Tensor4D : Efficient Neural 4D Decomposition for High-fidelity Dynamic Reconstruction and Rendering
Supplementary Material

Ruizhi Shao, Zerong Zheng, Hanzhang Tu, Boning Liu, Hongwen Zhang, Yebin Liu
Tsinghua University

This Supplementary Material provides additional details about network training and the camera system used in our paper. We also add more experimental results that are not included in the main paper due to limited space.

In Section A, we introduce our sparse multi-view camera system and the captured video sequences. Then in Section B, we provide more implementation details of our approach. In Section C, we present additional ablation study and more results of our method. Please see our Supplemental Video for more visualization.

A. Dataset and Camera System

In our experiment, we apply our method to two sparse-view settings: 1) a front-view imaging system consisting of 4 static RGB cameras that are distributed in front of the dynamic objects, and 2) a circular imaging system which includes 12 static RGB cameras surrounding the dynamic objects.

A.1. Front-view imaging system

Our front-view imaging system includes 4 static RGB cameras which are fixed at the corners of the LookingGlass monitor [1]. Each camera can capture images at 30fps with a resolution of 1024×1024. The cameras are synchronized and well-calibrated. To capture testing views for quantitative evaluation, we add two cameras in the center of the LookingGlass monitor. Using this system, we collect several video sequences capturing various human poses, gestures, expression, and objects interaction. We select 100 consecutive frames from each video sequence to build the training and evaluation dataset. To achieve holographic display on the LooKingGlass monitor, we render a dense 3D light field with 45 novel views along the horizontal center line at the middle of the monitor.

A.2. Circular Imaging System

Our circular imaging camera system contains 12 static RGB cameras which are installed on the ring cage. The camera we used is the same as the cameras in the front-view imaging system. We capture a dancing actor wearing a gauze dress for qualitative presentations.

A.3. Segmentation

Since we focus on the dynamic objects in the scene, we adopt BackGroundMattingV2 [3] to segment the foreground objects in the images for our dataset.

B. More Implementation Details

For our 4D tensor decomposition, we define the 4D Plane module which contains 18 feature planes. Nine of them are low-resolution planes and the others are high-resolution planes. Each low-resolution feature plane has 32 channels and their resolution is 128×128. Each high-resolution feature plane has 16 channels and their resolution is 512×512. For 3D tensor decomposition, we define the 3D Plane module with 6 feature planes. Three feature planes are low-resolution (128×128) and have 32 channels. The other 3 feature planes are high-resolution (512×512) and have 16 channels.

In the multi-view setting, our geometry MLP E_g has 3 layers and all layers have 256 hidden dimensions except for the first layer. The input of the first layer has 486 dimensions, where $(32 + 16) \times 9$ dimensions are queried from feature planes, $3 + 3 \times 14$ dimensions are the positional encoding of (x, y, z) and $1 + 1 \times 8$ dimensions are the positional encoding of t. The color MLP E_c has 3 layers. The input of the first layer has 283 dimensions, where 256 dimensions are the output of geometry MLP and $3 + 3 \times 8$ are the positional encoding of view direction. The second layer has 256 hidden dimensions and the final layer outputs the 3-dimensions vector for the RGB values.

Similarly, in the monocular setting, the flow MLP E_f has 3 layers and all layers have 256 hidden dimensions except for the first layer. The input of the first layer has 342 dimensions, where 32×9 dimensions are queried from feature planes, $3 + 3 \times 14$ dimensions are the positional encoding of (x, y, z), and $1 + 1 \times 8$ dimensions are the positional

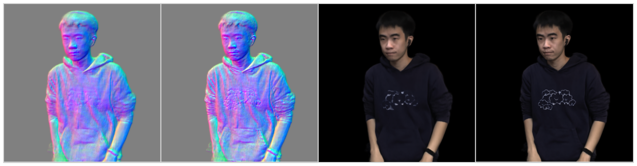
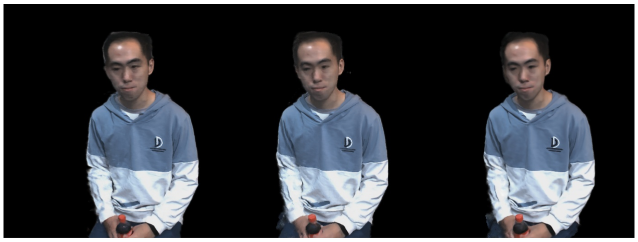


Figure A1. Coarse-to-fine detail improvements on geometry normal and novel rendering (coarse normal, fine normal, coarse rendering, fine rendering from left to right).



2 Views Input **3 Views Input** **4 Views Input**

Figure A2. The rendering results under different numbers of input views.

Table A1. Quantitative comparisons of training efficiency. Here 5k and 40k are the number of iterations.

Method	Sequence1-thz (PSNR)		Lego (PSNR)	
	5k	40k	5k	40k
D-NeRF [4]	19.33	23.11	19.47	20.42
TiNeuVox [2]	21.56	22.72	**23.57**	24.84
Neus-T [5]	19.07	22.41	-	-
Ours	**23.41**	**27.05**	20.16	**26.89**

encoding of t. The geometry MLP E_g has 3 layers and the last three layers have 256 hidden dimensions. The input of the first layer has 189 dimensions, where $(32+16)\times 3$ are queried from feature planes and $3+3\times 14$ dimensions are the positional encoding of (x,y,z). For color rendering, We adopt the same color MLP that we used in the multi-view setting.

For fair comparisons, we adopt the same learning rate and batch size for our method and existing methods. In the monocular setting, we set the batch size to 512 and the learning rate to $5e-4$. We also decay the learning rate by 0.005 after every 10k steps. In the multi-view setting, we adopt the same learning rate setting and the training batch size is 1024. For training loss, we set λ_c to 1.0, λ_r to 0.2 and λ_e to 0.01.

C. Additional Experiment

C.1. Coarse-to-fine strategy

We qualitatively ablate the coarse-to-fine strategy in our method. As shown in Fig. A1, the geometry and rendering results are coarse and smooth at the coarse level. At the fine level, the learned feature planes focus more on details and produce high-fidelity rendering results.

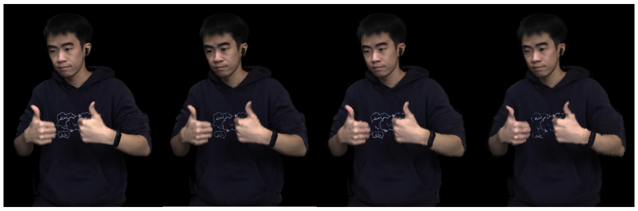


Figure A3. Training results on the sequence with different number of frames (30, 60, 100, 200 from left to right).

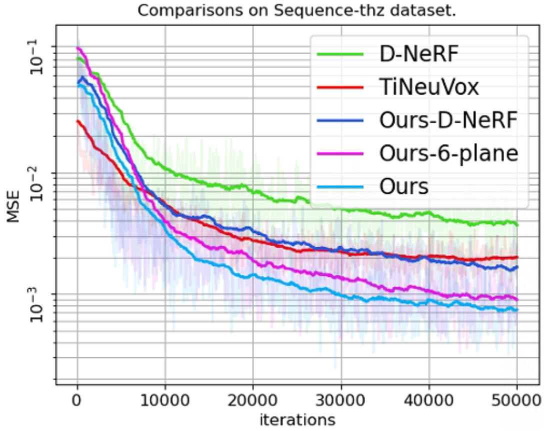


Figure A4. The curves of training loss.

C.2. Number of views

We ablate our method using different numbers of views as input. The results are shown in Fig. A2. Our method can achieve photo-realistic rendering even with only 2 input views, which indicates the robustness and generalization ability of our method.

C.3. Time of video.

We qualitatively ablate the ability of our method by training with different numbers of temporal frames. We train our method on the "thumbsup" multi-view sequence with 30, 60, 100, and 200 frames. We keep the same training iterations (100k), and the novel view rendering results are shown in Fig. A3.

C.4. Training efficiency.

We compare the training efficiency of our method with existing methods including NeRF-T [4], D-NeRF [4], TiNeuVox [2] and Neus-T. In the experiment, we compare PSNR values changing with different number of training iterations on both a monocular synthetic dataset (Lego) and a multi-view dataset (Sequence1-thz). For fair comparison, we also fix the batch size of the sampling rays and keep the same learning rate. As shown in Tab. A1, our 4D decomposition is the most efficient method that achieve the best trade-off between complexity and quality. The rendering quality can be much higher given with sufficient training iterations. The curves of training loss for several methods are additionally plotted in Fig. A4, which also demonstrates Tensor4D's most efficient convergence.

References

[1] The looking glass holographic display. https://lookingglassfactory.com/. 1

[2] Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. Fast dynamic radiance fields with time-aware neural voxels. In *arXiv preprint arXiv:2205.15285*, 2022. 2

[3] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sengupta, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Real-time high-resolution background matting. *arXiv*, pages arXiv–2012, 2020. 1

[4] Albert Pumarola, Enric Corona, and and Francesc Moreno-Noguer Gerard Pons-Moll. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 2

[5] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*, 2021. 2