Exploring 3D-aware Latent Spaces for Efficiently Learning Numerous Scenes

Antoine Schnepf[*1,3], Karim Kassab[*1,2],
Jean-Yves Franceschi[1], Laurent Caraffa[2], Flavian Vasile[1], Jeremie Mary[1],
Andrew Comport[3], Valérie Gouet-Brunet[2]

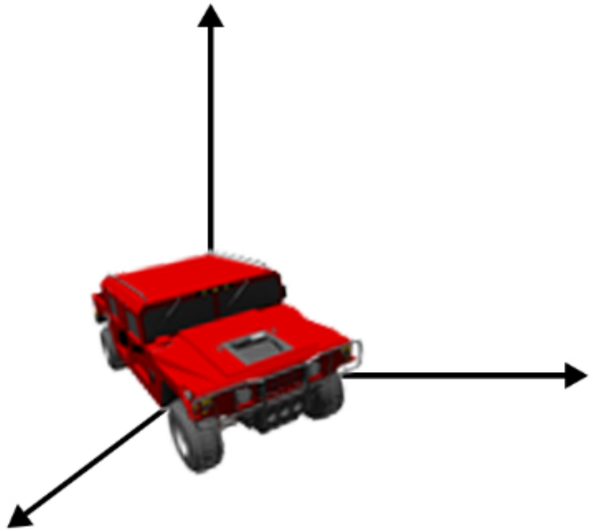
[*] Equal Contributions
[1] Criteo AI Lab, Paris, France
[2] LASTIG, Université Gustave Eiffel, IGN-ENSG, F-94160 Saint-Mandé
[3] Université Côte d'Azur, CNRS, I3S, France

work under review at 3DMV - CVPR

K. Kassab, March 2024

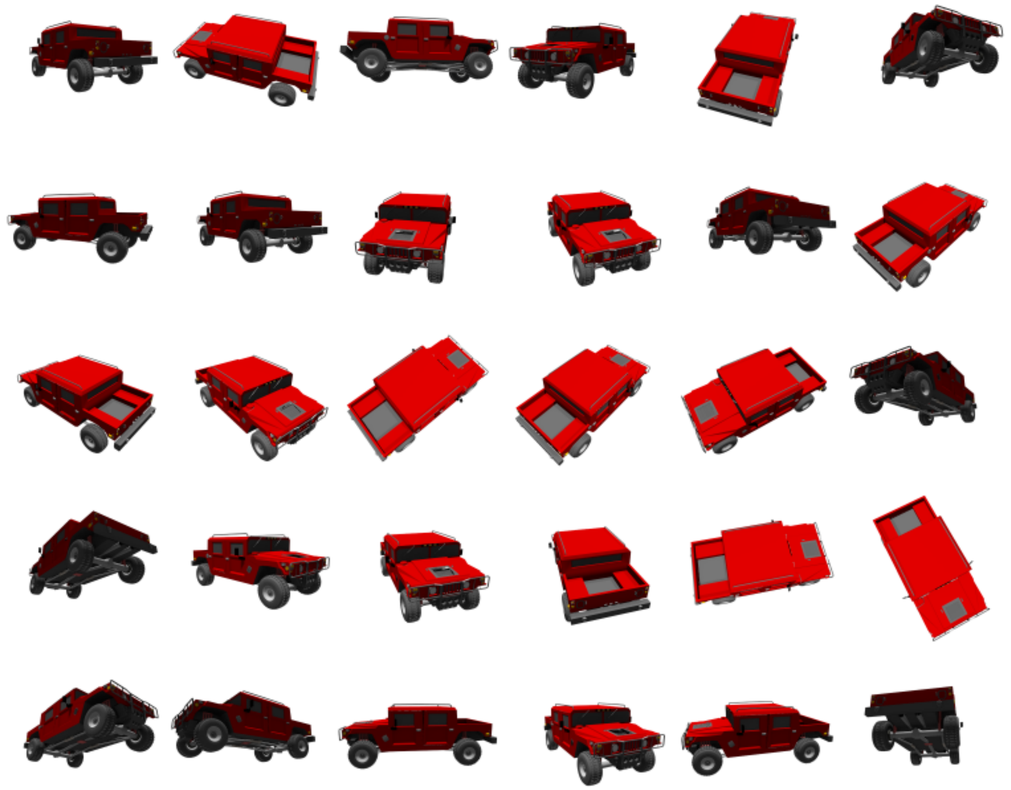
Pre-requisites

The inverse graphics problem

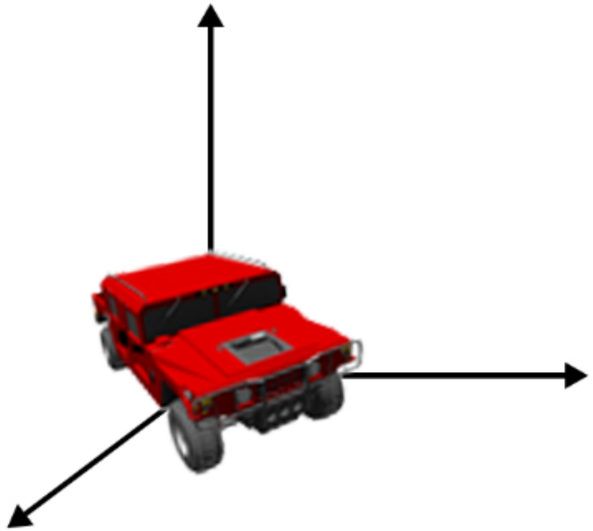


Mesh
Voxel grid
Point Cloud
SDF

Rendering

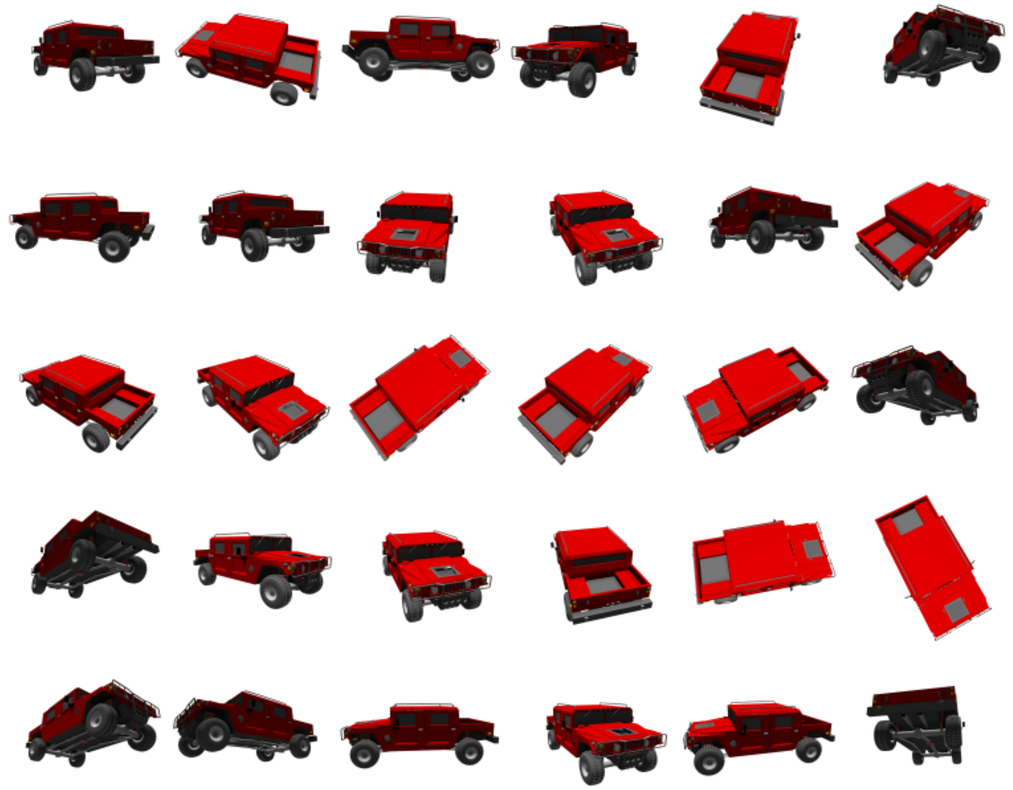


The inverse graphics problem



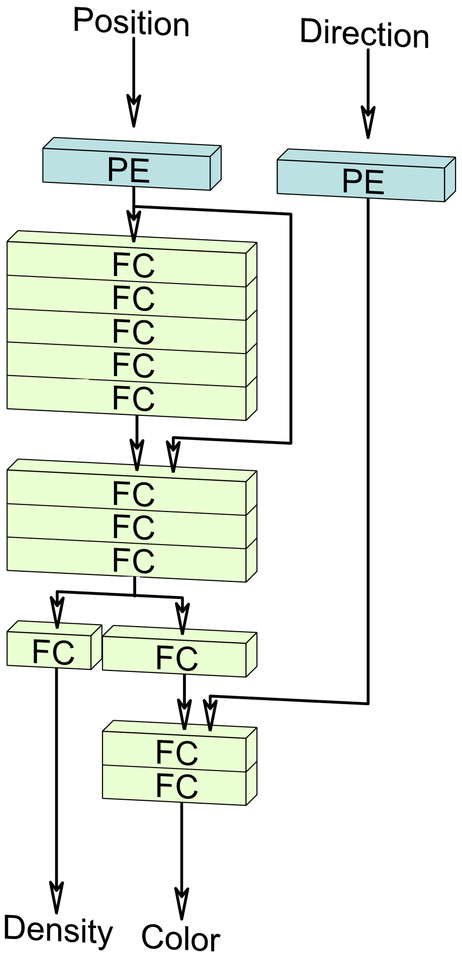
Rendering

Inverse Graphics

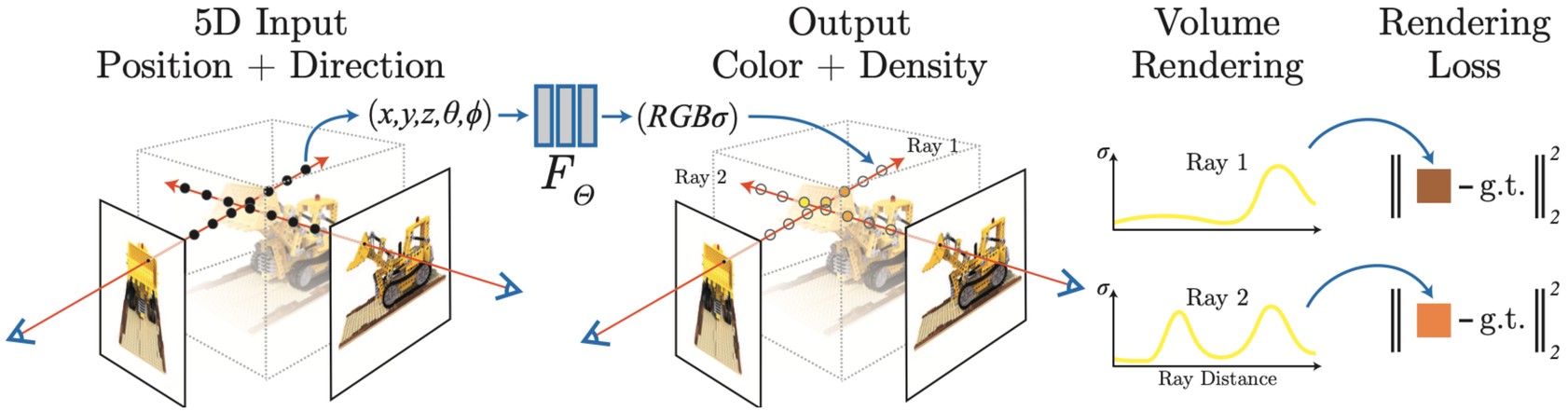


Mesh
Voxel grid
Point Cloud
SDF

Neural Radiance Fields



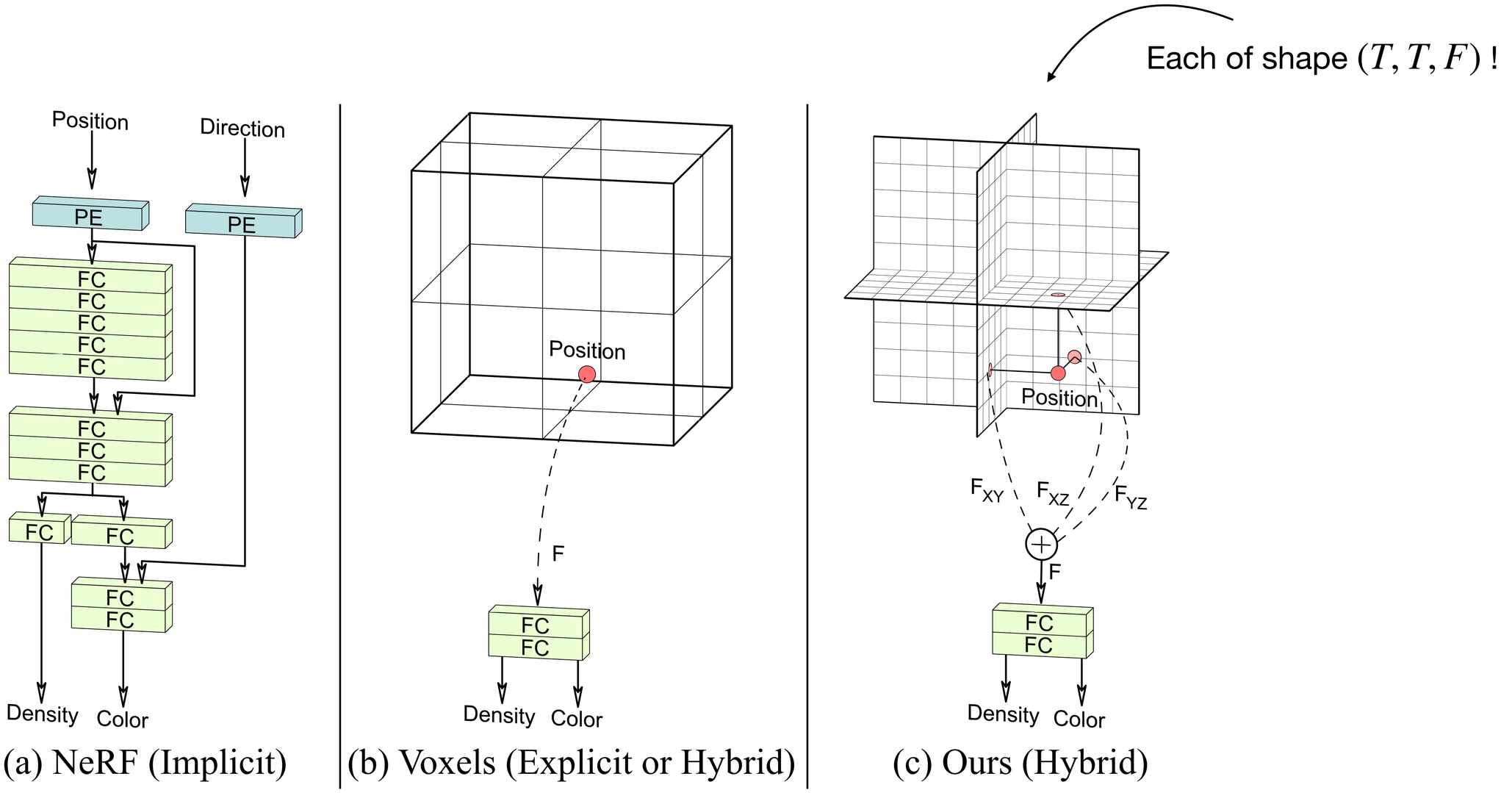
[Efficient Geometry-aware 3D GANs]



[NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis]

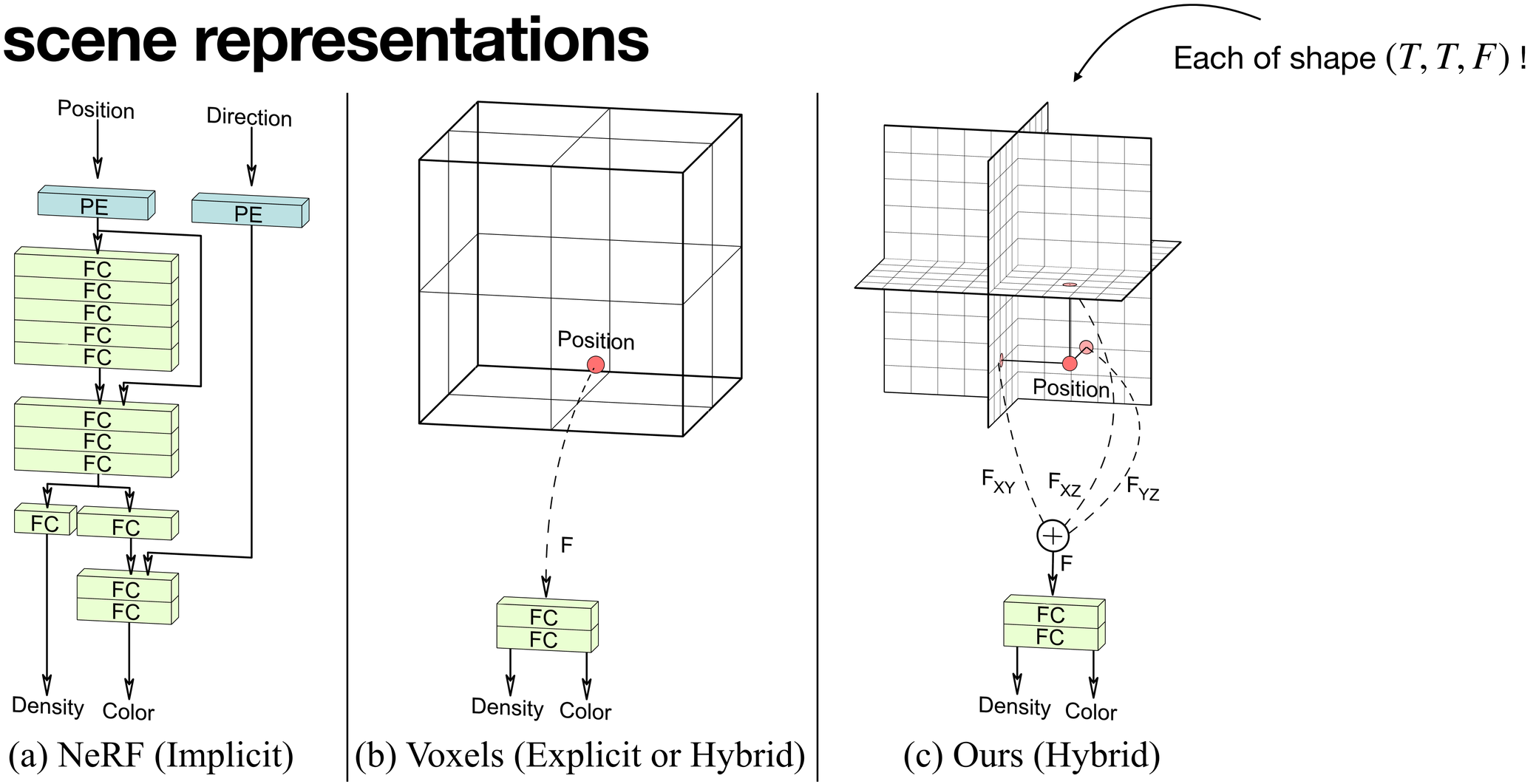
- **NeRFs are implicit representations trained to replicate views of a scene**
- **NeRF ≅ MLP + Volume Rendering Equations**

Tri-Planes scene representations



[Efficient Geometry-aware 3D GANs]

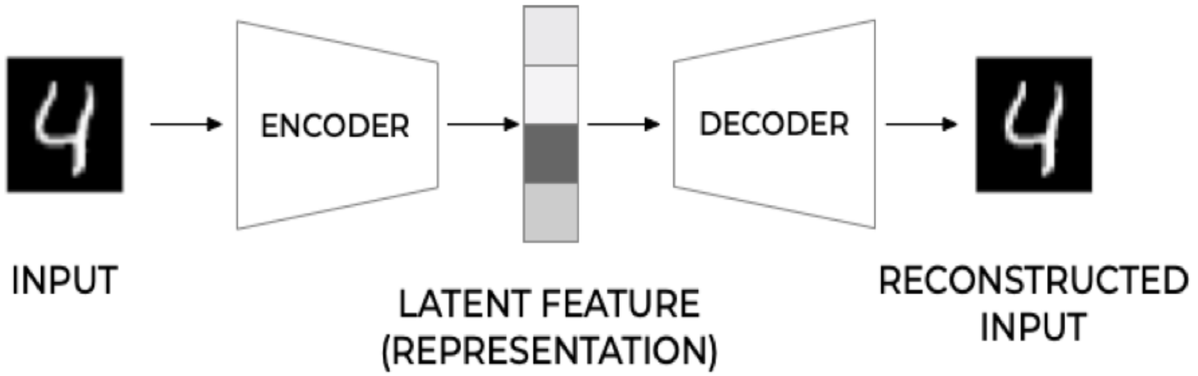
Tri-Planes scene representations



[Efficient Geometry-aware 3D GANs]

- **Tri-Planes are explicit-implicit representations**
- **Tri-Planes $\cong$ Three planes + tinyMLP + Volume Rendering Equations**
- **Both NeRFs and Tri-Planes are not scalable**

Auto-Encoders



[An Introduction to Autoencoders]

$$z = E_\psi(x) \ ,$$
$$\hat{x} = D_\phi(z) \ ,$$
$$\mathcal{L}_{\text{ae}}(\psi, \phi) = \mathbb{E}_x \|x - \hat{x}\|_2^2 \ ,$$

Exploring 3D-aware Latent Spaces for Efficiently Learning Numerous Scenes

Inverse Graphics Problem

How to model a scene using its captured images?

(Scaled) Inverse Graphics Problem

How to model abundantly many scenes at once?

3D-aware latent space

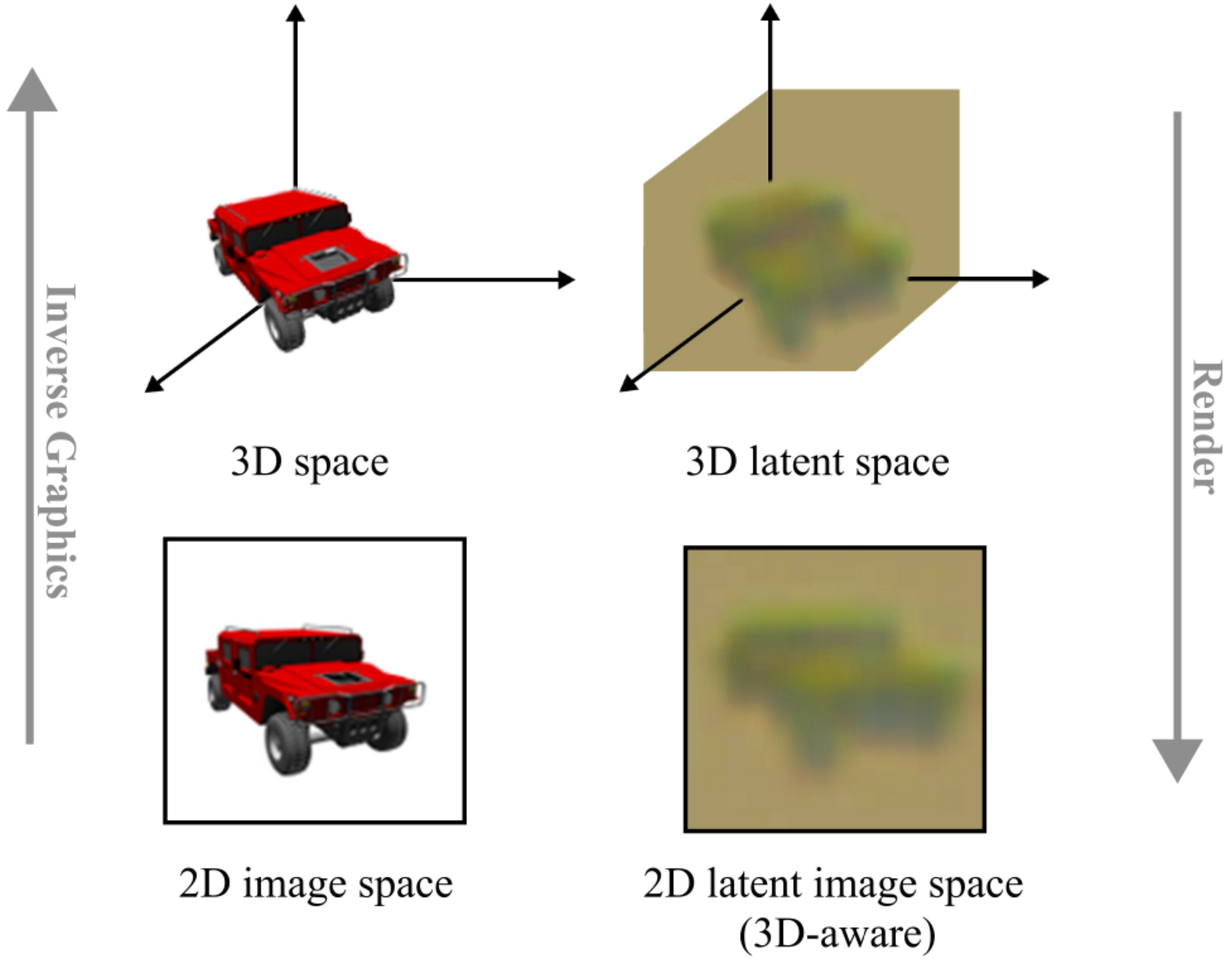
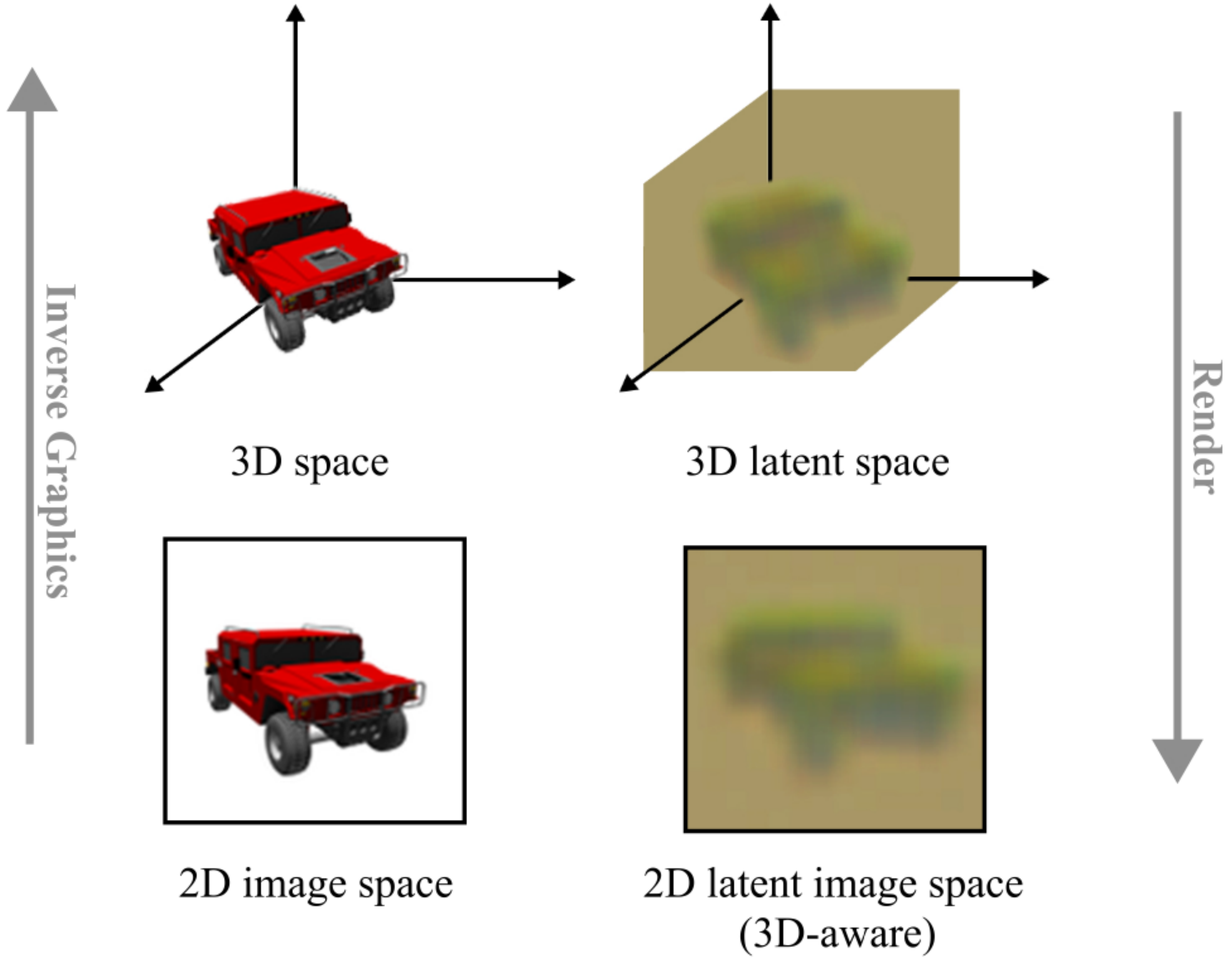
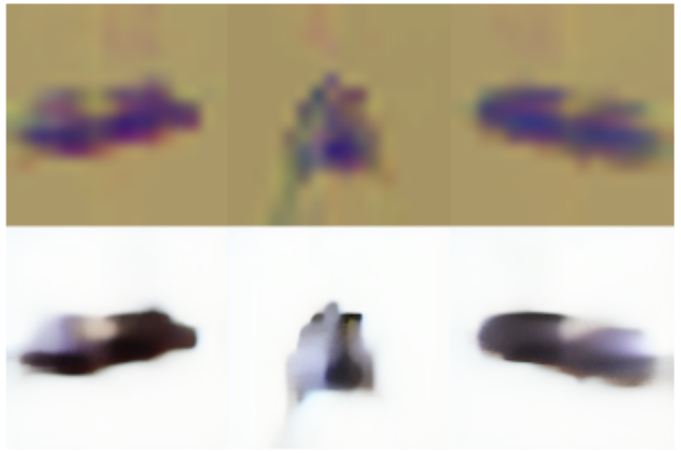


Figure 1. **3D-aware latent space.** We draw inspiration from the relationship between the 3D space and image space and introduce the idea of a 3D latent space. We propose a 3D-aware autoencoder that encodes images into a 3D-aware (2D) latent image space, in which we train our scene representations.

3D-aware latent space





(a) Standard AE

Figure 1. **3D-aware latent space.** We draw inspiration from the relationship between the 3D space and image space and introduce the idea of a 3D latent space. We propose a 3D-aware autoencoder that encodes images into a 3D-aware (2D) latent image space, in which we train our scene representations.

How to learn a scene in a 3D-aware latent space?

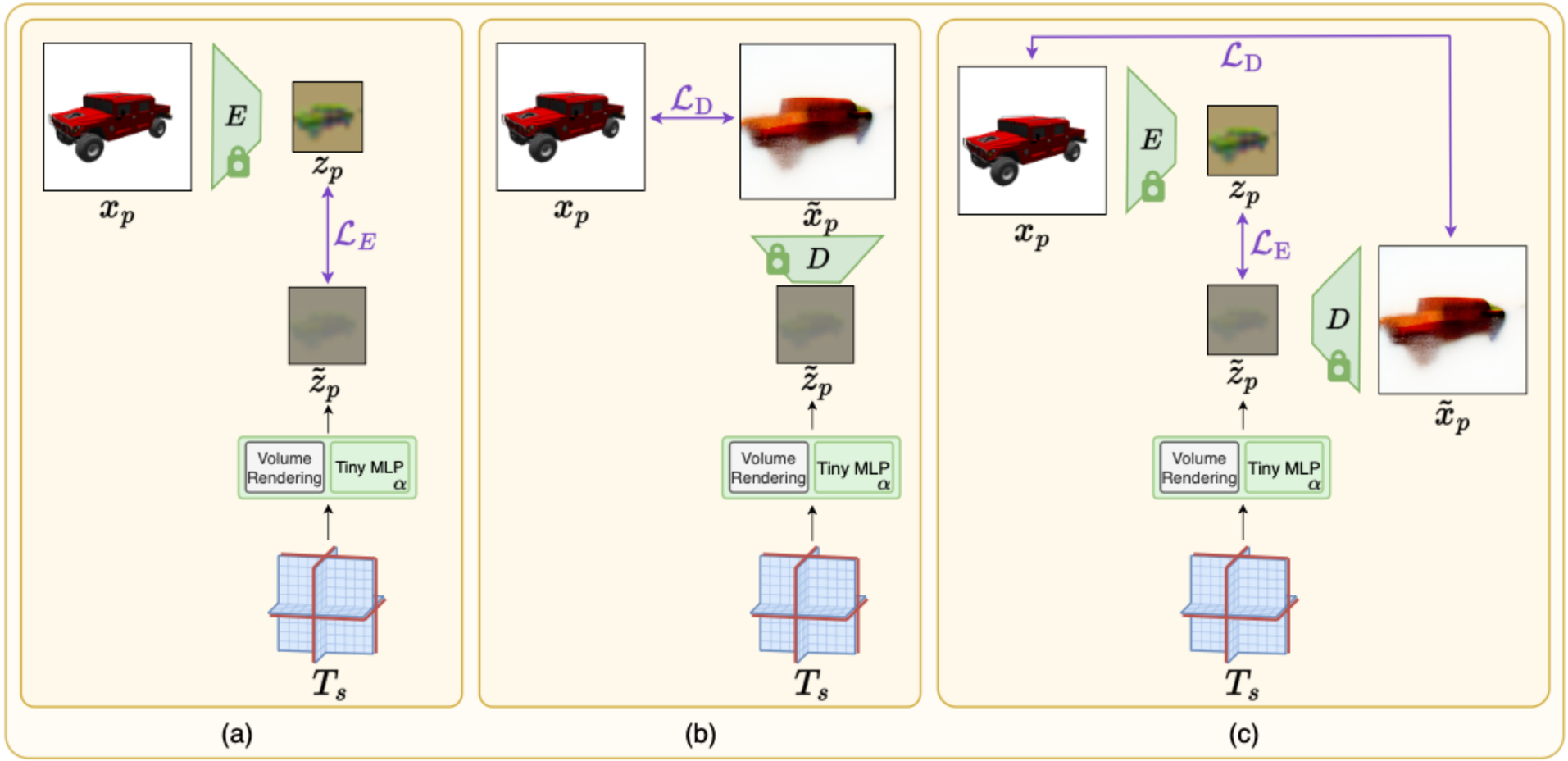


Figure 2. **Methods for learning scenes in a 3D-aware latent space.** Diagrams for (a) Encode-Scene, (b) Decode-Scene, and (c) Encode-Decode-Scene, the proposed methods to train Tri-Plane scene representations in a 3D-aware latent space.

How to train a 3D-aware autoencoder?

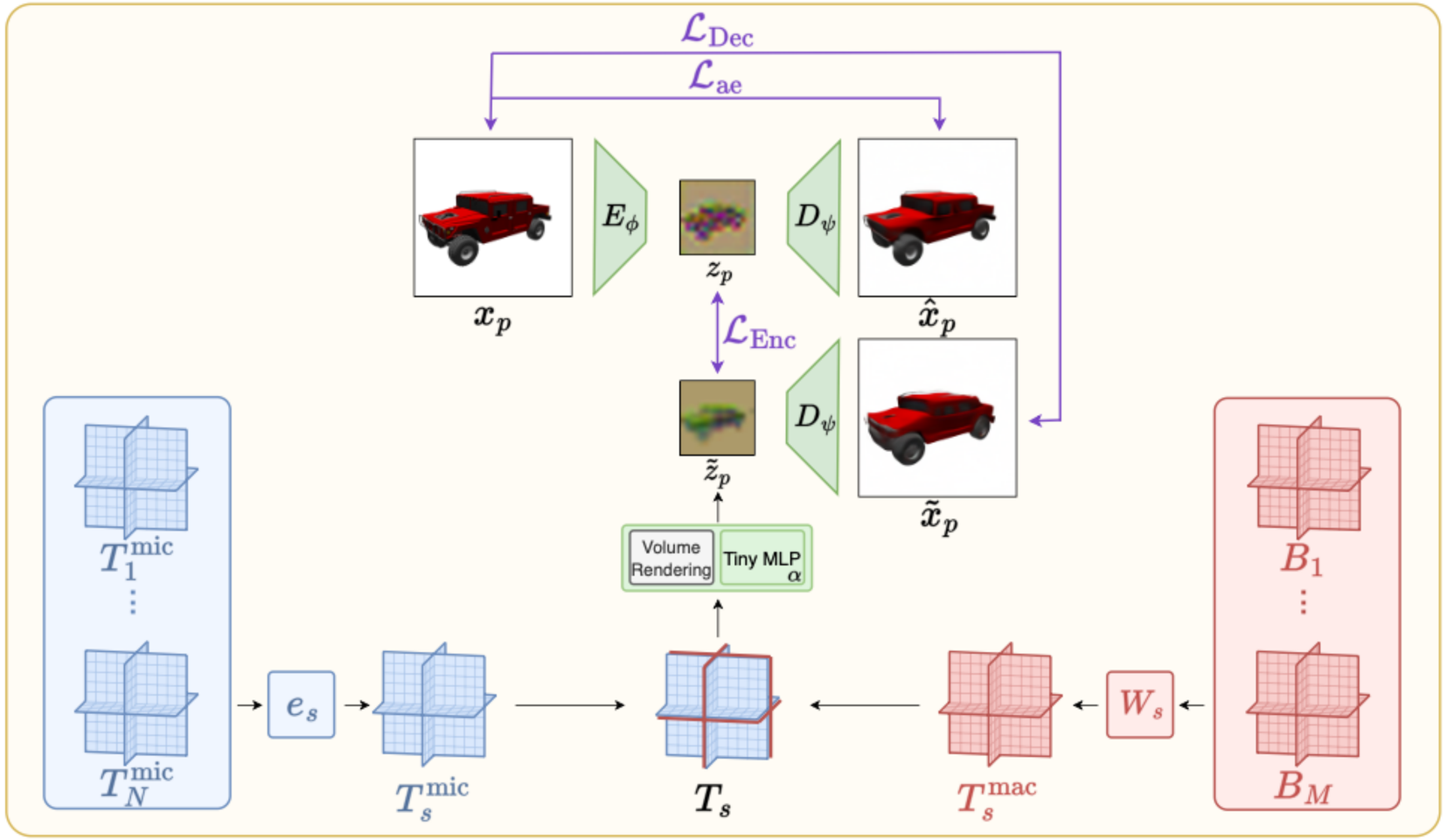
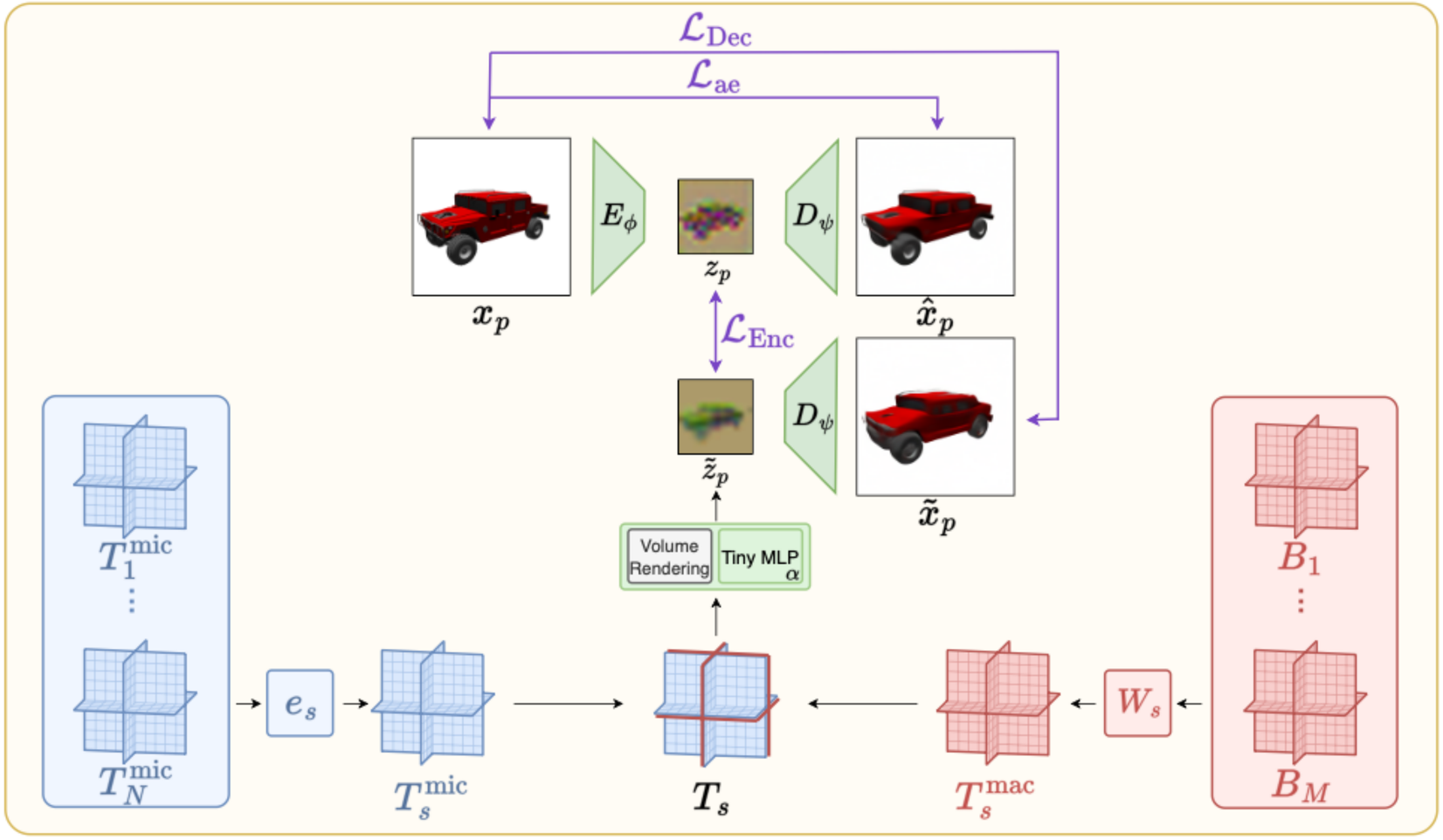


Figure 5. **3Da-AE training.** We learn a 3D-aware latent space by regularizing its training with 3D constraints. To this end, we jointly train the encoder E_ϕ, the decoder D_ψ and N scenes in this latent space. For each scene s, we learn a Tri-Planes representation T_s, built from the concatenation of local Tri-Planes T_s^{mic} and global Tri-Planes T_s^{mac}. T_s^{mic} is retrieved via a one-hot vector e_s from a set of scene-specific planes stored in memory. T_s^{mac} is computed from a summation of M globally shared Tri-Planes, weighted with weights W_s.

How to train a 3D-aware autoencoder?



$$\min_{\phi,\psi,\alpha,T} \lambda_{\text{ae}}\mathcal{L}_{\text{ae}}(\phi,\psi) + \lambda_{\text{Enc}}\mathcal{L}_{\text{Enc}}(\phi,\alpha,T) + \lambda_{\text{Dec}}\mathcal{L}_{\text{Dec}}(\psi,\alpha,T)$$

with

$$\begin{cases} \mathcal{L}_{\text{ae}}(\phi,\psi) = \mathbb{E}_{x_p}\|x_p - D_\psi(E_\phi(x_p))\| , \\ \mathcal{L}_{\text{Enc}}(\phi,\alpha,T) = \mathbb{E}_{x_p}\|E_\phi(x_p) - \mathcal{R}_\alpha(T,p)\| , \\ \mathcal{L}_{\text{Dec}}(\psi,\alpha,T) = \mathbb{E}_{x_p}\|x_p - D_\psi(\mathcal{R}_\alpha(T,p))\| , \end{cases}$$

Results

3D-aware autoencoder

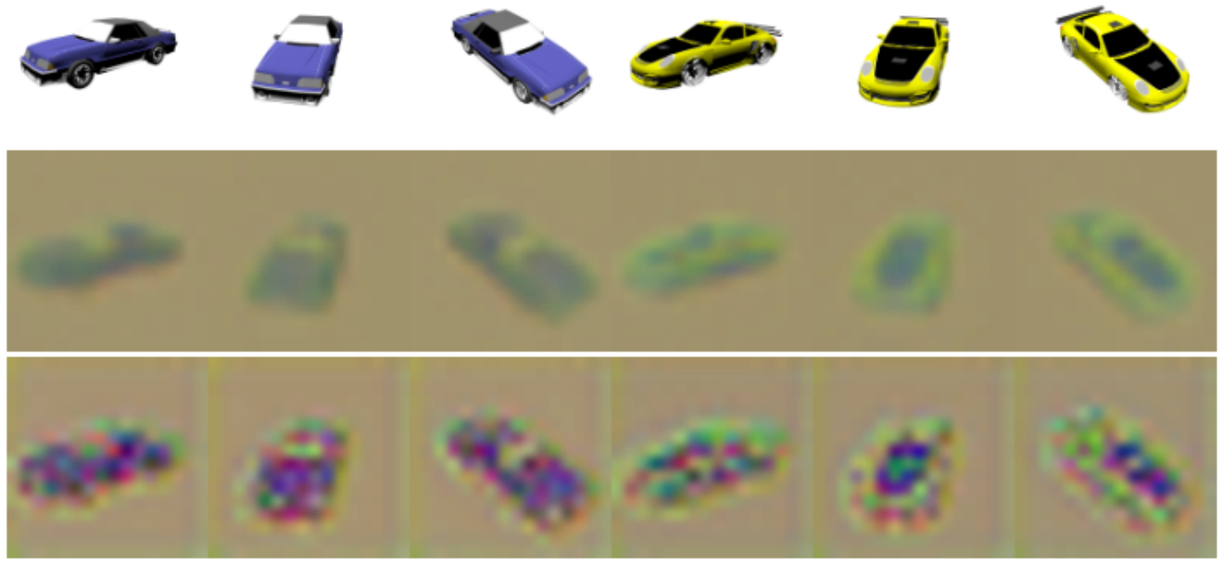
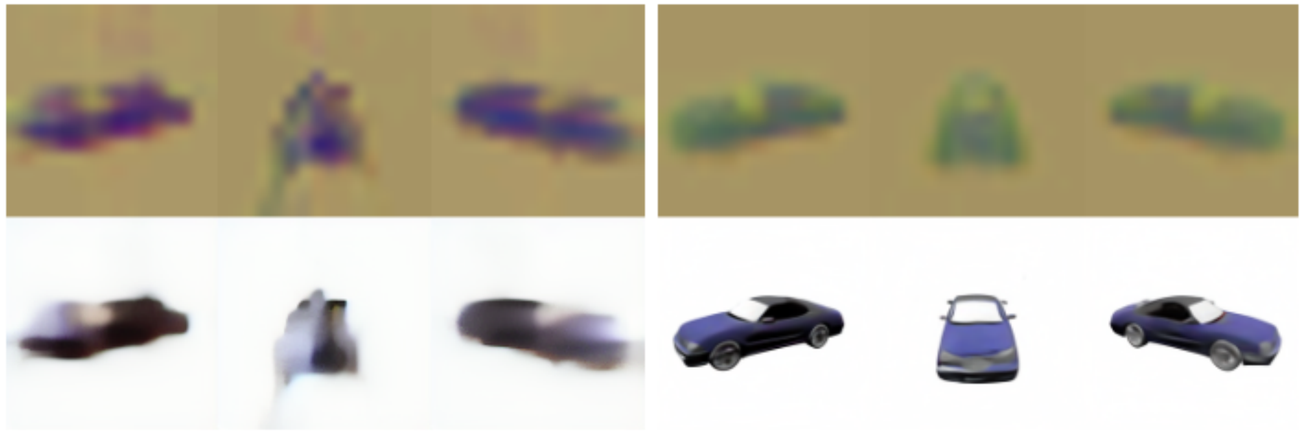


Figure 3. **Latent space comparison.** Top: ground truth image. Middle: latent image obtained with the 3D-aware encoder. Bottom: latent image obtained with the baseline encoder. Qualitative results show that our 3D-aware encoder better preserves 3D consistency and geometry in the latent space.



(a) Standard AE (b) 3D-aware AE

Figure 4. **Autoencoder comparison**. Visualization of Tri-Planes renderings and their corresponding decodings after learning scenes in the latent space of a standard AE and our 3D-aware AE. All Tri-Planes are trained using the Encode-Scene pipeline.

Renderings

Experiment	Latent Space	Micro-Planes	Macro-Planes	Train scenes	Exploit scenes
Ours-Micro	✓	✓	✗	26.52	26.95
Ours-Macro	✓	✗	✓	25.67	26.10
Tri-Planes-Macro (RGB)	✗	✗	✓	27.84	28.00
Tri-Planes (RGB)	✗	✓	✗	**28.24**	28.40
Ours-No-Prior	✓	✓	✓	27.72	28.13
Ours	✓	✓	✓	28.05	**28.48**

Table 2. **Quality comparison.** Average PSNR demonstrated by our method with a comparison to Tri-Planes and ablations of our pipeline. All metrics are computed on never-seen test views. Here, we consider $N_{\text{train}} = 500$, $N_{\text{exploit}} = 100$, and $M = 50$. For compute constraints, Tri-Planes metrics are averaged on 50 scenes.

(a) Tri-Planes (RGB) (b) Ours (c) Ground truth

Figure 8. **Visual comparison**. Visual comparison of novel view synthesis quality for our method and Tri-Planes (RGB).

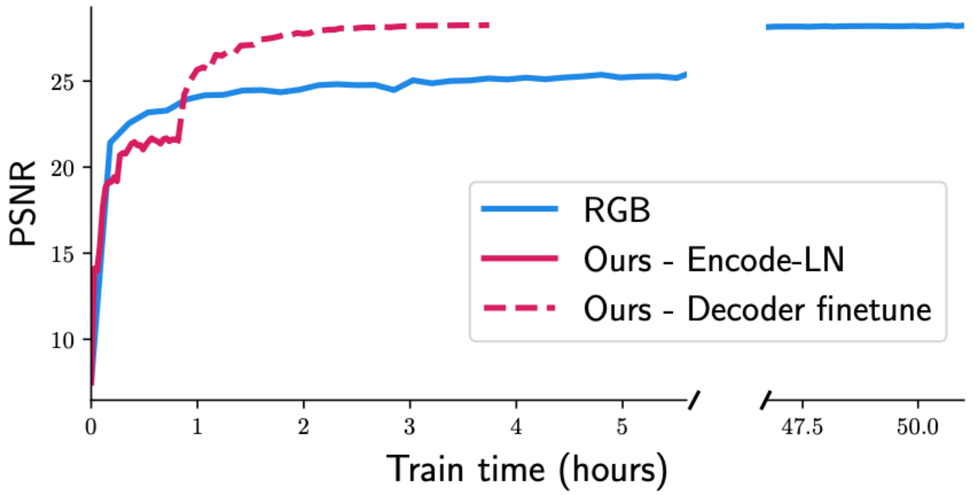


Figure 6. **Quality evolution.** Evolution of the average test-view PSNR demonstrated in the exploit phase of our method compared to RGB Tri-Planes ($N_{\text{exploit}} = 100$). Our method achieves comparable quality in less training time.

Resource costs

	t_{scene} (min)	$t^{\text{eff}}_{\text{scene}}$ (min)	m_{scene} (MB)	$m^{\text{eff}}_{\text{scene}}$ (MB)	Rendering Time (ms)	Rendering Resolution
Encoder	—	—	0	0.13	—	—
Decoder	—	—	0	0.19	9.7	128×128
Tri-Planes (RGB)	32	32	1.5	1.5	23.3	128×128
Our method	2	4.5	0.48	0.84	11.0	128×128

Table 1. **Cost comparison.** Per scene cost comparison with Tri-Planes trained in the image space. Here, we consider $N_{\text{train}} = 500$, $N_{\text{exploit}} = 1000$, $t_{\text{EC}} = 40$ hours, $M = 50$, $F^{\text{mac}} = 22$. Our method reduces the effective training time by 86% per scene, and the effective memory cost by 44% per scene.

Resource costs

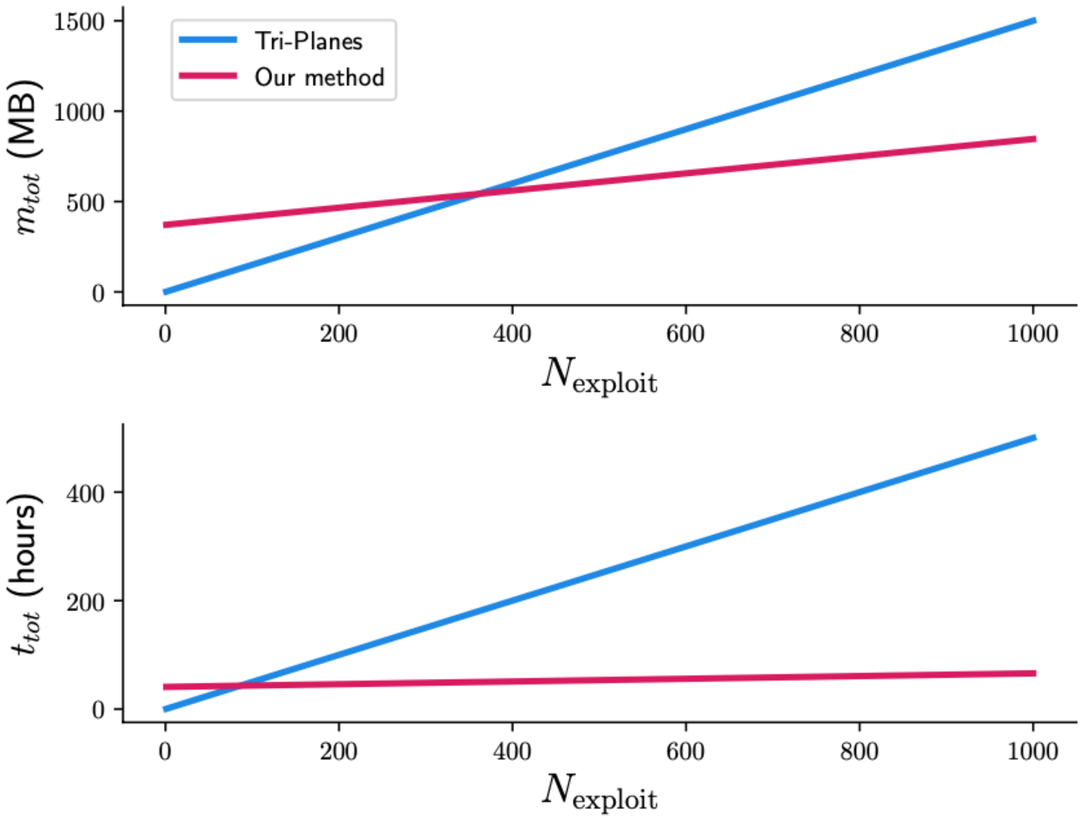


Figure 7. **Cost evolution.** Total memory and train time evolution when scaling the number of trained scenes N_{exploit}. The entry training cost t_{EC} and memory costs m_{EC} are taken into account. Our method demonstrates more favorable scalability properties as compared to Tri-Planes (RGB).

Thank you !

https://3da-ae.github.io/

Exploring 3D-aware Latent Spaces for Efficiently Learning Numerous Scenes

Antoine Schnepf[*1,3], Karim Kassab[*1,2],
Jean-Yves Franceschi[1], Laurent Caraffa[2], Flavian Vasile[1], Jeremie Mary[1],
Andrew Comport[3], Valérie Gouet-Brunet[2]

[*] Equal Contributions
[1] Criteo AI Lab, Paris, France
[2] LASTIG, Université Gustave Eiffel, IGN-ENSG, F-94160 Saint-Mandé
[3] Université Côte d'Azur, CNRS, I3S, France

K. Kassab, March 2024