RAP: Fast Feedforward Rendering-Free Attribute-Guided Primitive Importance Score Prediction for Efficient 3D Gaussian Splatting Processing

Supplementary Material

8. Runtime breakdown and scalability of KNN-based normalization.

We provide a runtime breakdown and scalability analysis of the KNN-based normalization. The runtime is dominated by KNN search (83.45%), followed by feature computation (16.03%), while MLP inference accounts for only 0.52%. Fig. 6 further shows that the KNN cost scales smoothly from 100K to 5M Gaussians with $k = 128$. For a typical training-scale scene with about 1M Gaussians, the total preprocessing time is approximately 2 seconds, and for the largest scene (*bicycle*) with around 5M Gaussians, it is about 9 seconds.

For KNN search, we employ the GPU-accelerated IVF-Flat approximate KNN method from the cuVS library. In practice, training converges stably with approximate neighbors, indicating that exact KNN is unnecessary and that locally approximate neighborhoods are sufficient for accurate importance estimation.

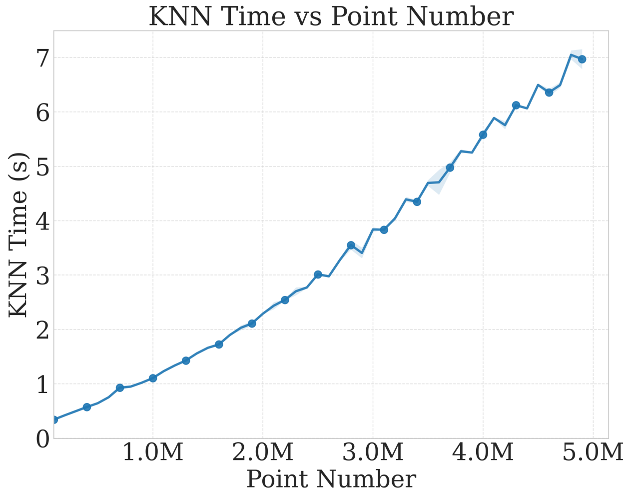

Figure 6. KNN Time vs Point Number

9. Robustness of rendering-based methods.

As discussed in Section 5.1, rendering-based baselines estimate importance by projecting Gaussians from all available training views (i.e., $7/8$ of the total views). To analyze their sensitivity to the number of views, we uniformly subsample $1/8$, $2/8$, $\ldots$, $7/8$ of the training views to recompute importance scores and evaluate post-hoc pruning results. As shown in Fig. 7, rendering-based methods exhibit notable instability when the number or distribution of input views changes. For instance, LightGaussian's pruning quality on the *bicycle* scene fluctuates by up to 1.5 dB PSNR across different view counts, while C3DGS on *playroom* suffers a 4 dB drop when using only $1/8$ of the views and still varies by about 1 dB across the remaining subsets. Besides, using more views does not always yield better scores: LightGaussian achieves its best performance with $4/8$ views, and C3DGS peaks at $5/8$. These results suggest that rendering-based importance estimation is highly dependent on the number and spatial distribution of selected views, whereas our RAP, relying solely on point-level features without rendering, is inherently view-agnostic and thus more robust.

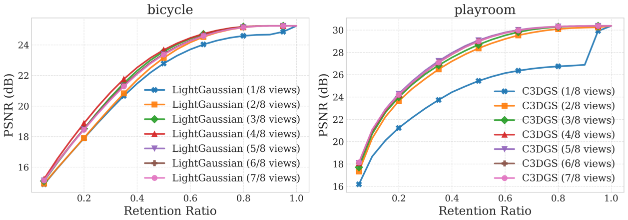

Figure 7. Robustness of rendering-based importance estimation to the number of training views used. Left: LightGaussian on the *bicycle* scene from Mip-NeRF360-Outdoor. Right: C3DGS on the *playroom* scene from Deep Blending. Each curve corresponds to a different subset of views ($1/8$–$7/8$) used for score computation.

10. Post-hoc Pruning: Additional Results

Per-scene results are shown in Fig. 10, RAP consistently achieves the highest PSNR on nearly all scenes, with the only exceptions being *bicycle* and *stump* from the Mip-NeRF360 Outdoor dataset, where EAGLES is approximately 0.3 dB better around the 60% pruning ratio. Visibility-based approaches such as LightGaussian, MesonGS, and EAGLES tend to perform slightly worse than RAP (typically within 0.3 dB). Although these methods differ in their use of opacity, blending opacity, or volume weighting, their overall behavior is largely comparable. Each achieves slightly better results on certain scenes and slightly worse on others, but none exhibits a consistent or systematic advantage across datasets.

In contrast, gradient-based methods (C3DGS, PUP-3DGS) are far less stable. They show substantial degradation on several scenes—*kitchen*, *playroom*, *truck*, and *train*—where their PSNR drops by 2–3 dB relative to RAP. At certain pruning ratios, their performance even falls be-

(a) Original scene

(b) RAP (ours, 5% retained)

(c) EAGLES (visibility-based, 5% retained)

(d) C3DGS (gradient-based, 5% retained)

Figure 8. Pruning behavior of different importance estimators at 5% retention. EAGLES favors central regions, C3DGS keeps edges, while RAP produces a more uniform, structure-preserving subset.

low the naive opacity baseline (e.g., C3DGS on *playroom*, PUP-3DGS on *truck*).

For SSIM and LPIPS, RAP typically ranks second or third on most Mip-NeRF360 scenes, with EAGLES often obtaining the best perceptual scores. To better understand these differences, we visualize the retained primitives on the *garden* scene in Fig. 8. The comparison reveals distinct selection biases across methods:

- **Visibility-based methods (e.g., EAGLES).** These methods strongly favor primitives near the scene center. This arises because: (1) training views surround the central object, causing central primitives to accumulate many projected contributions while background primitives receive far fewer; and (2) projected area decreases with depth, causing distant primitives to appear smaller and thus receive lower scores. Consequently, foreground structures are well preserved, but background regions are severely underrepresented.
- **Gradient-based methods (e.g., C3DGS).** These approaches mainly retain high-frequency edges, while smooth surfaces lose most of their support. Under heavy pruning, this leads to incomplete geometry and strong artifacts.
- **RAP (ours).** RAP produces a more uniform selection across the entire scene, preserving both foreground and background content. This explains why RAP achieves the strongest PSNR across datasets—PSNR benefits from globally consistent coverage—whereas SSIM and LPIPS, which emphasize structural similarity, may sometimes favor visibility-based techniques.

11. Loss Function Analysis and Score Distribution Visualization

To better understand how the pruning-aware loss and the distribution regularization shape the predicted scores, Fig. 9 visualizes the score distributions produced by RAP and its ablated variants. Without the pruning-aware loss, the network tends to assign overly large scores to most primitives, making it difficult to identify truly important points. In con-

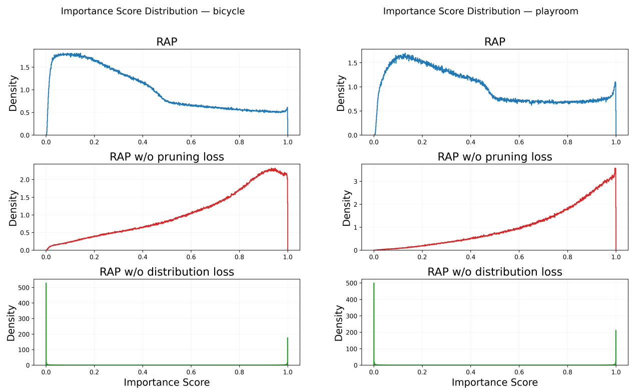

Figure 9. Predicted score distributions for RAP, RAP without pruning-aware loss, and RAP without distribution loss.

trast, removing the distribution loss causes the scores to collapse toward near-binary values around 0 and 1, which prevents setting flexible pruning thresholds and makes fine-grained importance discrimination unreliable. The full RAP model produces a smooth and well-spread distribution—dominated by low scores but with clear separation among mid- and high-importance primitives—enabling stable and accurate pruning across a wide range of ratios.

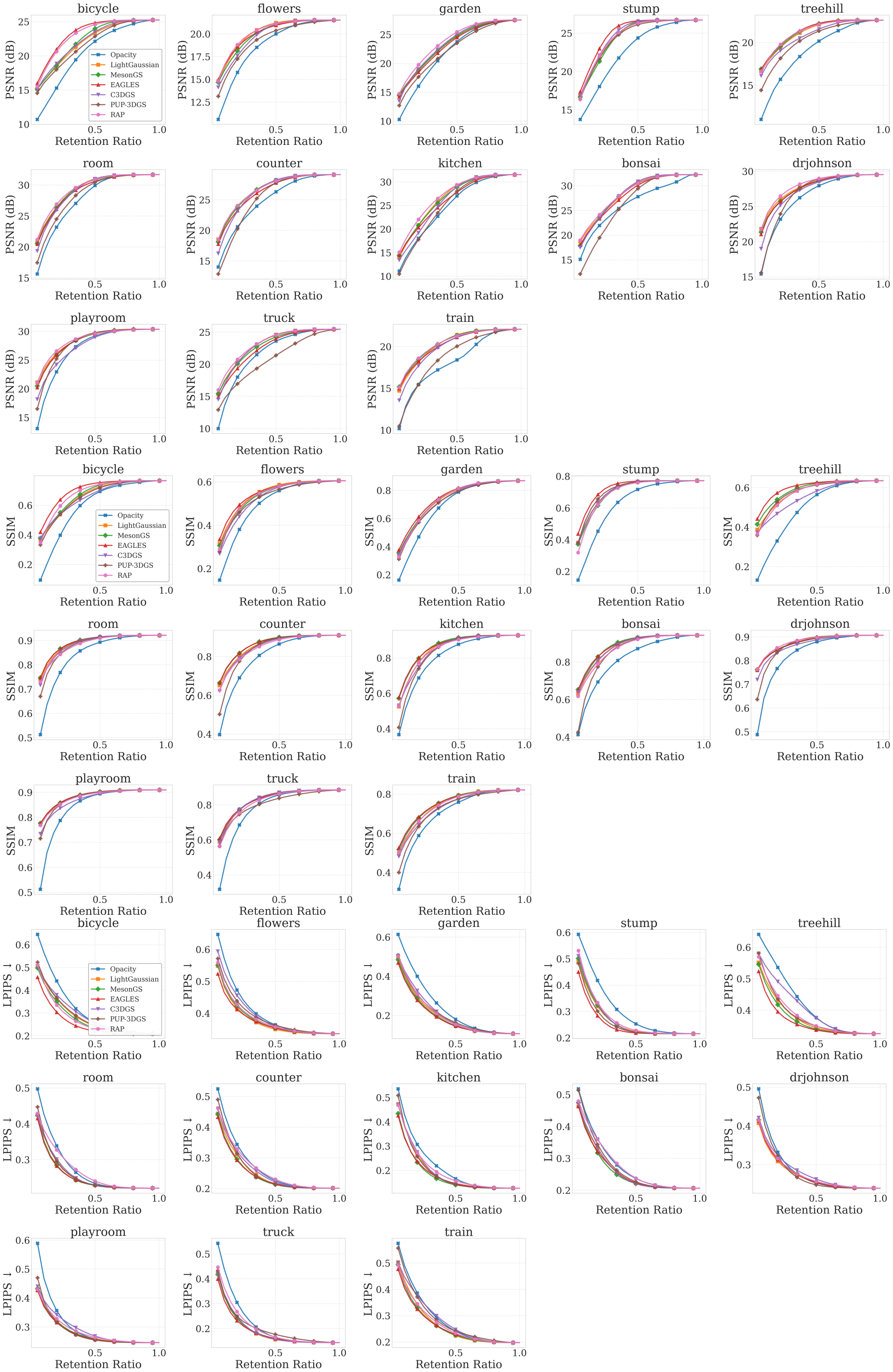

Figure 10. Per-scene post-hoc pruning results across three metrics: PSNR (top), SSIM (middle), and LPIPS (bottom).