, a low-frequency basis function in angular domain, can suppress noisy alpha values in MPI along multiple rays.

the task of image denoising, and achieved comparable results with BM3D.

3. APPROACH

Given a set of multi-view images $\{\mathbf{I}_i\}_{i=1}^{i=N_i}$ of a scene where N_i is the image number, our goal is to learn a 3D scene representation that can render photo-realistic novel views. Inspired by Nex, we propose a novel neural regulariser, i.e., ANR, to denoise the noisy alpha values across the angular domain. The denoised alpha, can more accurately reflect the scene geometry, and prevent the MPI from overfit to training images, and produce better visual quality, especially when the number of training images are small.

3.1. Background on MPI and its Angular Extension - Nex

The MPI representation $\mathbf{M} = \{\mathbf{C}_d, \mathbf{A}_d\}_{d=1}^{D} \in \mathcal{R}^{H\times W\times 4}$ consists of D sets of planar images reflecting different scene depth. H and W are height and width. For each depth plane d of the MPI, $\mathbf{C}_d \in \mathbb{R}^{H\times W\times 3}$ denotes RGB color values and $\mathbf{A}_d \in \mathbb{R}^{H\times W\times 1}$ denotes the alpha values. Rendering of the MPI can be done according to the following equation:

$$\hat{I} = \mathcal{O}(\mathcal{W}(\mathbf{A}), \mathcal{W}(\mathbf{C})), \tag{1}$$

where $\mathbf{A}$ and $\mathbf{C}$ represents the alpha and RGB color planar images sorted in the order from back to front. $\mathcal{W}$ denotes homography operation which warps the planar images from the reference to the target viewing angle, $\hat{I}$ is the rendering output view at the target angle. The *over*-compositing [19] operator $\mathcal{O}$ works from back-to-front ($d = 1, \cdots D$):

$$\mathcal{O}(\mathbf{A}, \mathbf{C}) = \sum_{d=1}^{D} \mathbf{C}_d \mathcal{T}_d(\mathbf{A}), \mathcal{T}_d(\mathbf{A}) = \mathbf{A}_d \prod_{i=d+1}^{D} (1 - \mathbf{A}_i). \tag{2}$$

The operator $\mathcal{T}$ is differentiable, and allows MPI to be learned from pixel rendering loss.

MPI can simulate view-dependent effects to some extent by dynamically warping the planar images to different viewing angles. However, variation is limited and restricted by geometry. Nex introduces neural expansion to the MPI's color values $\mathbf{C}$ to capture extra angular variation:

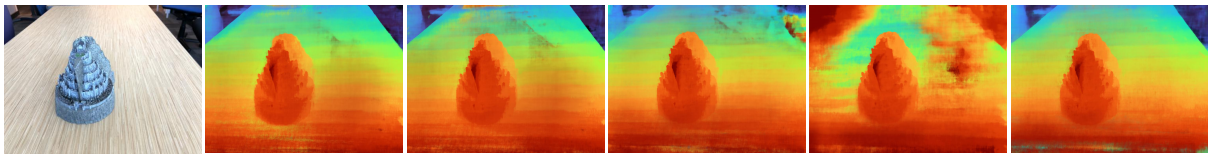
$$\hat{\mathbf{C}}_d(x, v) = \mathbf{K}_0^c(x) + \sum_{n=1}^{N_c} \mathbf{K}_n^c(x) \times \mathbf{b}_n^c(v), \tag{3}$$

where x denotes the 3D coordinate of a spatial point, and v denotes the viewing direction in the angular domain. $\{\mathbf{b}_n^c \in \mathcal{R}^{3\times N_v}\}_{n=1}^{N_c}$ are the angular basis which captures *the variation along the angular dimension*. N_v denotes the dimensionality of the expanded angular space, and N_c denotes the number of angular basis. $\{\mathbf{K}_n^c \in \mathbb{R}^3\}_{n=0}^{N_c}$ are the color coefficients for each basis. Both the basis $\{\mathbf{b}_n^c\}_{n=1}^{N_c}$ and their corresponding coefficients $\{\mathbf{K}_n^c\}_{n=1}^{N_c}$ are explicitly learned by multi-layer perceptrons (MLPs).

As defined in Eq. (3), the color component $\hat{\mathbf{C}}_d(x, v)$ at point x, depth plane d, and when viewed from direction v is expressed by *linear combination of neural expanded bases and their coefficients*. The final rendered image at the given viewing direction v can be composited by replacing $\mathbf{C}_d(x)$ with $\hat{\mathbf{C}}_d(x, v)$ in Eq. (3). The final output from Nex is consequently able to produce a photo-realistic image with much better view-dependent effects.

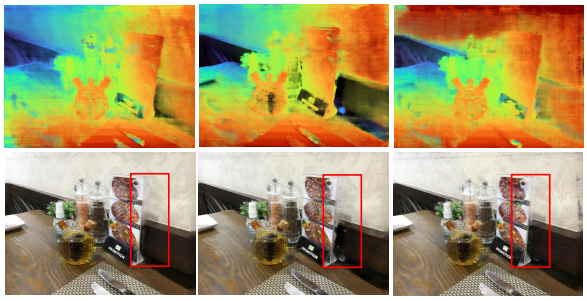
3.2. Alpha Neural Regulariser

Our alpha denoising solution is inspired by neural basis learning for image denoising. Alpha's neural basis $\{\mathbf{b}_n^\alpha \in \mathbb{R}^{1\times N_v}\}_{n=1}^{n=N_\alpha}$ is predicted by ANR $\{\Psi_v(r(v)) : \mathbb{R}^{N_v} \rightarrow \mathbb{R}^1\}$ where $r(\cdot)$ is positional encoding method in [4], then multiplied and summed along channel dimensions with alpha's



(a) (b) (c) (d) (e) (f)

Fig. 2. Visualization of ANR's denoising functionality. (a) shows the ground truth RGB image of fortress from NeRF's real forward-facing dataset; (b) shows the results of Nex; (c) shows the result of Nex without coefficients sharing; (d) and (e) show the results of Nex$^+$ setting the number of alpha basis N_α as two and three; The result of Nex$^+$ (with ANR and without coefficients sharing, N_α is one) is shown in (f) and does not have noisy alpha values.



(a) (b) (c)

Fig. 3. Visualization of effects of viewing direction's encoding level. The number of training and testing images are 4 and 41 respectively. (a) and (b) are results from Nex$^+$ setting N_r as 3 and 16 respectively, where strong rendering degradation and overfit can be observed when N_r increased. (c) shows that Nex produces strong cloudy artifacts.

(a) (b) (c)

Fig. 4. Qualitative results on Nex's Shiny dataset. The output of Nex$^+$ (a) captures more fine details than Nerf (b) and Nex (c) in Pasta scene.

coefficients $\mathbf{K}_n^\alpha \in \mathbb{R}^1$, which is given as:

$$\hat{\mathbf{A}}_d(x, v) = \sum_{n=1}^{N_\alpha} \mathbf{K}_n^\alpha(x) \times \mathbf{b}_n^\alpha(v), \tag{4}$$

where $\hat{\mathbf{A}}_d(x, v)$ denotes the regularised alpha in point x given viewing direction v, n is the index of basis and coefficient, N_α is the number of alpha's coefficients as well as the number of alpha's neural basis. So the rendering equation is replacing alpha $\mathbf{A}_d(x)$ in Eq. (2) with ANR's neural denoised alpha $\hat{\mathbf{A}}_d(x, v)$.

The rationale of the regularising operation is that the alpha of a scene point x should show small variance along different viewing directions, or the density of a 3D point is angular invariant. The neural basis $\{\mathbf{b}_n^\alpha\}_{n=1}^{N_\alpha}$ is a low-frequency basis that can reconstruct alpha with low-frequency constraints in the angular domain. And different from 2D image denoising, ANR is working on 5D space (i.e. 3D spatial coordinates and the additional 2D angular directions).

The angular low-frequency neural basis function, ANR, is parameterised by a Multi-Layer Perception (MLP) network. A learnable neural basis is flexible, and proved to have better performance when using the same number of color coefficients in Nex's experiments. More importantly, the learnable neural basis can be interpolated smoothly among various viewing directions v, which is compulsory to the continuous change of viewing angles. The low-frequency property of ANR arises from two factors: (1) the number of positional encoding levels N_r to input viewing directions v. Inspired by NeRF^{++}, the smaller number of encoding level is used, the more high-frequency components are neglected. But efficient number of encoding level has to be reserved to encode input into a representative space. So the number of encoding levels N_r for ANR's input ray v is set as three. Experimental results of using different encoding levels for ANR are shown in Fig. 3; (2) the number of the basis, N_α. Increasing the number of basis and the coefficients results in a more diverse linear combination. In Nex, N_c is set as eight to model varying specular components and angular high-frequency effects. But for ANR – a angular low-frequency function, a higher N_α impairs its low-frequency property. So N_α is set as one. We also made experiments of changing N_α to two and three, results are shown in Fig. 2 (d) to (e) .

3.3. Explicit and Implicit Learning Strategy

Inspired by Nex, color parameters $\mathbf{K}_0^c$ are directly learned, color coefficients $\mathbf{K}_n^c$ and alpha coefficients $\mathbf{K}_n^\alpha$ are predicted by coefficient networks Ψ_{coef} parameterised by MLP. $\mathbf{b}_n^\alpha$ and $\mathbf{b}_n^c$ are low- and high-frequency neural basis respectively for alpha and color on angular domains. Learnable parameters in Nex$^+$ are supervised by image reconstruction loss $\mathcal{L}$:

(a) (b) (c) (d) (e) (f)

Fig. 5. Qualitative results on NeRF's forward-facing dataset. Column (a) shows ground-truth test views of Room and Horn. Column (b) shows the output of Nex+ that captures more accurate geometry and details than results of Nerf (c), Nex (d), SRN (e) and LLFF (f).

$$\mathcal{L} = ||\hat{I} - I||^2 + \omega||\nabla\hat{I} - \nabla I||_1, \quad (5)$$

where $\hat{I}$ is a render image, I is a ground truth image from the same view, ω is a balancing weight and ∇ is gradient operator.

4. EXPERIMENTS

4.1. Implementation Details

The model is implemented in PyTorch, and optimized for each scene. The model takes 48 hours to be trained for 4000 epochs on one Nvidia Tesla V100 GPU. The number of color basis N_c is 8. The level of positional encoding for coordinates and viewing directions are 8 and 3 respectively.

4.2. Comparison to the State of the Art

Our model is trained and evaluated against state-of-the-art methods of NVS on Real Forward-Facing Dataset and Shiny Dataset, both contain eight real-world scenes. The resolution of output image is 1008 $\times$ 756 pixels. Our method is evaluated against NeRF [4], Nex [3], LLFF [9] and SRN [20] using three metrics: PSNR, SSIM and LPIPS [21]. We use the same method of splitting training and testing images as NeRF and Nex. Our method achieves the highest average score in three evaluating metrics as shown in Table 1, and visualized in Fig. 4 and Fig. 5.

4.3. Ablation Study

Experimental results of using different number of positional encoding levels for viewing directions and different number of alpha basis are shown in Fig. 3 (a), (b) and Fig. 2 (d), (e). The experiments of different combinations of ANR and coefficients sharing are shown in Fig. 2 (b), (c). The above figures prove that ANR can denoise alpha efficiently. Setting N_α as one and setting N_r as three generate the best denoising results.

Table 1. Average scores across 8 scenes in NeRF's Real Forward-Facing dataset and Nex's Shiny dataset.

Method	Dataset	PSNR ↑	SSIM ↑	LPIPS ↓
SRN [20]	Forward-Facing	21.82	0.744	0.464
LLFF [9]	Forward-Facing	24.41	0.863	0.211
NeRF [4]	Forward-Facing	26.76	0.883	0.246
NeX [3]	Forward-Facing	27.26	0.904	0.178
NeX$^+$ (Ours)	Forward-Facing	**28.04**	**0.915**	**0.151**
NeRF [4]	Shiny	25.60	0.851	0.259
NeX [3]	Shiny	26.45	0.890	0.165
NeX$^+$ (Ours)	Shiny	**27.27**	**0.902**	**0.148**

5. CONCLUSION

In this paper, we propose a novel approach of solving alpha's ambiguity in MPI from the perspective of image denoising. Our approach is able to denoise alpha efficiently, produce complex view-dependent effects, and help prevent MPI from being overfitted. Nex$^+$ achieves state-of-the-art performance on rendering quality. We believe Alpha Neural Regulariser based on basis learning for image denoising can be applied to general problem of solving neural radiance field's ambiguity.

6. REFERENCES

[1] Jonathan Shade, Steven Gortler, Li-wei He, and Richard Szeliski, "Layered depth images," in *ACM SIGGRAPH*, 1998, pp. 231–242.

[2] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely, "Stereo magnification: Learning view synthesis using multiplane images," *ACM Transactions on Graphics*, vol. 37, no. 4, pp. 1–12, 2018.

[3] Suttisak Wizadwongsa, Pakkapon Phongthawee, Jiraphon Yenphraphai, and Supasorn Suwajanakorn, "Nex: Real-time view synthesis with neural basis expansion," in *CVPR*, 2021, pp. 8534–8543.

[4] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020, pp. 405–421.

[5] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhöfer, "Deepvoxels: Learning persistent 3d feature embeddings," in *CVPR*, 2019, pp. 2437–2446.

[6] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh, "Neural volumes: Learning dynamic renderable volumes from images," *ACM Trans. Graph.*, vol. 38, no. 4, 2019.

[7] Thomas Porter and Tom Duff, "Compositing digital images," *SIGGRAPH Comput. Graph.*, vol. 18, no. 3, pp. 253–259, 1984.

[8] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun, "Nerf++: Analyzing and improving neural radiance fields," *arXiv:2010.07492*.

[9] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar, "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines," *ACM Transactions on Graphics*, vol. 38, no. 4, pp. 1–14, 2019.

[10] Pratul P. Srinivasan, Richard Tucker, Jonathan T. Barron, Ravi Ramamoorthi, Ren Ng, and Noah Snavely, "Pushing the boundaries of view extrapolation with multiplane images," in *CVPR*, 2019, pp. 175–184.

[11] Richard Tucker and Noah Snavely, "Single-view view synthesis with multiplane images," in *CVPR*, 2020, pp. 548–557.

[12] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng, "Weighted nuclear norm minimization with application to image denoising," in *CVPR*, 2014, pp. 2862–2869.

[13] Michael Elad and Michal Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[14] Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin, "An iterative regularization method for total variation-based image restoration," *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 460–489, 2005.

[15] Xiangyang Lan, Stefan Roth, Daniel Huttenlocher, and Michael J Black, "Efficient belief propagation with learned higher-order markov random fields," in *ECCV*, 2006, pp. 269–282.

[16] Viren Jain and Sebastian Seung, "Natural image denoising with convolutional networks," in *NIPS*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds., 2009, vol. 21, pp. 769–776.

[17] Harold C. Burger, Christian J. Schuler, and Stefan Harmeling, "Image denoising: Can plain neural networks compete with bm3d?," in *CVPR*, 2012, pp. 2392–2399.

[18] S. Zhang and E. Salari, "Image denoising using a neural network based non-linear filter in wavelet domain," in *ICASSP*, 2005, vol. 2, pp. 989–992.

[19] Thomas Porter and Tom Duff, "Compositing digital images," in *SIGGRAPH*, 1984, pp. 253–259.

[20] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein, "Scene representation networks: Continuous 3D-structure-aware neural scene representations," in *NeurIPS*, 2019, pp. 1119–1130.

[21] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018, pp. 586–595.