WavePlanes: A Compact Wavelet Representation for Dynamic Neural Radiance Fields (Supplementary Materials)

Adrian Azzarelli
Visual Information Laboratory
University of Bristol
a.azzarelli@bristol.ac.uk

Nantheera Anantrasirichai
Visual Information Laboratory
University of Bristol
n.anantrasirichai@bristol.ac.uk

David R Bull
Visual Information Laboratory
University of Bristol
dave.bull@bristol.ac.uk

1. NeRF Background

Rendering Equation. Given set of color and density samples, $\mathbf{c}$ and σ, indexed by i along a ray, we use the same equation as in NeRF [7], Equation 1, to render each pixel of a novel view.

$$C = \sum_{i=1}^{N} \exp(-\sum_{j=1}^{i-1} \sigma_j \delta_j)(1 - \exp(\sigma_i \delta_i))\mathbf{c}_i, \tag{1}$$

where δ_i is the width of a volumetric sample along a ray, the first exponential represents the transmittance of sample i w.r.t. earlier samples along a ray, and the second term, 1-exp($\cdot$), is the absorption of the sample. Combined, these model the contribution of each color sample along a ray, the sum of which approximates the color of each pixel.

2. Additional Hyper Parameters

All configuration files are provided with our code: `https://github.com/azzarelli/waveplanes/`. For all experiments we use cosine annealing with a brief linearly increasing warm-up (512 steps) as our learning function. The notable differences between experiments are:

1. For the LLFF data set [6] we use 4x down sampling, with normalized device coordinates, a learning rate of 0.02, a 4-layer linear feature decoder and 40k training steps. All LLFF experiments share the same configuration file.
2. For the real DyNeRF data set [5] we use 8x down sampling to generate the IST weights and 2x down sampling to train the model for 120k steps with a learning rate of 0.01. All DyNeRF experiments share the same configuration file, however we find that the ZMM and ZAM fusion schemes work better with a 3-layer and 4-layer linear feature decoder, respectively; demonstrated in Table 1.
3. For the synthetic D-NeRF data set [8] we use 2x down sampling, normalized device coordinates and a learning rate of 0.01 for 30k training steps. For this experiment, we tune our model per-scene. Individual configuration files have been made available and differ in resolution, as discussed in the paper.

Scheme	# Layers	PSNR ↑	SSIM ↑
Coffee Martini Scene			
ZMM	3	27.40	0.8767
ZMM	4	25.80	0.8733
ZAM	3	27.51	0.8832
ZAM	4	27.17	0.8854
Flame Steak Scene			
ZMM	3	30.40	0.9271
ZMM	4	30.31	0.9312
ZAM	3	28.67	0.9312
ZAM	4	30.49	0.9331

Table 1. **Comparing 3 and 4 layered linear feature decoders** for the ZMM and ZAM fusion schemes on DyNeRF scenes.

3. Additional Results

3.1. Per-Scene Results

Real static scenes. In Table 2 we compare our model with the state-of-the-art, including the masked wavelet representation [9], for the LLFF data set. We also evaluate our model with feature length B =32 and 64. With B =32 the model consists of 27M parameters and trains in just over 1 hour, whereas a feature length of 64 uses 51M parameters and trains in under 2 hours. We selected $B = 64$ for our main experiment as it provides better quantitative and qualitative results. In Figure 1 we provide visual examples of LLFF scenes modelled with WavePlanes.

Real dynamic scenes. In Table 3 we breakdown the results from each scene in the DyNeRF data set to provide a quantitative comparison the HP, ZMM and ZAM feature fusion schemes. In Figure 2 we provide visual examples these scenes modeled with WavePlanes using the ZAM fusion scheme.

Synthetic dynamic scenes. In Table 4 we breakdown and compare the results from the D-NeRF data set for the HP, ZMM and ZAM fusion schemes. In Figure 3 we provide visual examples of each D-NeRF scene - rendered using ZAM fusion.

Per-scene compression results are provided in Table 5. We find that the proposed compression scheme performs optimally in the presence of empty space, i.e. when the wavelet coefficients are near-zero. This is best exemplified by the results from the LLFF scenes, where the model undergoes higher compression for scenes with more space. On the other hand, many of the DyNeRF scenes are captured in the same room with similar objects thus have similar model size. Moreover, the D-NeRF scenes are all object-centric with plain backgrounds therefore, in some cases, it achieves a smaller dynamic model than even TiNeuVox-S (8MB). With regards to performance after compression, for all scenes we observe a maximum PSNR and SSIM loss of around 0.8 and 0.05, respectively. Yet, similar to model size, this is less exaggerated for scenes that contain more empty space where we see as no PSNR or SSIM difference.

3.2. Visualising Planes

For the purpose of demonstration, in Figure 4 we visualize a set wavelet coefficient planes pertaining to a space-only plane with a resolution of 256×256. To provide images representing of the decomposed wavelet coefficients, we separate each level into it's horizontal, vertical and diagonal filter components and normalize the average across all features.

Feature Planes. The feature plane, $\mathbf{P}^0_{xy}$, is visualized in Figure 5 for the D-NeRF T-Rex scene.

3.3. Decomposing Static and Dynamic Components

In Figure 6, we exemplify the decomposition of our representation into static and dynamic components. To render a space-only scene (visualizing only static features), we force the condition $f_{c_t}(\mathbf{q}) = 1$ by zeroing the wavelet coefficients for all space-time planes. The space-only rendered frames are subtracted from the final render to visualize the effect of space-time features. Note that we do not use $f_{c\neq c_t}(\mathbf{q}) = 1$ to render space-time features directly as they can not be interpreted by the colour and density decoder. This adds to our interpretation of space-time features, whereby we treat them as linear transformations for the space-only features that can be interpreted as basis features. Interestingly, this behaviour is similar to dynamic NeRFs that use deformation fields to linearly transform the position of volumes in a static field. Though instead, the plane representation only modifies the density and color of volumes in time.

In Figure 7 we provide further visual comparison between the static decomposition of the HP and ZMM feature fusion schemes.

3.4. Ablation Study

N-level wavelet scaling coefficients. To produce fair comparisons with the final model we used the results in Table 6 to tune a our model for a 2, 3 and 4 level of wavelet decomposition. We find that higher levels of decomposition results in slower training time and delayed convergence.

Varying weights for regularization. The following outlines our process for tuning the weights for regularization - accomplished using the T-Rex and Hell Warrior D-NeRF scenes. We first optimized the TV weight in Table 7, where we selected a weight of 0.00001. After this we varied the SST weights. Table 8 outlines the results, showing that TV has a less of an impact on the performance. Finally, with an SST weight of 0.1 we varied the weight for ST regularization in Table 9.

Varying grid resolution. For synthetic dynamic scenes our model performs best with a spatial resolution of $H = W =$256 for high-frequency and 128 for low-frequency scenes, as discussed in the paper. Results indicating this behavior are provided in Table 10. We notice longer training times with worse quantitative results when P^1_c is significantly high in resolution, exemplified by the Lego and T-Rex scenes.

4. Paper Limitations

4.1. Quantitative Image Assessment Metrics

It is challenging to discern true dynamic performance from the variety of metrics that have been proposed, such as SSIM, D-SSIM [1], MS-SSIM [10] and LPIPS [11]. For synthetic RGBA data sets we also propose to separate foreground and background predictions during testing and validation. Considering that all these metrics evaluate stationary frames at different times, we are limited by our ability to evaluate temporal features such as smoothness and consistency. Additionally, the data sets we use do not support this type of evaluation. For instance, the D-NeRF data set is strictly provided as a set of ”teleporting” frames so could not be used for evaluating spatiotemproal smoothness using ground truth renders.

PSNR ↑

	Room	Fern	Leaves	Fortress	Orchids	Flower	T-Rex	Horns	Means
NeRF	32.64	25.17	20.92	31.16	20.36	27.40	26.80	27.45	26.50
Plenoxels [3]	30.22	25.46	21.41	31.09	20.24	27.83	26.48	27.58	26.29
TensoRF (large) [2]	32.35	25.27	21.30	31.36	19.87	28.60	26.97	28.14	26.73
Masked Wavelets* (small) [9]	31.19	25.05	21.08	30.75	19.76	27.94	26.77	27.48	26.25
Masked Wavelets* (large) [9]	31.73	25.27	21.17	31.01	20.02	28.20	27.07	27.88	26.54
K-Planes (explicit) [4]	32.72	24.87	21.07	31.34	19.89	28.37	27.54	28.40	26.78
K-Planes (hybrid) [4]	32.64	25.38	21.30	30.44	20.26	28.67	28.01	28.63	26.92
Ours-64	31.47	23.89	20.90	30.36	19.21	27.61	27.02	28.33	26.10
Ours-32	31.29	23.99	21.03	30.08	19.41	26.08	26.80	28.00	25.84
SSIM ↑									
NeRF	0.948	0.792	0.690	0.881	0.641	0.827	0.880	0.828	0.811
Plenoxels [3]	0.937	0.832	0.760	0.885	0.687	0.862	0.890	0.857	0.839
TensoRF (large) [2]	0.952	0.814	0.752	0.897	0.649	0.871	0.900	0.877	0.839
K-Planes (explicit) [4]	0.955	0.809	0.738	0.898	0.655	0.867	0.909	0.884	0.841
K-Planes (hybrid) [4]	0.957	0.828	0.746	0.890	0.676	0.872	0.915	0.892	0.847
Ours-64	0.9492	0.7762	0.7277	0.8934	0.6277	0.8392	0.8989	0.8856	0.8247
Ours-32	0.9453	0.7694	0.7354	0.8838	0.6320	0.8108	0.8905	0.8725	0.8175

Table 2. **Quantitative results from the real LLFF data set [6]**. We evaluate our 3-D model with feature length 32 and 64. *SSIM results are not provided.

Leaves

Fern

Orchids

Flower

T-Rex

Horns

Fortress

Room

Figure 1. **Visual results from the LLFF data set [6].**

4.2. Hardware Failure Case

Our work began with 24 GB of GPU memory and 32 GB of RAM. This works for the D-NeRF and LLFF data sets. However, for the DyNeRF data set the amount of RAM required for IST weight generation is significant (>256GB). We were unable to attain during our research. Instead, we found that 98 GB of RAM was sufficient and low cost, despite limiting us to 8x down sampling for IST weight generation. Note, this has been previously discussed in the official K-Planes repository[1].

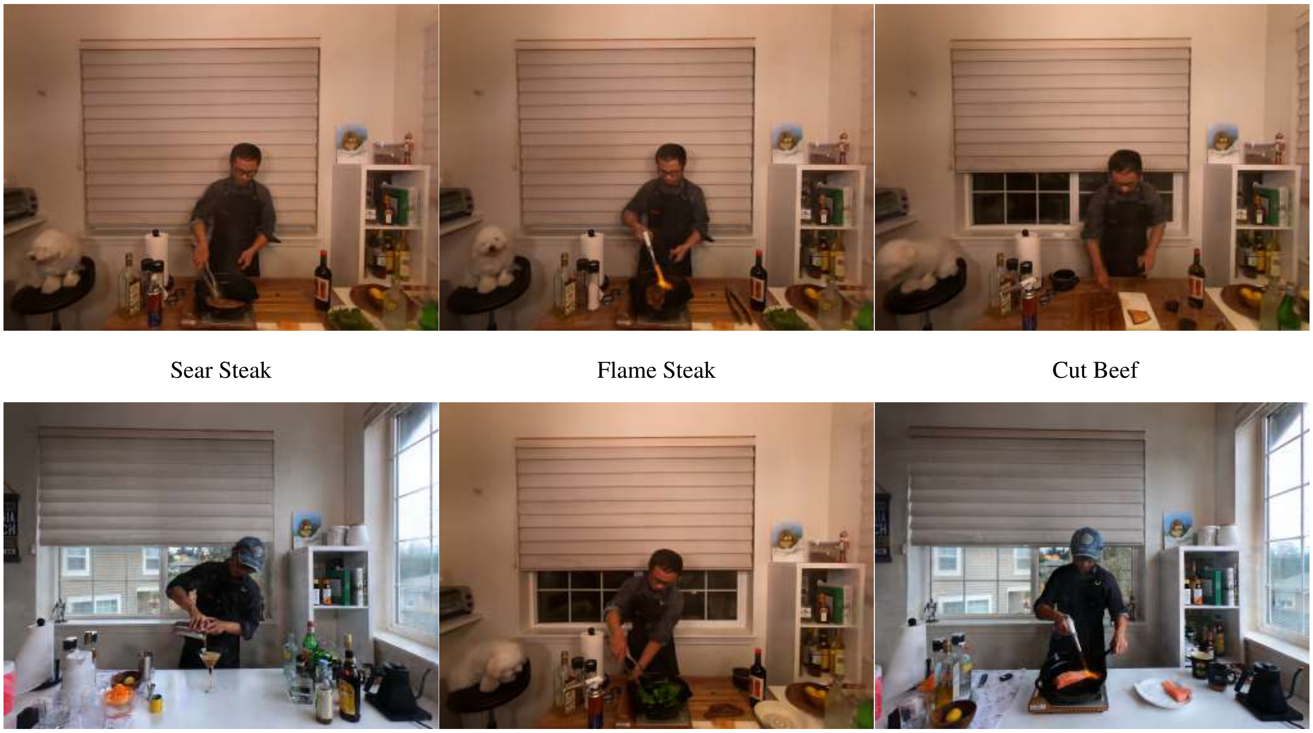
5. Failed Designs

The final design was not the first solution we conceived. In this section we detail several technical designs that failed.

[1] Accessible here: `https://github.com/sarafridov/K-Planes`

	PSNR ↑							
	Coffee	Spinach	Cut	Flame	Flame	Sear	Mean ↑	
	Martini (CM)		Beef	Salmon	Steak	Steak	All	No CM
DyNeRF [5]	-	-	-	29.58	-	-	-	-
LLFF [6]	-	-	-	23.24	-	-	-	-
K-Planes (explicit)	28.74	32.19	31.93	28.71 **	31.80	31.89	30.88	31.30
K-Planes (hybrid)	29.99	32.60	31.82	30.44 **	32.38	32.52	31.63	31.92
HexPlane	-	32.042	32.545	29.470	32.080	32.085	-	31.71
HexPlane+	-	31.860	32.712	29.263	31.924	32.085	-	31.57
Ours-HP*	28.01	29.78	31.39	27.31	30.71	30.23	29.57	29.88
Ours-ZMM*	27.30	30.00	30.89	27.6	30.4	29.88	29.35	29.75
Ours-ZAM*	27.17	30.16	31.45	28.25	30.49	30.37	29.65	30.14
	SSIM ↑							
DyNeRF [5]	-	-	-	0.961	-	-	-	-
LLFF [6]	-	-	-	0.848	-	-	-	-
K-Planes (explicit)	0.943	0.968	0.965	0.942 **	0.970	0.971	0.960	-
K-Planes (hybrid)	0.953	0.966	0.966	0.953 **	0.970	0.974	0.964	-
Ours-HP*	0.8896	0.9191	0.9338	0.8928	0.9364	0.9271	0.9165	-
Ours-ZMM*	0.8767	0.9215	0.9258	0.8934	0.9271	0.9321	0.9145	-
Ours-ZAM*	0.8854	0.9235	0.9345	0.9001	0.9331	0.9385	0.9191	-

Table 3. **Quantitative esults from the multi-view real DyNeRF dynamic scenes [5]**. *NoCM* indicates mean without the Coffee Martini scene. * Uses 8x down sampling for IST weights. **K-Planes trains the Flame Salmon scene on the first 10 seconds of a 40 second clip.



Sear Steak

Flame Steak

Cut Beef

Coffee Martini

Cook Spinach

Flame Salmon

Figure 2. **Visual results from the DyNeRF data set [5].** Using the ZAM feature fusion scheme.

5.1. K-Planes Time Smoothness Regularizer

As our implementation initially branched off the K-Planes code, our earliest design utilized the TS regularizer pro-

PSNR ↑

	Hell Warrior	Mutant	Hook	Balls	Lego	T-Rex	Stand-Up	Jumping Jacks	Mean ↑
D-NeRF	25.02	31.29	29.25	32.80	21.64	31.75	32.79	32.80	29.67
Tensor4D	-	-	-	-	26.71	-	36.32	34.43	-
TiNeuVox-S	27.00	31.09	29.30	39.05	24.35	29.95	32.89	32.33	30.75
TiNeuVox-B	28.17	33.61	31.45	40.73	25.02	32.70	35.43	34.23	32.67
V4D	27.03	36.27	31.04	42.67	25.62	34.53	37.20	35.36	33.72
K-Planes (explicit)	25.60	33.56	28.21	38.99	25.46	31.28	33.27	32.00	31.05
K-Planes (hybrid)	25.70	33.79	28.50	41.22	25.48	31.79	33.72	32.64	31.61
HexPlanes	24.24	33.79	28.71	39.69	25.22	30.67	34.36	31.65	31.04
Ours-HP	25.92	32.50	27.98	38.12	25.19	31.38	32.05	31.75	30.61
Ours-ZMM	25.63	32.51	27.97	37.71	25.26	31.41	32.15	31.33	30.50
Ours-ZAM	25.54	32.54	27.99	37.52	25.18	31.43	31.96	31.42	30.45
SSIM ↑									
D-NeRF	0.95	0.97	0.96	0.98	0.83	0.97	0.98	0.98	0.95
Tensor4D	-	-	-	-	0.953	-	0.983	0.982	-
TiNeuVox-S	0.95	0.96	0.95	0.99	0.88	0.96	0.98	0.97	0.96
TiNeuVox-B	0.97	0.98	0.97	0.99	0.92	0.98	0.99	0.99	0.97
V4D	0.97	0.98	0.97	0.99	0.92	0.98	0.99	0.98	0.97
K-Planes (explicit)	0.951	0.982	0.951	0.989	0.947	0.980	0.980	0.974	0.969
K-Planes (hybrid)	0.952	0.983	0.954	0.992	0.948	0.981	0.983	0.977	0.971
HexPlanes	0.94	0.98	0.96	0.99	0.94	0.98	0.98	0.98	0.97
Ours-HP	0.9545	0.9766	0.9470	0.9876	0.9414	0.9783	0.9744	0.9725	0.9655
Ours-ZMM	0.9523	0.9736	0.9470	0.9861	0.9417	0.9781	0.9755	0.9705	0.9655
Ours-ZAM	0.9521	0.9741	0.9479	0.9859	0.9399	0.9785	0.9752	0.9709	0.9656

Table 4. **Quantitative results from the monocular synthetic D-NeRF dynamic scenes [8].**

Size (MB) ↓

Room	Fern	Leaves	Fortress	Orchids	Flower	T-Rex	Horns	Mean
62	104	132	32	134	107	78	96	93
Coffee Martini	Spinach	Cut Beef	Flame Salmon	Flame Steak	Sear Steak			
65	58	58	57	49	62			58
Hell Warrior	Mutant	Hook	Balls	Lego	T-Rex	Stand-Up	Jumping Jacks	
13	4	15	5	14	17	12	19	12

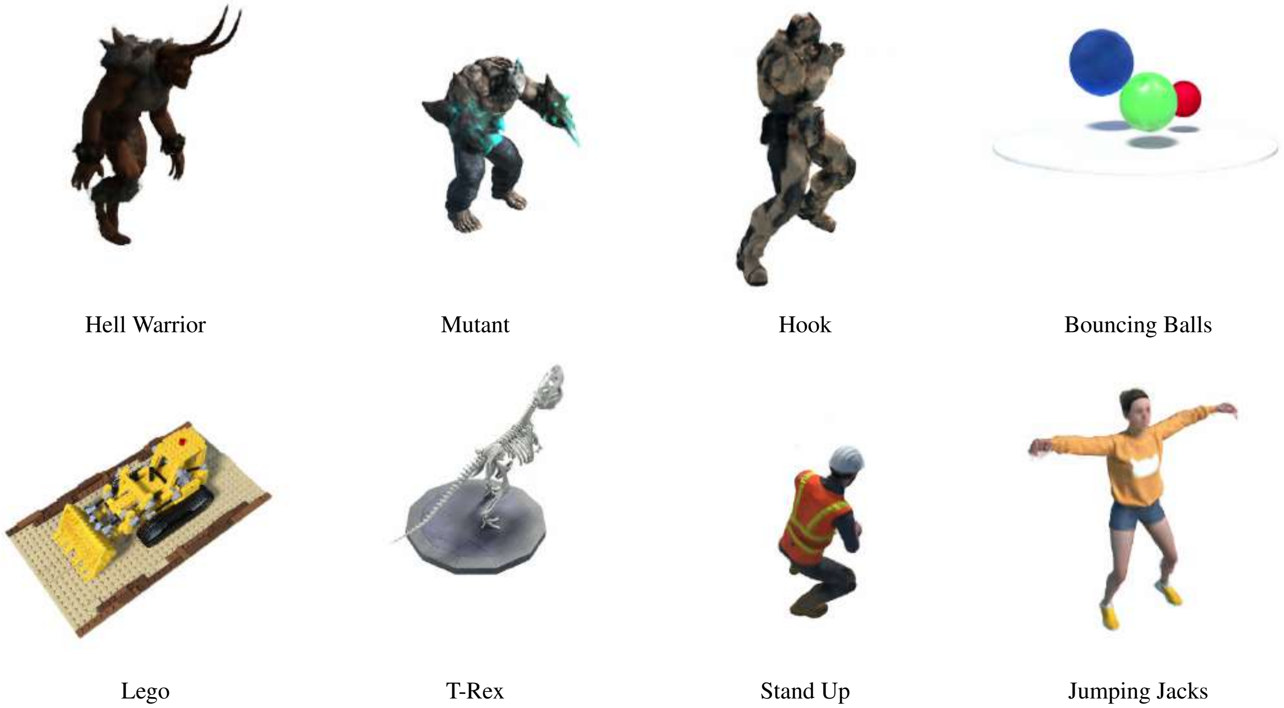
Table 5. **Per-scene final model sizes**. Results are collected using HP fusion for static scenes as ZMM fusion for dynamic scenes. Note that the final model size for the ZMM-based model is around 0.2MB larger than the HP-based model for high resolution scenes. For the ZAM-based model this difference is around 0.4MB.

posed in [4] instead of the proposed SST regularizer. In Table 11 we compare the results from using TS regularization and SST regularization, showing that for WavePlanes the SST regularizer is a better fit.

$$\mathcal{L}_{TV} = \frac{1}{|C|n^2} \sum_{c,i,t} ||\mathbf{P}_c^{i,t-1} - 2\mathbf{P}_c^{i,t} + \mathbf{P}_c^{i,t+1}||_2^2 \qquad (2)$$

5.2. Directly Smoothing Temporal Coefficients with Orientation

Each set of mother wavelet coefficients contains components for horizontal, diagonal and vertical filters which we define as $\Omega_{c,f} \in \mathbf{\Omega}_c$ where $f \in [horizontal, vertical, diagonal]$. This offers the opportunity to refactor the SST function (a 1-d Laplacian approximation of the planes second derivative) to regularize coefficient filters directly. This also allows us to prioritize smoothness for each filter direction, where the horizontal filter exists along the time-axis, the vertical filter along the spatial axis and the diagonal filter along $t = x = z = y$. Hence we use Equation 3 for each filter where $c = c_t$. To regularize the father wavelet coefficients, which do not contain directional filters, we apply the 1-D Laplacian approximation across the horizontal and vertical axis and add the result to $\mathcal{L}_{SST-horizontal}$ and $\mathcal{L}_{SST-vertical}$, respectively. For $f = diagonal$, we average the second order approximations along both axis' of the father wavelet planes and



Hell Warrior

Mutant

Hook

Bouncing Balls

Lego

T-Rex

Stand Up

Jumping Jacks

Figure 3. **Qualitative results from the D-NeRF data set [8].** Using the ZAM feature fusion scheme.

Scaling **k**	PSNR ↑ Whole	 Front	 Back	SSIM ↑
$N = 2$ levels, 72 mins				
$[1, 1, 1]$	31.30	20.63	76.01	0.977
$[1, 0.8, 0.4]$	31.31	20.64	75.78	0.978
$[1, 0.4, 0.2]$	31.34	20.67	78.05	0.978
$[1, 0.8, 0.2]$	31.33	20.66	76.75	0.978
$N = 3$ levels, 84 mins				
$[1, 1, 1, 1]$	30.52	19.53	75.39	0.975
$[1, 0.8, 0.6, 0.4]$	30.98	20.27	75.41	0.977
$[1, 0.5, 0.3, 0.1]$	30.64	19.97	78.00	0.975
$[1, 0.8, 0.4, 0.2]$	30.37	19.70	76.21	0.975
$N = 4$ levels, 95 mins				
$[1, 1, 1, 1]$	30.03	19.37	62.34	0.973
$[1, 0.5, 0.4, 0.2, 0.1]$	30.65	19.98	75.08	0.975
$[1, 0.8, 0.6, 0.4, 0.2]$	30.22	19.56	69.29	0.974

Table 6. **Level-dependant scaling coefficients, k**, used for the comparing different levels of wavelet decomposition. Accomplished on the T-Rex D-NeRF scene. Training time is provided in minutes.

add the result to $\mathcal{L}_{SST-diagonal}$. This ensures that all coefficients are regularized for a given orientation. The result of using these separately is compared with the proposed SST regularization in Table 12.

$$\mathcal{L}_{SST-f} = \frac{1}{|C|n^2} \sum_{c,i,t} ||\mathbf{\Omega}_{c,f}^{i,t-1} - 2\mathbf{\Omega}_{c,f}^{i,t} + \mathbf{\Omega}_{c,f}^{i,t+1}||_2^2 \quad (3)$$

Weight	PSNR ↑ Whole	 Front	 Back	SSIM ↑
0.01	29.84	19.17	76.31	0.9696
0.001	30.70	20.03	78.89	0.9752
0.0001	30.89	20.22	77.52	0.9763
0.00001	30.97	20.31	74.36	0.9763

Table 7. Comparing different weights for ST regularisation on the T-Rex D-NeRF scene.

Weight	PSNR ↑ Whole	 Front	 Back	SSIM ↑
0.1	31.06	20.39	78.58	0.9770
0.01	30.53	19.86	77.66	0.9756
0.001	29.45	18.83	77.18	0.9720
0.0001	28.88	18.21	76.50	0.9688

Table 8. Comparing different weights for SST regularisation on the T-Rex D-NeRF scene [8].

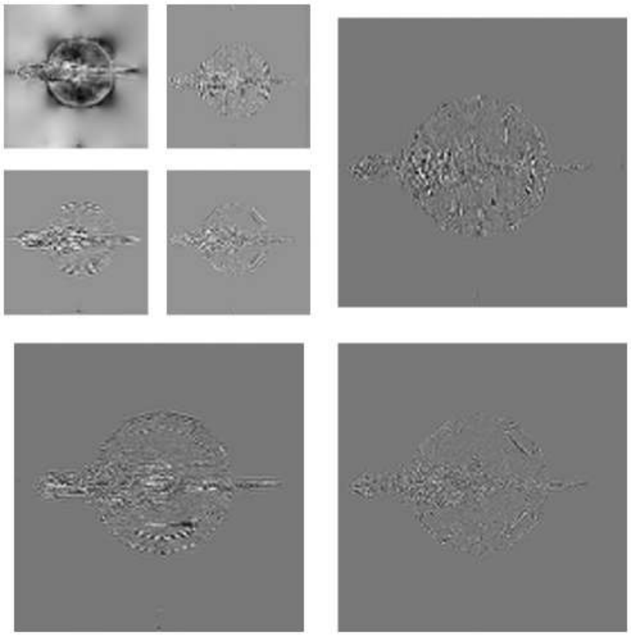
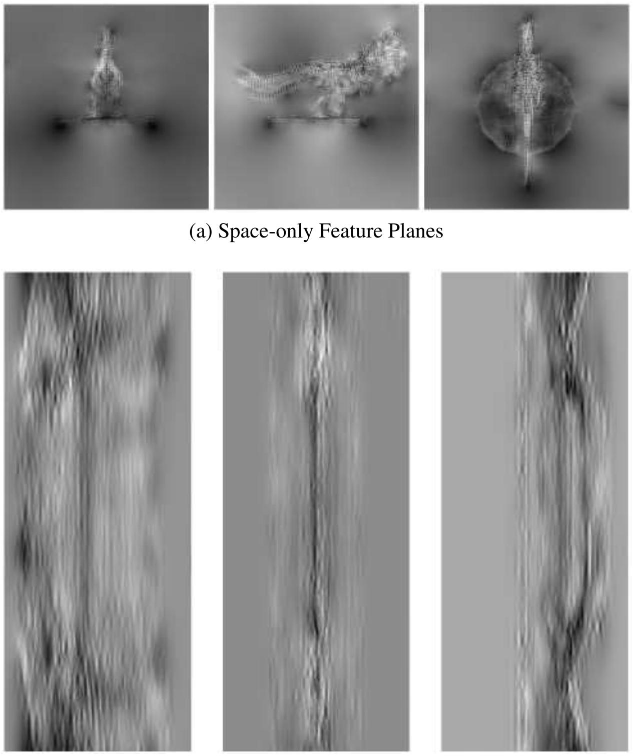


Figure 4. **Visualising wavelet coefficient planes** for the T-Rex D-NeRF scene representation. We use the 2-level wavelet decomposition that produces the $\mathbf{P}^s_{xy}$ feature plane to illustrate the coefficients at each level and for separate filter orientations. **Small Images**: The father wavelet plane (top left) has a resolution of 64×64 and is surrounded by the vertical (top right), horizontal (bottom left) and diagonal (bottom right) mother wavelet coefficients. **Large Images**: The mother wavelet coefficients have a resolution of 128×128 where the vertical (top right), horizontal (bottom left) and diagonal (bottom right) planes are shown.

Weight	PSNR ↑			SSIM ↑
	Whole	Front	Back	
0.01	21.55	11.77	51.46	0.9109
0.001	24.47	14.68	59.21	0.9415
0.0001	25.43	15.63	67.79	0.9504
0.00001	25.92	16.13	70.70	0.9545
0.000001	25.60	15.81	67.92	0.9520

Table 9. Comparing different weights for TV regularisation on the Hell Warrior D-NeRF scene.



(a) Space-only Feature Planes

(b) Space-time Feature Planes

Figure 5. **Visualising the space-only and space-time feature planes**. Black to white pixels indicate negative to positive feature values, respectively. (a) Space-only feature planes are visualized. (b) Space-time features are visualized where the horizontal axis represents time.

Resolution		PSNR ↑	SSIM ↑	Time ↓
P^0_c	P^1_c			
T-Rex scene, D-NeRF [8]				
128	64	30.88	0.9749	62
256	128	31.34	0.9782	72
512	256	30.75	0.9761	127
Lego, D-NeRF [8]				
128	64	25.25	0.9380	60
256	128	25.19	0.9876	68
512	256	24.76	0.9377	113
Bouncing Balls, D-NeRF [8]				
128	64	37.71	0.9876	70
256	128	36.66	0.9840	74
512	256	34.74	0.9802	110

Table 10. **Varying the wavelet plane's resolution** for various D-NeRF scenes scenes. The resolution of the resulting features planes is shown, where the wavelet coefficients will have a maximum resolution half that of P^0_c.

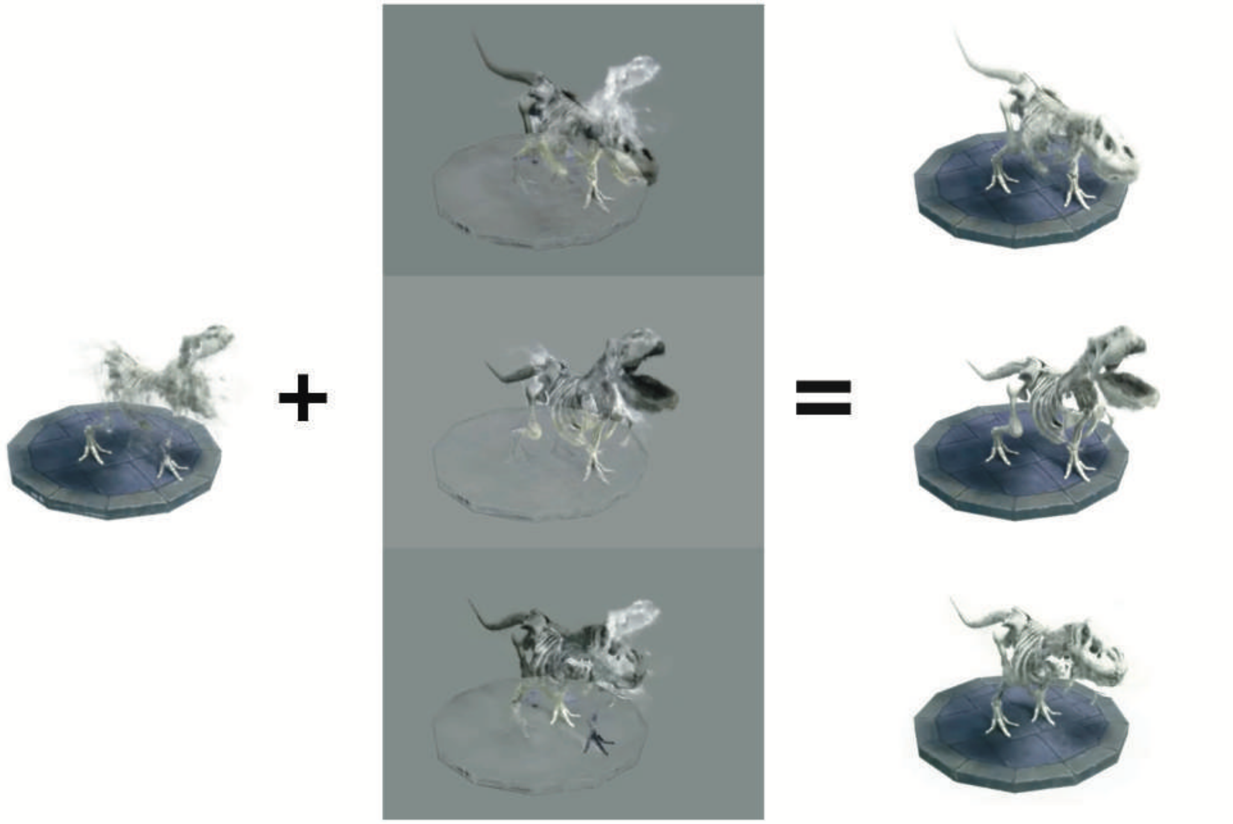


Figure 6. **Visualising space and space-time separation** on the T-Rex D-NeRF scene. **Left**: Space-only features are rendered. **Right**: All features are rendered. **Center**: All rendered features are subtracted from the space-only render, representing the effect of space-time features on the final render.

	PSNR ↑			SSIM ↑
	Front	Back	Whole	
ZMM-SST	20.74	76.41	31.41	0.9781
ZMM-TS	18.70	75.29	29.37	0.9721

Table 11. **Comparing SST and TS regularizers** on the T-Rex D-NeRF scene [8].

f	PSNR ↑	
	Whole	Front
*	31.34	20.67
horizontal	29.23	18.55
diagonal	29.14	18.41
vertical	29.08	18.48

Table 12. **Comparing directionally dependant smoothness regularizers** on the wavelet coefficients for the T-Rex D-NeRF data set [8]. * indicates the final Wave Planes model which uses the proposed SST regularization.

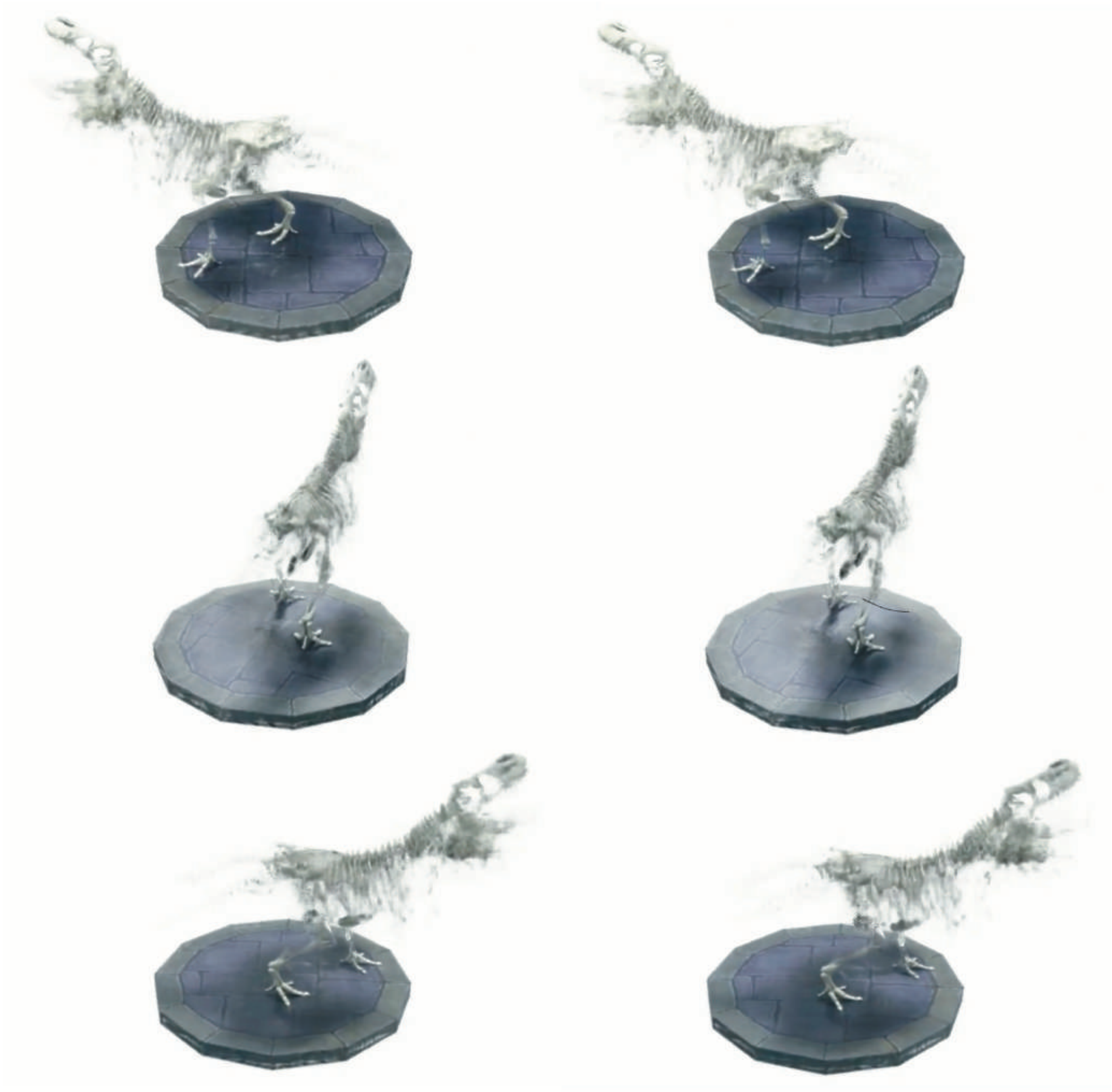


Figure 7. **Visualising the static decomposition from HP** (left) **and ZMM** (right) **fused WavePlane models** on the T-Rex D-NeRF scene. Less noise is present for the ZMM approach, e.g. around the tail and the lower jaw bones.

References

[1] Allison H Baker, Alexander Pinard, and Dorit M Hammerling. Dssim: a structural similarity index for floating-point data. *arXiv preprint arXiv:2202.02616*, 2022.

[2] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision*, pages 333–350. Springer, 2022.

[3] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5501–5510, 2022.

[4] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023.

[5] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5521–5531, 2022.

[6] 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 (TOG)*, 2019.

[7] 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. *Communications of the ACM*, 65(1):99–106, 2021.

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

[9] Daniel Rho, Byeonghyeon Lee, Seungtae Nam, Joo Chan Lee, Jong Hwan Ko, and Eunbyung Park. Masked wavelet representation for compact neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20680–20690, 2023.

[10] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, pages 1398–1402. Ieee, 2003.

[11] 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.