A. Further Implementation Details for Different Methods

EWC. EWC [12] regularizes the current model $\boldsymbol{\theta}_t$ to be close to $\boldsymbol{\theta}_{t-1}$. The training loss of EWC at time step t is

$$\mathcal{L}_{\text{EWC}}(\boldsymbol{\theta}_t) = \mathcal{L}_{\text{NeRF}}(\boldsymbol{\theta}_t) + \sum_i \frac{\lambda}{2} F_i(\boldsymbol{\theta}_{t,i} - \boldsymbol{\theta}_{t-1,i})^2, \quad (4)$$

where $\mathcal{L}_{\text{NeRF}}(\cdot)$ is the original NeRF training loss, F_i is the i-th element of the diagnal of the Fisher information matrix, $\boldsymbol{\theta}_{t,i}$ is the i-th element of $\boldsymbol{\theta}_t$, and λ is the hyper-parameter that controls the regularization strength. As in the original paper, the regularization term from EWC has a much smaller magnitude than $\mathcal{L}_{\text{NeRF}}$. To make regularization effective, we tune λ by grid search from $1\text{e}3$ to $1\text{e}9$ on the Synth-NeRF dataset, and picking $1\text{e}5$ that provides the best performance (making the regularization term roughly 10% of $\mathcal{L}_{\text{NeRF}}$).

ER. Experience replay (ER) [5] uses the loss on both new and historical data to update the model $\boldsymbol{\theta}_t$. Due to the ineffectiveness of biased sampling for continual NeRF, we uses random sampling for ER to produce the best performance and uniformly weight the losses of all ray samples.

MEIL-NeRF. MEIL-NeRF [6] forms each mini-batch of training data by sampling $\frac{2}{3}$ of the rays from new data, and the rest from old ones. The training loss of MEIL-NeRF is

$$\tfrac{1}{|\mathbf{X}_c|} \textstyle\sum_{\mathbf{X}_c} \mathcal{L}_{\text{NeRF}}(\mathbf{X}_c, \boldsymbol{\theta}_t) + \tfrac{\lambda_{\text{MEIL}}}{|\mathbf{X}_o|} \textstyle\sum_{\mathbf{X}_o} \rho(\hat{C}(\mathbf{X}_o|\boldsymbol{\theta}_{t-1}) - \hat{C}(\mathbf{X}_o|\boldsymbol{\theta}_t)), \quad (5)$$

where $\mathbf{X}_c$ and $\mathbf{X}_o$ are respectively the rays from new and old data, $\mathcal{L}_{\textbf{NeRF}}(\cdot)$ is the original loss for NeRF training, $\rho(\cdot)$ is the Charbonnier penalty function, $\hat{C}(\mathbf{X}|\boldsymbol{\theta})$ is the color of ray $\mathbf{X}$ generated by $\boldsymbol{\theta}$, and λ_{MEIL} is the hyper-parameter that controls the regularization strength from old data. Following S2 of the original paper, λ_{MEIL} is scheduled as

$$\lambda_{\text{MEIL}} = \frac{\cos(\pi(1+r)) + 1}{2}, \quad (6)$$

where r is the training progress rate (from 0 to 1) in each time step. Note that we correct the typo of the original paper and add a "$+1$" after $\cos(\cdot)$ to ensure that λ grows gradually from 0 to 1 (consistent with Figure S5 of the original paper).

CLNeRF Depending on the resource limit, the generative replay of CLNeRF can be implemented in both online and offline fashion. For applications where new data are generated only once a while (e.g., city scans are uploaded once a couple of days), the model is mostly in the deploy mode. Hence during the infrequent case of model update, we can assign a temporal storage to store all images generated by $\boldsymbol{\theta}_{t-1}$, and then use them to update $\boldsymbol{\theta}_t$. The benefit of this implementation is that we only need to load 1 model into the GPU memory, and the temporal storage can be released once we finish updating $\boldsymbol{\theta}_t$ (which takes only 5-20 mins for CLNeRF). For applications where no temporal storage can be used, we load $\boldsymbol{\theta}_{t-1}$ (evaluation mode to same memory) and $\boldsymbol{\theta}_t$ into the GPU at the same time, and generate the replay supervision signal on-the-fly, which requires an extra forward pass per training iteration. In our test, such implementation can still fit into a single RTX6000 GPU, and increased the training time by roughly 60%, which is still fast for NGP. We use the first implementation in our experiments due to its simplicity.

B. Further Qualitative Results on WAT

Here, we show further qualitative results on individual scenes of WAT. To better demonstrate the advantage of CLNeRF in terms of architecture design and continual learning strategies, we show the results of other continual NeRF methods with (Fig. 8) and without (Fig. 7) using the proposed trainable embeddings (please refer to the video demo in our github repo for close-view comparisons). Methods without trainable embeddings cannot recover the geometry and appearance of past time steps, resulting in severe artifacts. Even with trainable embeddings, severe artifacts (NT, EWC) and detail lost (ER, MEIL-NeRF) still exist in other baselines.

C. Quantitative Results on Individual Scenes

This section shows the quantitative results (Tab. 6 to 10) on individual scenes of each used dataset. CLNeRF performs better than other continual NeRF approaches on *all* individual scenes, even without storing any historical image. The performance gap between CLNeRF and UB is also small for all scenes. Due to the training noise and the close practical performance, the best model of each scene changes between CLNeRF and CLNeRF-noER and UB.

D. Baseline Results with NGP

In the main experiments (Tab. 1), we choose the best architecture for different baselines and observe that methods like ER and NT perform better with vanilla NeRF. Here, we further report the baseline results using NGP architecture. Due to the NaN loss issue of NGP, we cannot perform EWC on it. Hence, we only report the results for NT and ER. Due to the resource limit, we only report the results on Synth-NeRF, NeRF++ and WAT. As discussed in Sec. 5.2.3, NGP without generative replay overfits to the training views and fails to generalize to novel views and past time steps.

E. More WAT Scenes

In this section, we add 4 more scenes to WAT, making the total number of scenes from 6 to 10. We use the same data creation process as described in Sec. 4, and call this enlarged dataset WAT+ (available in our code repository). As shown in Tab. 12, the results on these new scenes are consistent with Tab. 1.

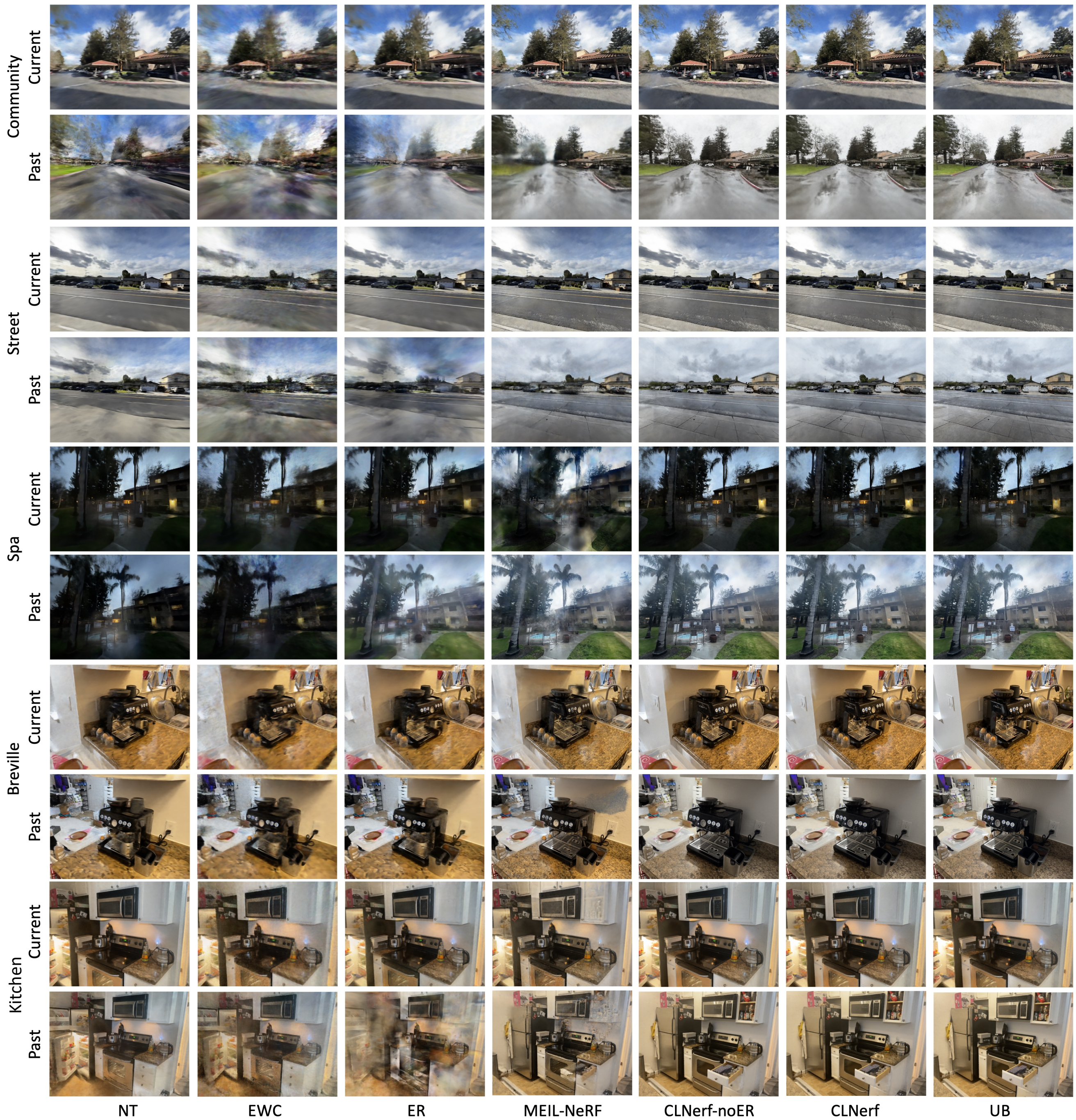

Figure 7. **Further qualitative results without using trainable embeddings for other continual NeRF baselines (zoom-in recommended).** Methods without trainable embeddings cannot properly recover the appearance and geometry of the scene at past time steps.

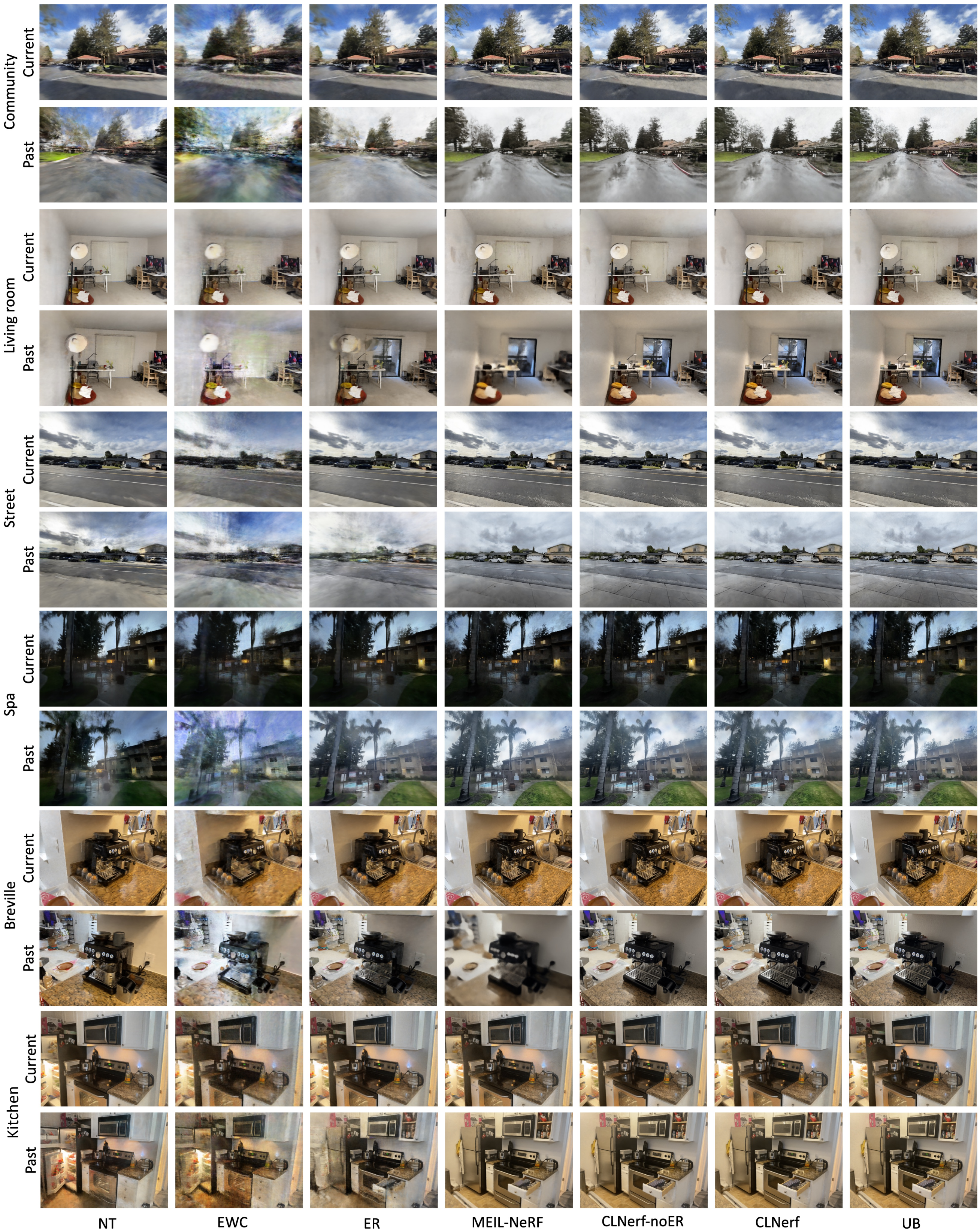

Figure 8. **Further qualitative results with trainable embeddings used in all methods (zoom-in recommended).** Each two rows show for different methods the rendered views on test data from current and previous steps. Similar performance difference can be observed as in the main experiments.

Method \ Scene	Breville	Kitchen	Living room	Community	Spa	Street
NT (NeRF) noEmbed	19.53/0.676	18.30/0.771	17.27/0.788	13.58/0.533	12.97/0.489	15.36/0.490
NT (NeRF)	19.95/0.695	18.69/0.771	17.65/0.790	13.97/0.538	14.71/0.612	15.24/0.486
EWC (NeRF) noEmbed	19.07/0.653	17.70/0.753	17.36/0.782	13.40/0.530	12.57/0.501	14.76/0.473
EWC (NeRF)	19.15/0.657	17.89/0.745	17.47/0.780	14.12/0.543	17.04/0.701	15.14/0.475
ER (NeRF) noEmbed	19.07/0.673	20.12/0.793	17.84/0.793	16.68/0.589	16.77/0.634	16.87/0.512
ER (NeRF)	23.46/0.717	25.03/0.836	21.41/0.813	17.65/0.600	21.92/0.752	18.71/0.538
MEIL-NeRF (NGP) noEmbed	22.12/0.781	22.08/0.822	18.11/0.785	21.91/0.612	20.67/0.741	20.64/0.581
MEIL-NeRF (NGP)	23.55/0.710	27.27/0.861	23.56/0.821	21.92/0.609	25.60/0.771	22.39/0.608
CLNeRF-noER (NGP)	27.61/0.808	**28.59/0.880**	24.56/0.829	22.80/0.628	**26.56/0.812**	**22.53/0.615**
CLNeRF (NGP)	**28.02/0.826**	28.40/0.877	**24.58/0.829**	**22.88/0.629**	26.28/0.811	22.53/0.612
UB (NGP)	28.62/0.838	28.53/0.878	24.51/0.826	23.65/0.634	26.92/0.812	22.87/0.615

Table 6. **Results on individual scenes of WAT**.

Method \ Scene	Lego	Chair	Drums	Ficus	Hotdog	Materials	Mic	Ship
NT (NeRF)	30.70/0.956	31.02/0.965	21.96/0.897	27.47/0.957	32.64/0.969	26.63/0.932	29.68/0.972	28.17/0.855
EWC (NeRF)	30.82/0.952	30.57/0.950	22.58/0.895	26.41/0.946	32.55/0.967	26.39/0.918	30.73/0.970	26.55/0.823
ER (NeRF)	33.21/0.969	32.76/0.973	23.25/0.915	29.56/0.968	34.22/0.973	27.50/0.939	32.79/0.984	29.06/0.866
MEIL-NeRF (NGP)	32.91/0.971	32.75/0.979	24.35/0.928	30.96/0.977	34.87/0.977	28.16/0.941	32.93/0.986	28.52/0.866
CLNeRF-noER (NGP)	34.34/0.975	34.29/**0.982**	**25.51/0.931**	32.91/0.979	36.20/0.979	28.92/0.941	34.38/0.987	29.15/0.878
CLNeRF (NGP)	**34.80/0.976**	**34.39**/0.980	25.46/0.931	**33.10/0.980**	**36.42/0.979**	**29.12/0.943**	**34.68/0.987**	**29.30/0.880**
UB (NGP)	35.70/0.978	35.41/0.982	25.74/0.932	33.96/0.982	37.24/0.980	29.43/0.944	35.83/0.989	30.19/0.888

Table 7. **Results on individual scenes of SynthNeRF**.

Method \ Scene	Ignatius	Truck	Barn	Caterpillar	Family
NT (NeRF)	24.51/0.924	16.76/0.739	15.98/0.614	13.64/0.701	22.03/0.863
EWC (NeRF)	24.83/0.925	16.28/0.730	12.46/0.619	14.34/0.898	22.68/0.880
ER (NeRF)	26.79/0.948	24.26/0.883	26.02/0.836	24.75/0.898	30.71/0.944
MEIL-NeRF (NGP)	29.56/0.954	25.85/0.903	22.99/0.825	26.33/0.916	31.24/0.949
CLNeRF-noER (NGP)	30.22/0.956	27.55/**0.920**	**27.55/0.851**	28.12/0.930	33.34/0.961
CLNeRF (NGP)	**30.41/0.957**	**27.59**/0.918	27.47/0.848	**28.29/0.933**	**33.68/0.961**
UB (NGP)	30.94/0.960	28.11/0.926	28.45/0.866	29.08/0.940	34.91/0.964

Table 8. **Results on individual scenes of NSVF**.

Method \ Scene	M60	Playground	Train	Truck
NT (NeRF)	15.87/0.569	15.70/0.444	12.73/0.365	14.91/0.468
EWC (NeRF)	13.89/0.465	16.28/0.462	13.38/0.366	16.56/0.478
ER (NeRF)	16.10/0.580	19.67/0.569	16.18/0.476	17.46/0.544
MEIL-NeRF (NGP)	18.11/0.621	21.53/0.592	17.16/0.533	20.76/0.635
CLNeRF-noER (NGP)	18.88/0.631	22.18/0.643	17.20/0.561	**22.99**/0.694
CLNeRF (NGP)	**19.04/0.634**	**22.37/0.643**	**17.31/0.563**	22.61/**0.695**
UB (NGP)	18.69/0.623	22.31/0.672	17.36/0.586	22.99/0.712

Table 9. **Results on individual scenes of NeRF++**.

Method \ Scene	Brandenburg Gate	Sacre Coeur	Trevi Fountain	Taj Mahal
NT (NGP)	21.11/0.793	15.78/0.642	19.43/0.613	19.82/0.722
ER (NGP)	24.20/0.803	16.91/0.682	19.57/0.616	19.85/0.723
MEIL-NeRF (NGP)	24.22/0.802	20.53/0.744	21.42/0.667	23.21/0.770
CLNerf-noER (NGP)	25.24/**0.803**	20.62/0.753	**21.53/0.668**	23.32/**0.781**
CLNerf (NGP)	**25.43**/0.802	**21.32/0.765**	21.44/0.667	**23.34**/0.780
UB (NeRFW)	24.23/0.881	21.59/0.833	21.99/0.853	23.32/0.726
UB (NGP)	25.58/0.813	21.22/0.785	21.64/0.676	23.76/0.780

Table 10. **Results on individual scenes of Phototourism**.

Dataset	Synth-NeRF	NeRF++	WAT
NT (NGP)	21.66/0.858	11.93/0.380	11.51/0.356
ER (NGP)	27.35/0.919	15.13/0.436	19.03/0.657

Table 11. **Baseline results on NGP architecture**. Comparing with Tab. 1, we can see that for baselines like NT and ER, NGP performs much worse than vanilla NeRF. See Sec. 5.2.3 for analysis on architectures.

Method \ Scene	Car	Grill	Mac	Ninja
NT (NeRF)	19.14/0.516	19.96/0.612	18.68/0.831	19.92/0.814
EWC (NeRF)	18.46/0.500	19.67/0.602	18.48/0.816	19.74/0.805
ER (NeRF)	19.10/0.510	21.03/0.618	23.07/0.871	21.49/0.827
MEIL-NeRF (NGP)	21.80/0.528	24.01/0.648	12.71/0.687	13.22/0.668
CLNeRF-noER (NGP)	22.63/0.539	24.81/0.652	**29.34/0.907**	26.42/0.869
CLNeRF (NGP)	**22.73/0.541**	**24.84/0.653**	29.33/0.906	**27.19/0.878**
UB (NGP)	22.62/0.538	25.12/0.661	30.16/0.905	26.71/0.875

Table 12. **Results on the 4 scenes added to WAT+**.