# Part2 D-NeRF 실습

Presenter: 형준하

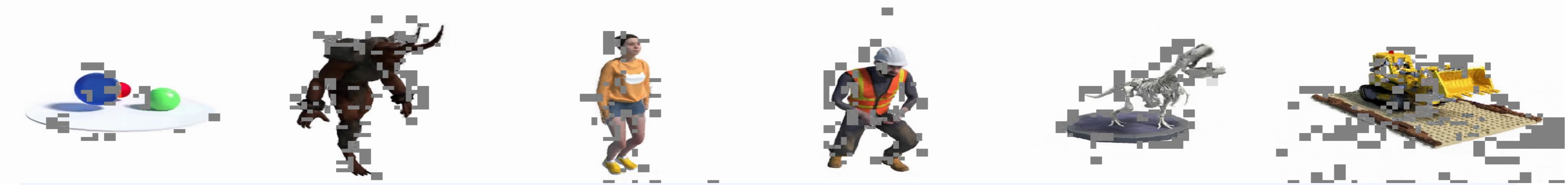
# D-NeRF: Neural Radiance Fields for Dynamic Scenes

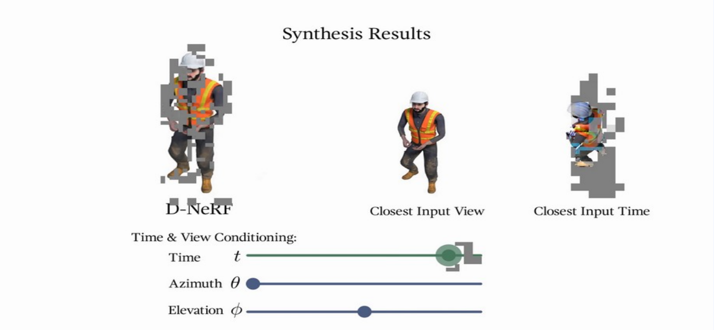
CVPR 2021

Albert Pumarola[1] Enric Corona[1] Gerard Pons-Moll[2,3] Francesc Moreno-Noguer[1]
[1]Institut de Robòtica i Informàtica Industrial, CSIC-UPC
[2]University of Tübingen
[3]Max Planck Institute for Informatics

# Introduction

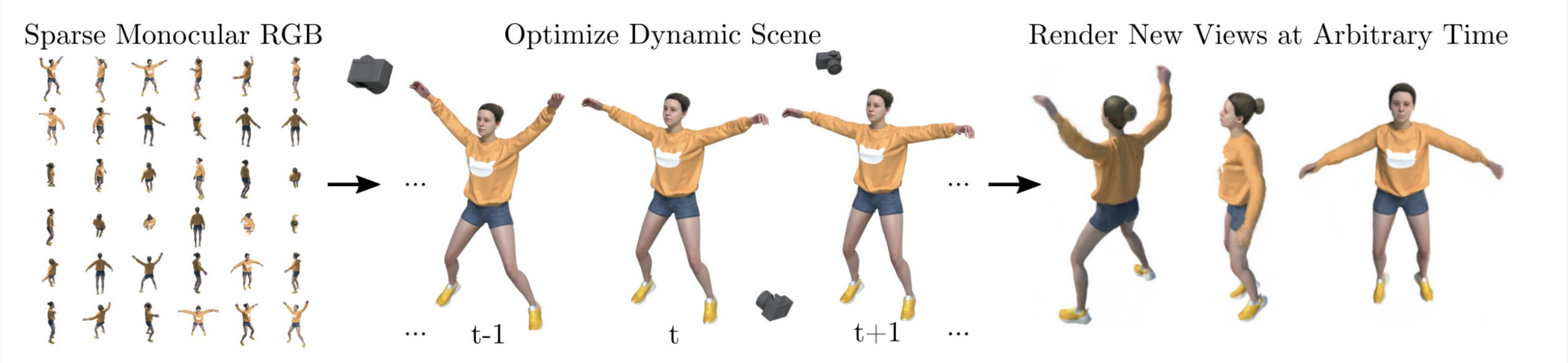
Task : Novel View Synthesis(NVS) or View Interpolation of Dynamic scene using monocular rgb camera





# Introduction

- NeRF cannot model dynamic and deformable scenes
- D-NeRF considers time component  as input
  - NeRF:
  - D-NeRF:
- Contribution
  - Only require single camera
  - Do not need to pre-compute 3D reconstruction
  - Can be trained end-to-end



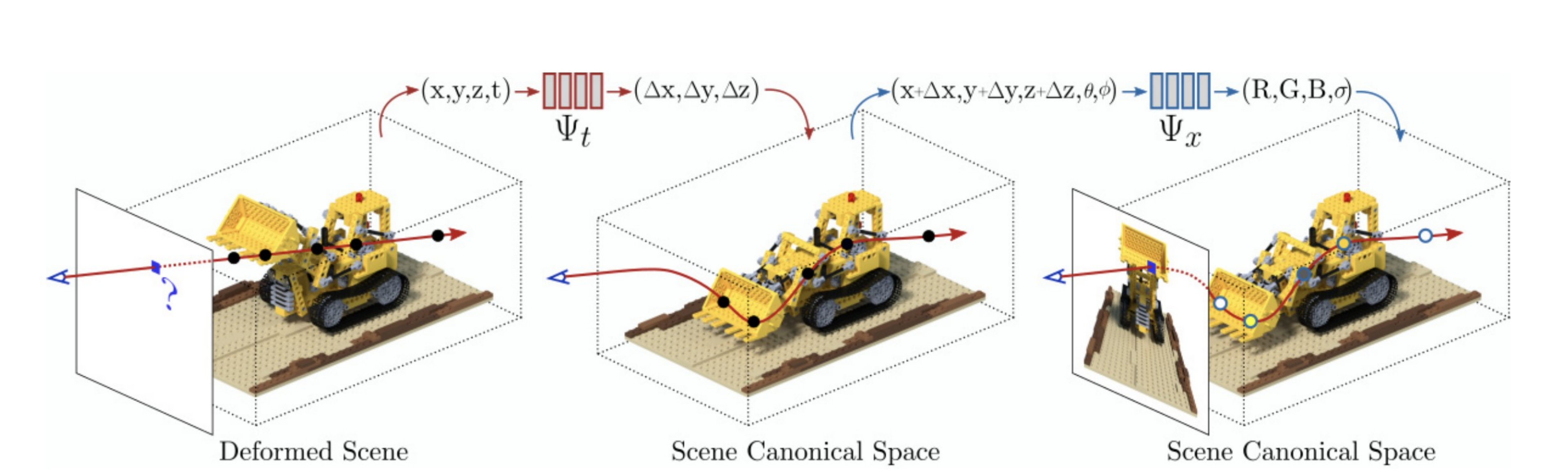
# Method

- Naïve approach:
  - Cannot exploit temporal redundancy
- Idea: Objects can move and deform, but typically do not appear or disappear
  - Define canonical scene , and represent other timesteps by deformation

Canonical scene
t=0

# Method

- Naïve approach:
  - Cannot exploit temporal redundancy
- Idea: Objects can move and deform, but typically do not appear or disappear
  - Define canonical scene , and represent other timesteps by deformation



# Camera Intrinsic and Extrinsic Parameters

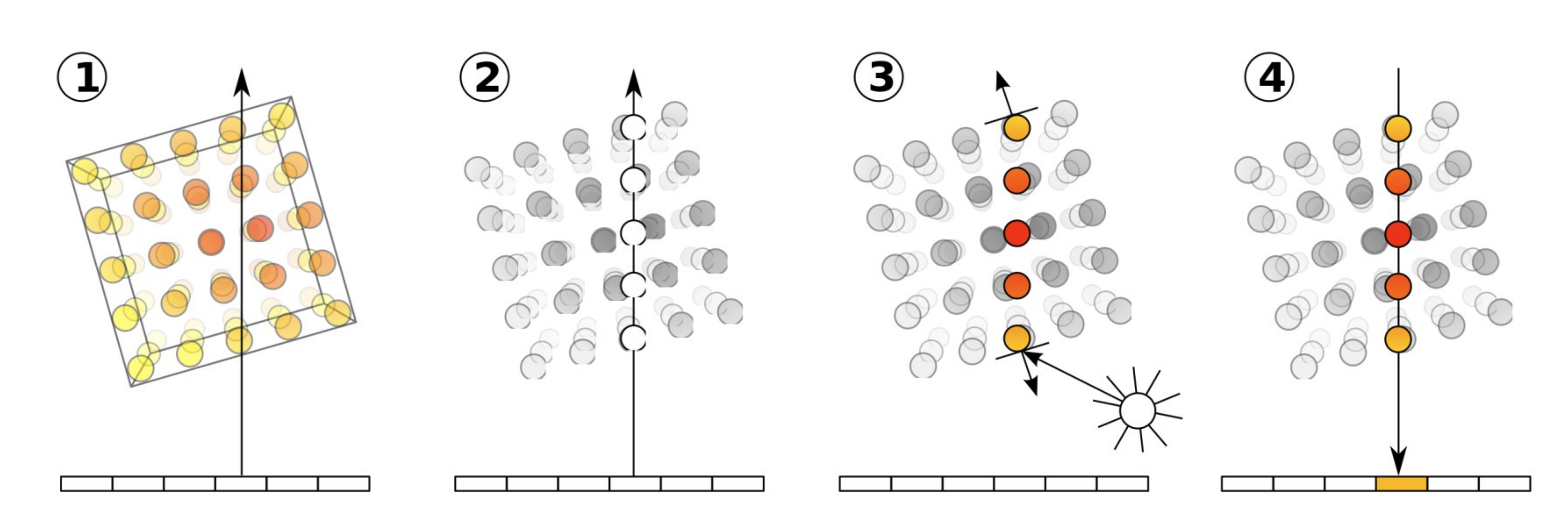
- Naïve approach:
  - Cannot exploit temporal redundancy
- Idea: Objects can move and deform, but typically do not appear or disappear
  - Define canonical scene , and represent other timesteps by deformation
- D-NeRF
  - Positional encoding also for

$$\Psi_t(\mathbf{x}, t) = \begin{cases} \Delta\mathbf{x}, & \text{if } t \neq 0 \\ 0, & \text{if } t = 0 \end{cases}$$

# Method

- Neural Radiance Fields

$$F_\Theta : (\mathbf{x}, \mathbf{d}) \rightarrow (\mathbf{c}, \sigma)$$



# Method

- Volume rendering is trivially differentiable

Rendering model for ray r(t) = o + td:

$$C \approx \sum_{i=1}^{N} T_i \alpha_i c_i$$

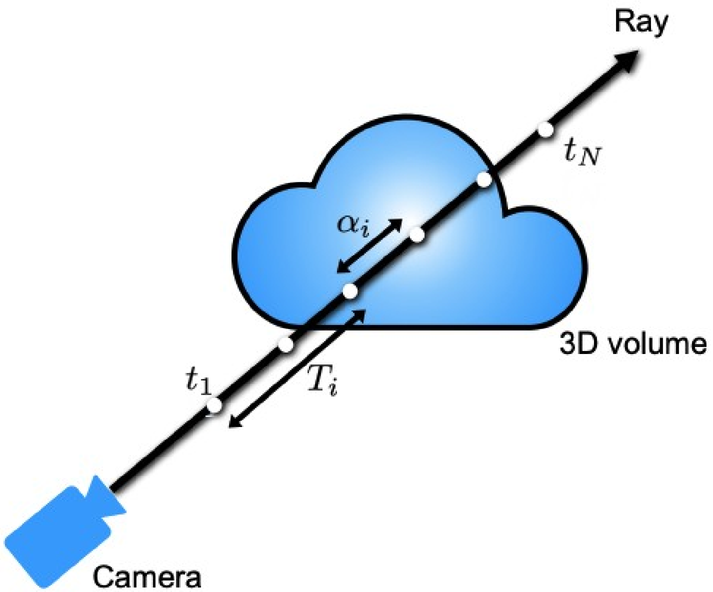
weights ($T_i$), colors ($c_i$)

How much light is blocked earlier along ray:

$$T_i = \prod_{j=1}^{i-1} (1 - \alpha_j)$$

How much light is contributed by ray segment *i*:

$$\alpha_i = 1 - e^{-\sigma_i \delta t_i}$$ ← Density * Distance Between Points



# Method

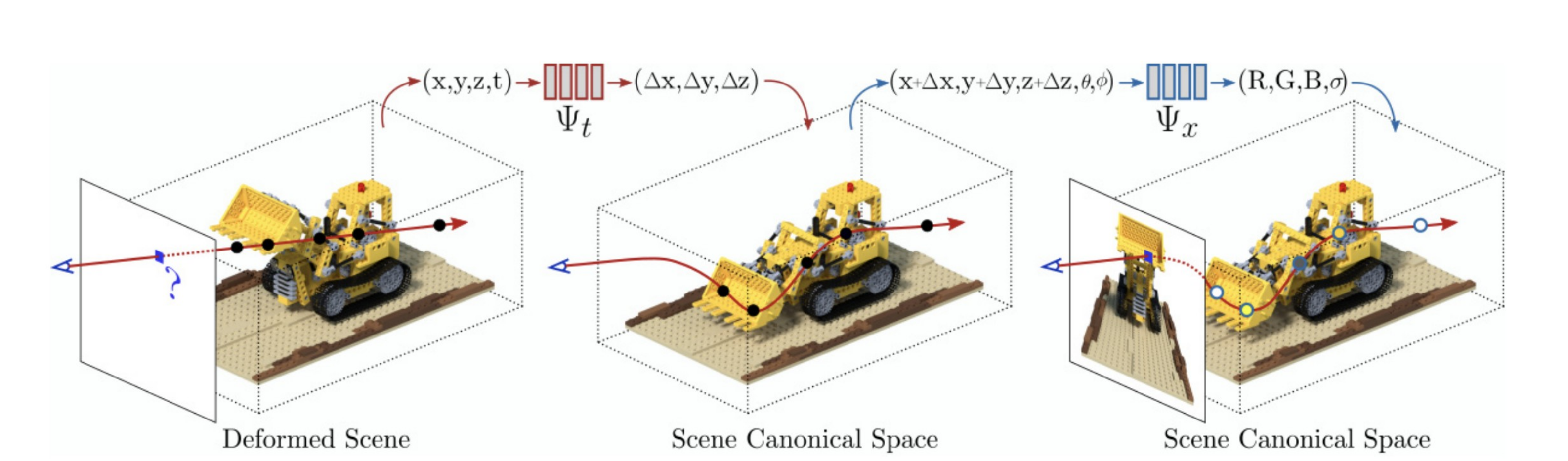
$$C'(p,t) = \sum_{n=1}^{N} \mathcal{T}'(h_n,t)\alpha(h_n,t,\delta_n)\mathbf{c}(\mathbf{p}(h_n,t),\mathbf{d}), \quad \textbf{(6)}$$

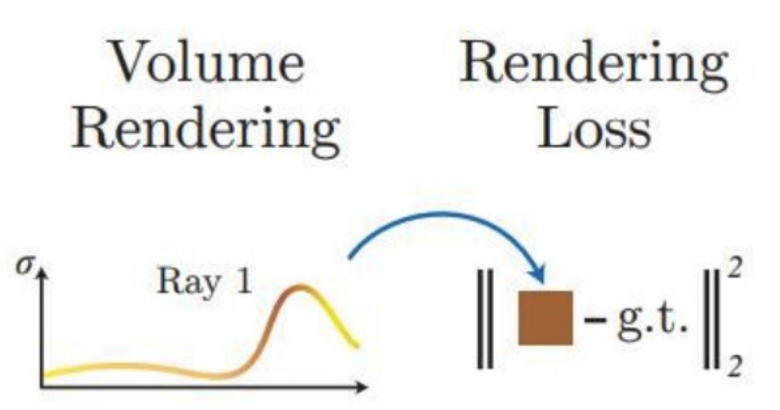
$$\textbf{where} \quad \alpha(h,t,\delta) = 1 - \exp(-\sigma(\mathbf{p}(h,t))\delta), \quad \textbf{(7)}$$

$$\textbf{and} \quad \mathcal{T}'(h_n,t) = \exp\left(-\sum_{m=1}^{n-1}\sigma(\mathbf{p}(h_m,t))\delta_m\right), \quad \textbf{(8)}$$

# Method

- Optimize with gradient descent on rendering loss




$$\mathcal{L} = \frac{1}{N_s} \sum_{i=1}^{N_s} \left\| \hat{C}(p,t) - C'(p,t) \right\|_2^2$$

# Results

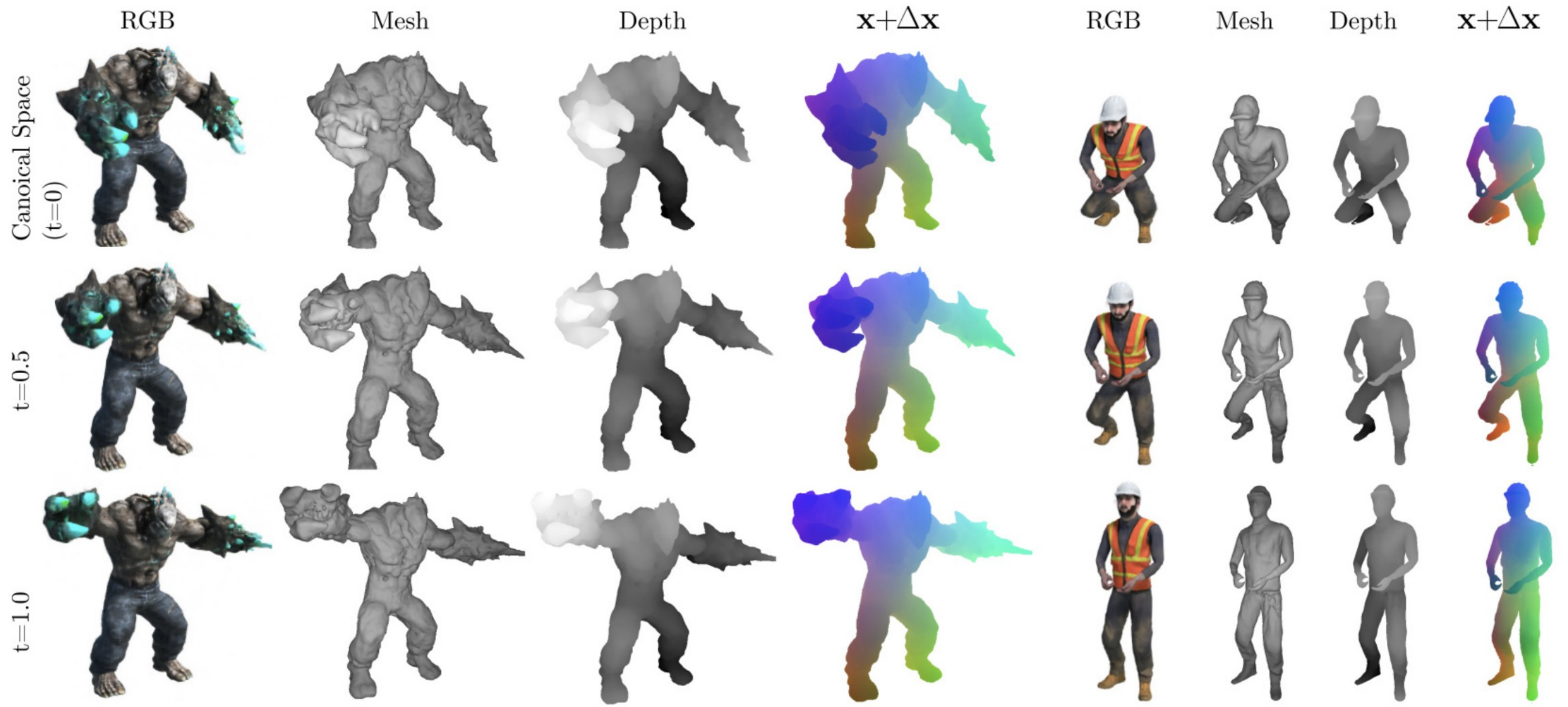



Figure 4: **Visualization of the Learned Scene Representation.** From left to right: the learned radiance from a specific viewpoint, the volume density represented as a 3D mesh and a depth map, and the color-coded points of the canonical configuration mapped to the deformed meshes based on $\Delta\mathbf{x}$. The same colors on corresponding points indicate the correctness of such mapping.

# Results

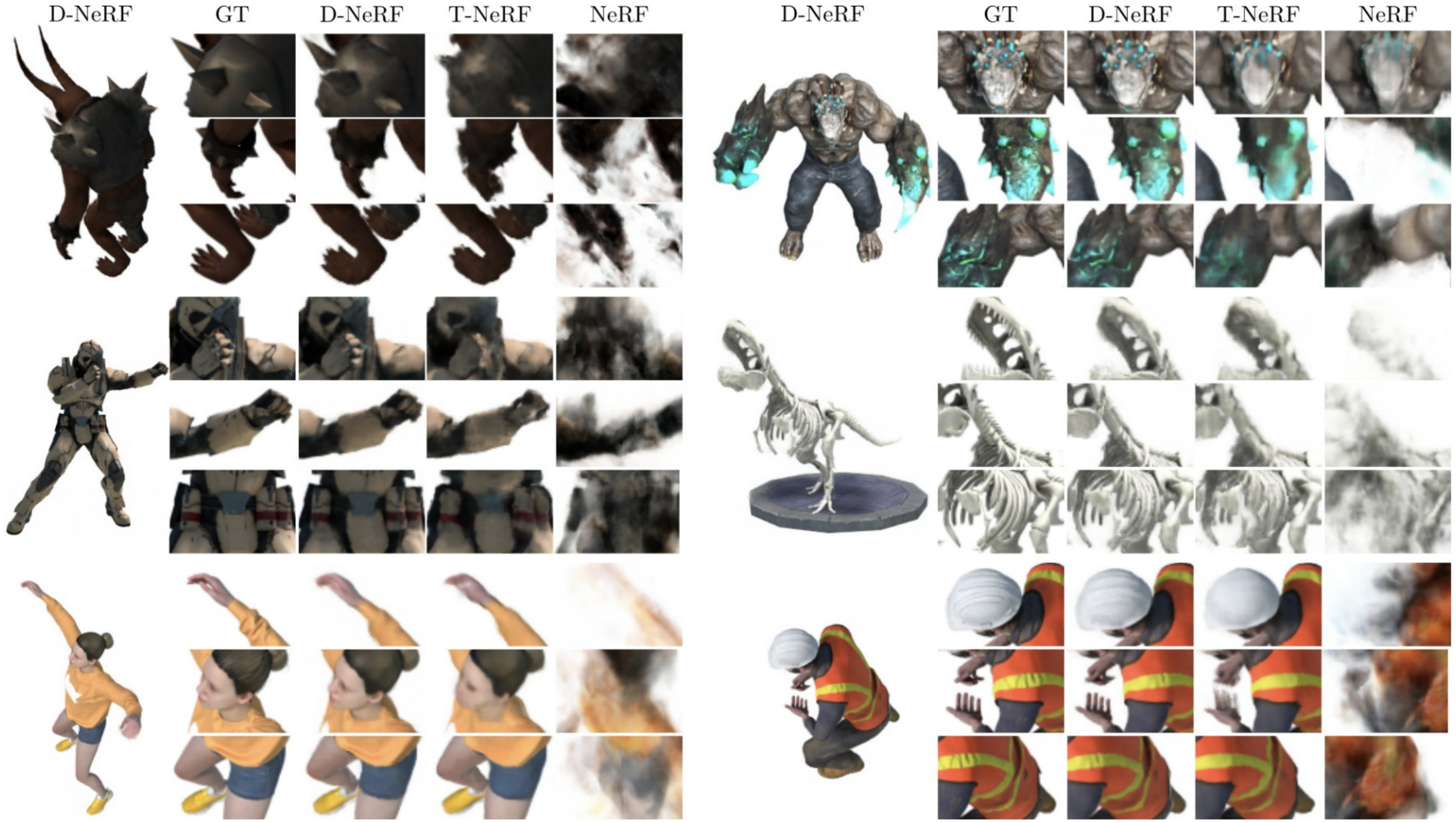



Figure 6: **Qualitative Comparison.** Novel view synthesis results of dynamic scenes. For every scene we show an image synthesised from a novel view at an arbitrary time by our method, and three close-ups for: ground-truth, NeRF, T-NeRF, and D-NeRF (ours).

# Results

| | Hell Warrior | | | | Mutant | | | | Hook | | | | Bouncing Balls | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | MSE↓ | PSNR↑ | SSIM↑ | LPIPS↓ | MSE↓ | PSNR↑ | SSIM↑ | LPIPS↓ | MSE↓ | PSNR↑ | SSIM↑ | LPIPS↓ | MSE↓ | PSNR↑ | SSIM↑ | LPIPS↓ |
| NeRF | 44e-3 | 13.52 | 0.81 | 0.25 | 9e-4 | 20.31 | 0.91 | 0.09 | 21e-3 | 16.65 | 0.84 | 0.19 | 94e-4 | 20.26 | 0.91 | 0.2 |
| T-NeRF | 47e-4 | 23.19 | 0.93 | 0.08 | 8e-4 | 30.56 | 0.96 | 0.04 | 18e-4 | 27.21 | 0.94 | **0.06** | 16e-5 | 37.81 | 0.98 | 0.12 |
| D-NeRF | **31e-4** | **25.02** | **0.95** | **0.06** | **7e-4** | **31.29** | **0.97** | **0.02** | **11e-4** | **29.25** | **0.96** | 0.11 | **12e-5** | **38.93** | **0.98** | **0.1** |

| | Lego | | | | T-Rex | | | | Stand Up | | | | Jumping Jacks | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | MSE↓ | PSNR↑ | SSIM↑ | LPIPS↓ | MSE↓ | PSNR↑ | SSIM↑ | LPIPS↓ | MSE↓ | PSNR↑ | SSIM↑ | LPIPS↓ | MSE↓ | PSNR↑ | SSIM↑ | LPIPS↓ |
| NeRF | 9e-3 | 20.30 | 0.79 | 0.23 | 3e-3 | 24.49 | 0.93 | 0.13 | 1e-2 | 18.19 | 0.89 | 0.14 | 1e-2 | 18.28 | 0.88 | 0.23 |
| T-NeRF | **3e-4** | **23.82** | **0.90** | **0.15** | 9e-3 | 30.19 | 0.96 | 0.13 | 7e-4 | 31.24 | 0.97 | 0.02 | 6e-4 | 32.01 | 0.97 | 0.03 |
| D-NeRF | 6e-4 | 21.64 | 0.83 | 0.16 | **6e-3** | **31.75** | **0.97** | **0.03** | **5e-4** | **32.79** | **0.98** | **0.02** | **5e-4** | **32.80** | **0.98** | **0.03** |

Table 1: **Quantitative Comparison.** We report MSE/LPIPS (lower is better) and PSNR/SSIM (higher is better).