PAPR in Motion: Seamless Point-level 3D Scene Interpolation

Supplementary Material

A. Video Results

Please visit our project website for the video results, where we include animations of our method and comparisons to the baseline.

B. Sensitivity Analysis

We perform a sensitivity analysis to evaluate our model's performance in relation to its hyperparameters. Specifically, we focused on examining the impact of varying the number of nearest neighbours, denoted as k, used in the regularization techniques. Figure 5 illustrates the findings from our analysis of k's influence. As shown, higher values of k generally lead to increased rigidity in the moving parts of objects. This heightened rigidity contributes to more accurate interpolations, as observed in the improved preservation of surface smoothness and structural integrity, particularly noticeable in the butterfly's wing during its motion.

However, it is important to note that as k increases, it imposes more constraints on object movements due to the larger neighbourhood size considered. This can be a limiting factor, particularly in scenes with intricate geometries, where an excessively high k value might overly restrict movement and hinder the parts from interpolating correctly. In practice, we choose a value of k that strikes a balance between maintaining structural rigidity and allowing sufficient flexibility for part movement.

We also show the effect of varying the interval m for the local displacement averaging step (LDAS). As shown in Figure 6, too small a value of m may slow down the geometry adaptation process. In practice, we choose a value of $m = 100$ that best balances the adaptation speed and the quality of the intermediate renderings.

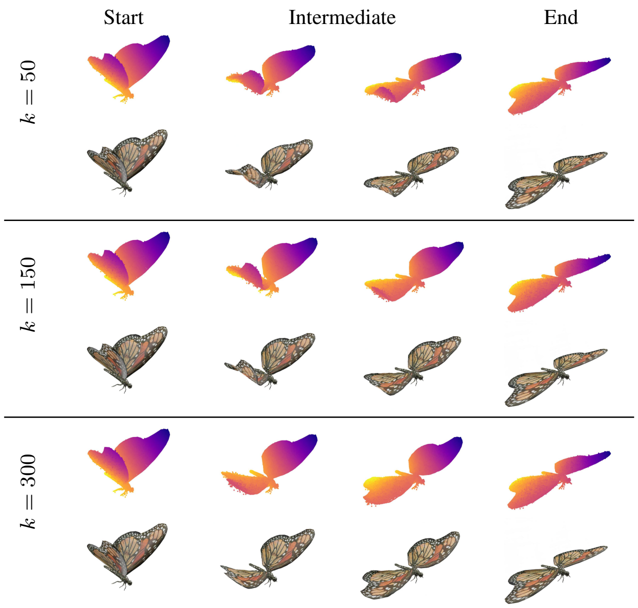

Figure 5. Sensitivity analysis on the effect of k, the number of nearest neighbour used in regularization calculations. The results show that larger values of k tend to increase the rigidity of moving parts while smaller values of k result in more flexible part movements. In this example, $k = 300$ exhibits the best surface continuity and smoothness throughout the interpolation process.

C. Implementation Details

Dataset Details We use multi-view RGB images from both the start and end states of each scene as our input data. The training set for each scene consists of 100 randomly selected views from the upper hemisphere for each state, while the evaluation set comprises 200 unseen test views. In synthetic scenes, all images are rendered at a resolution of 800×800 pixels. The real-world tablet stand scene is rendered at a resolution of 1008×756 pixels, and the lamp scene at 960×540 pixels.

Training Details The duration of the intermediate scene interpolation process in our method takes 16 epochs. To enhance efficiency, we found that finetuning the model on target images downsampled by a factor of two is sufficient. This results in a training time of approximately an hour on a single NVIDIA A100 GPU. The end state appearance finetuning takes 16 epochs. We choose the number of nearest neighbours k for each scene based on how rigid the object should be – the more rigid it is, the higher the value of k. The specific values of k for each scene are detailed in Table 2.

For the baseline method, Dynamic Gaussian [24], their original approach involves finetuning for 75 epochs at each subsequent time step after the initial one. To better accommodate larger scene changes in our context, we extend this significantly to 300 epochs.

D. Additional Results

Figure 7 and 8 shows additional qualitative comparisons between our method, PAPR in Motion, and Dynamic Gaussian [24]. The results show that Dynamic Gaussian [24] struggles with maintaining object geometry integrity during the interpolation process. For instance, in the Lego Bulldozer scene, points on the arm notably drift, and points

Synthetic Scenes						Real-world Scenes	
Butterfly	Crab	Dolphin	Giraffe	Lego Bulldozer	Lego Man	Stand	Lamp
300	150	200	70	100	200	200	200

Table 2. Different values of nearest neighbour k for each scene.

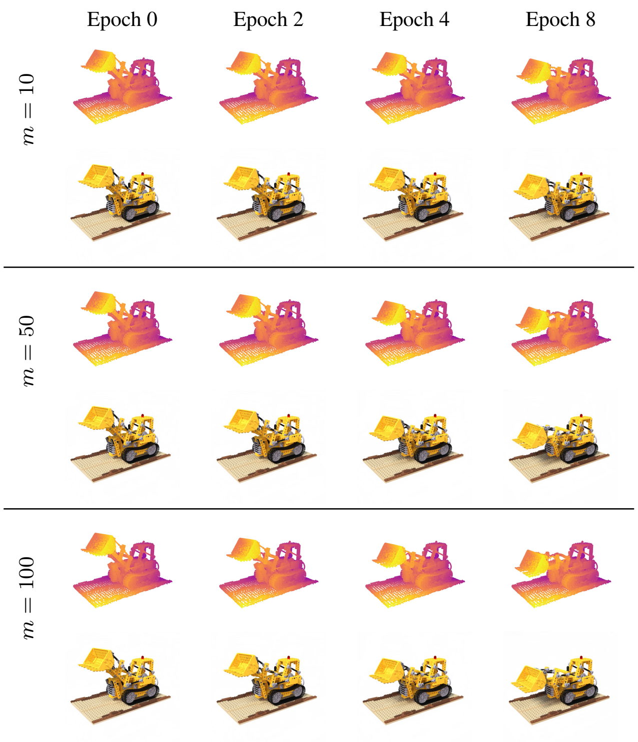

Figure 6. Sensitivity analysis on the effect of m, the interval size in terms of number of iterations to apply local displacement averaging step (LDAS).

from the back of the cabin erroneously travel to the front. Similarly, in the giraffe scene, a portion of the points on the giraffe's neck and legs do not move cohesively, leading to disjointed transitions. In scenes with drastic changes like the butterfly and crab scenes, the baseline fails to preserve the original appearance. In contrast, PAPR in Motion successfully handles these challenging scenarios, producing smooth and natural interpolations between states.

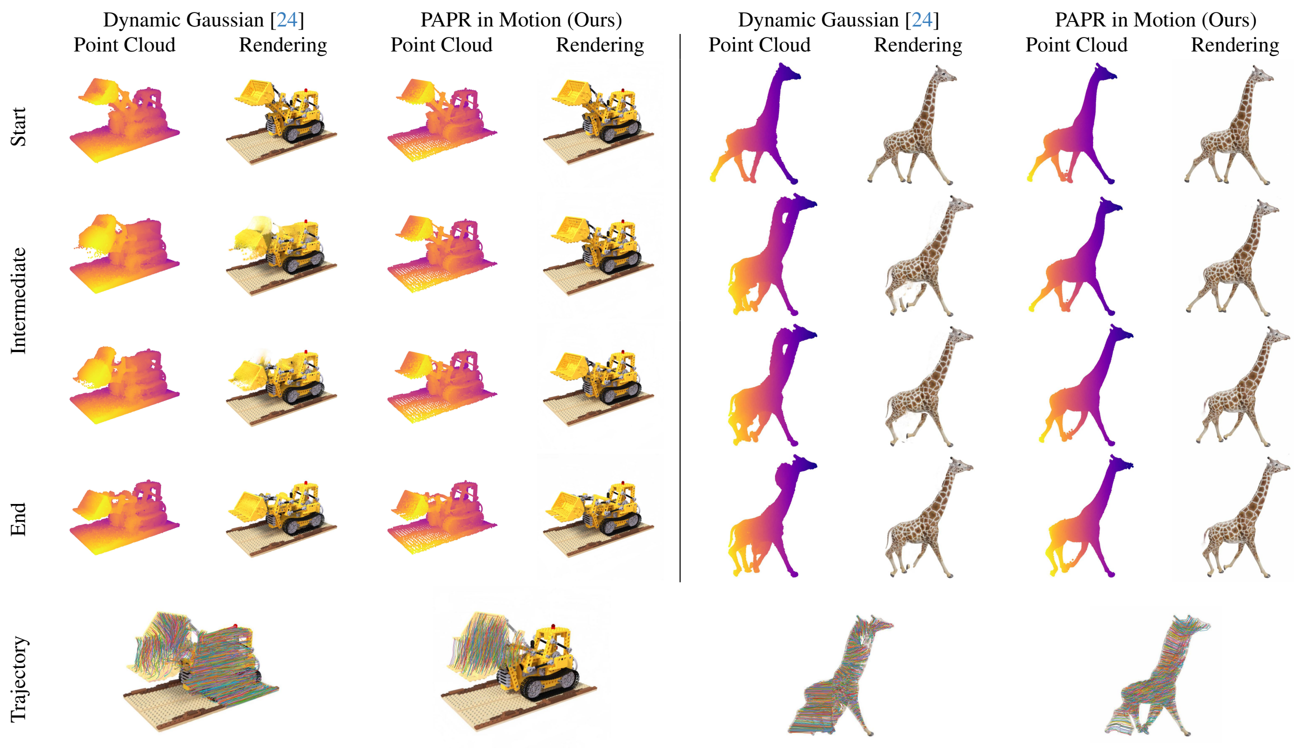

Figure 7. Qualitative comparison of 3D scene interpolation from start to end state using synthetic scenes. Both methods start by training a static model for the start state and subsequently finetune it towards the end state, all without any intermediate supervision. As shown, our PAPR in Motion method generates more plausible interpolation between states.

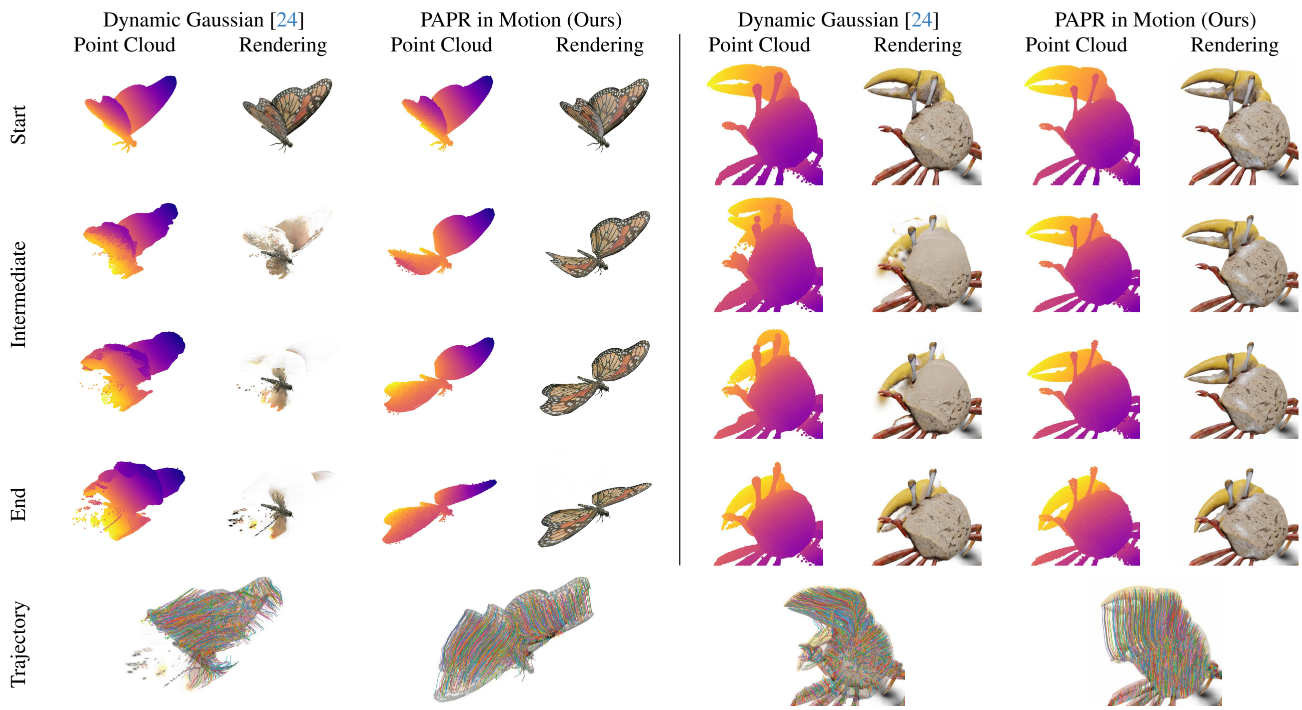

Figure 8. Qualitative comparison of 3D scene interpolation from start to end state using synthetic scenes. Both methods start by training a static model for the start state and subsequently finetune it towards the end state, all without any intermediate supervision. Dynamic Gaussian [24] fails to handle scene changes with large displacements, as shown by the butterfly example where the wings disappear, and in the crab scene, where the claw's geometry distorts during transition and fails to retain appearance details on the crab's shell. In contrast, PAPR in Motion produces smooth interpolations between states.