Uncertainty-Aware Gaussian Splatting: Per-Splat Variance from Multi-View Visibility

Tarun Gangadhar Vadaparthi

Abstract

We provide a report on the uncertainty analysis of 3D Gaussian Splatting (3DGS) via a simple, reproducible pipeline on a pretrained stage (.ply). Given multi-view visibility, the defined uncertainty score is assigned per splat: Gaussian contributions are measured across an orbit of camera poses and summarized through the cross-view dispersion (standard deviation) to capture flicker owing to instabilities and view dependencies. No retraining/specific supervision by scene is needed. The method runs on a free Colab T4 GPU, rendering both a standard RGB image and an uncertainty heatmap where splats with large variance are highlighted. Qualitatively, tests in a "ship" scene show the method brings out the outer and noisy regions plus depth-unstable contributions yet keeps confident geometry that is visible consistently. This procedure can work as a concise diagnostic of 3DGS quality and pave the way for Bayesian, semantic, or task-aware uncertainty estimators going forward.

1 Introduction

3D Gaussian Splatting (3DGS) has recently acquired worldwide fame as the unique solution for real-time novel view synthesis and scene reconstruction. Since 3DGS stores scene content as a collection of 3D Gaussian primitives, each with its appearance and opacity, it can render scenes more efficiently than neural radiance fields, putting splats into rasterization in a flash while maintaining high quality. Rendering performance has pretty much been dissected; however, the twist of *uncertainty* in splat contributions remains largely unexplored.

Uncertainty estimation is crucial for evaluating the reliability of reconstruction results, identifying unstable regions, as well as informing downstream applications in robotics, simulation, or visual effects. For example, peripheral splats with noisy surfaces or inconsistent depths become unstable in renderings across different viewpoints but are usually not rendered into RGB by standard pipelines.

We implement a simple uncertainty analysis pipeline for 3DGS. A pretrained scene point cloud (.ply) is loaded, and splat contributions are evaluated along a number of camera poses on an orbital path. The uncertainty of a splat is measured as the variance across the views of contribution scores. Splats with very high variance are those that show instability or view dependence, while those with low variance are consistent and confident geometries. This initial diagnostic serves as groundwork for more advanced uncertainty-aware Gaussian splatting techniques.

2 Method

A pipeline of uncertaincies consists of three components: the sampling made from the circular positions of the camera, the score of contributions per-splat, and the initialization of variances among all landscapes.

2.1 Camera Orbit Sampling

Camera poses are defined along a circular orbit around the scene center. Each pose is built through a look-at transform with positions of eye evenly distributed on the azimuth, to assure uniform coverage and splats are exposed to multiple view directions. In practice, 12-24 orbit views offer enough diversity for stable statistics.

2.2 Per-Splat Contribution Score

For each view, splats are projected into the image plane by means of a pinhole camera model. The screen-space radius is approximated from the 3D scale parameters and depth. The contribution of each splat is derived from the opacity (sigmoid of the stored logit) and the footprint size of the splat after projection. A scalar visibility score is defined as:

$$s_i^{(v)} = \alpha_i \cdot \frac{r_i^2}{z_i^2},$$

where α_i is the splat alpha, r_i is its screen-space radius in pixels, and z_i is the camera-space depth. This proxy describes the extent with which this splat is involved in rendering for view v.

2.3 Cross-View Variance

Each sharp orbit view is scored by splat which is agglomerated. For evaluation one obtains an uncertainty defined as the quantified standard deviation.

$$u_i = \mathrm{Std\_}v(s_i^{(v)}),$$

where a small u_i denotes stable contributions and a large u_i denotes instability (flicker or disappearance across views). Lastly, to prevent extreme outliers, uncertainties are normalized using robust percentiles and mapped to a color scale (like Viridis) for visualization.

3 Experiments and Results

3.1 Setup

A pretrained 3DGS scene of a ship is used for the experiment. A stored point cloud file in PLY format, which includes positions, spherical harmonic coefficients, opacity logits, scale vectors, and rotation quaternions, is used to load the Gaussian splats. Splats are subsampled to 40k points in order to maintain computation viability on a Colab T4 GPU. The resolution of the rendered images is 256×256.

3.2 Uncertainty Statistics

Twelve orbit views are used to calculate visibility scores. Using robust statistics (5th and 95th percentiles) for normalization, the resulting per-splat uncertainty values span multiple orders of magnitude. While outliers draw attention to unstable splats at scene edges or surfaces with uneven depth, typical uncertainty ranges focus on low values.

3.3 Quantitative Statistics

Basic statistics were used to summarize the per-splat uncertainty distribution. Values varied from 0.0 to 2157.2 across 40k splats in the ship scene, with a mean of 0.28. The majority of splats show low uncertainty, according to robust percentiles ($p5 = 0.0009$, $p95 = 0.3365$), while the upper tail is dominated by a tiny percentage of outliers. The visual observation that high-uncertainty splats occur at scene peripheries and noisy regions is supported by these statistics.

3.4 Visualization

Two visualization modes are produced:

- **Normal render:** Standard RGB compositing of splats with depth sorting.
- **Uncertainty heatmap:** Using the Viridis colormap, each splat is colored according to normalized uncertainty. Stable splats appear dark blue/green, whereas high-variance splats appear as bright yellow regions.

Figure 1 shows an example view, comparing the normal render and the uncertainty heatmap side by side. For both normal and uncertainty-colored renders, orbit sequences consisting of 24 views are produced. The results, which are assembled as GIFs, show how uncertainty focuses on thin structures and peripheral geometry from various angles. These animations provide dynamic proof of the uncertainty metric's efficacy.

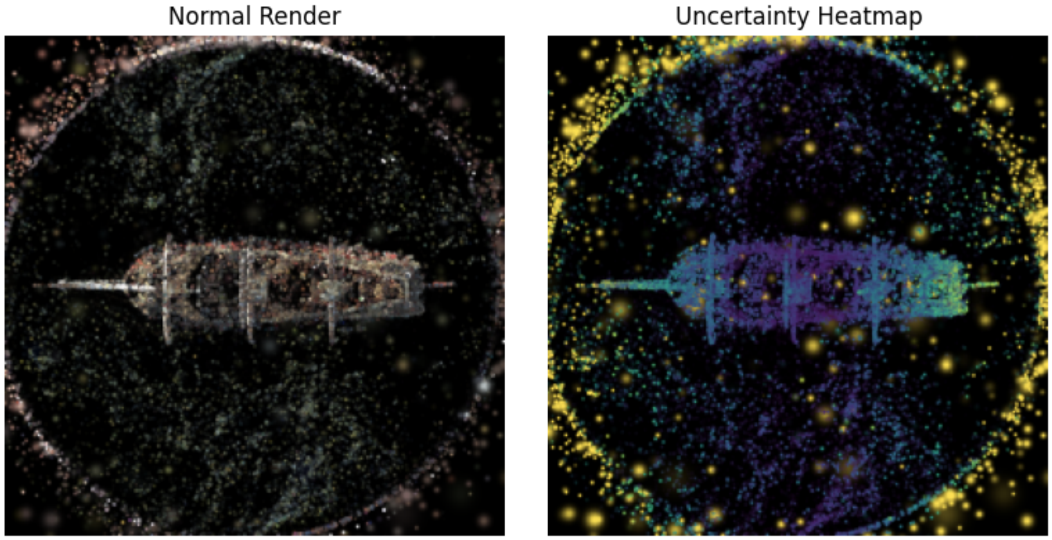

Figure 1: Comparison of a normal RGB render (left) and uncertainty heatmap (right). High-variance splats highlight unstable regions.

4 Discussion and Limitations

Without the need for retraining or additional data, the suggested uncertainty metric identifies view-dependent instability in 3D Gaussian Splatting. The findings indicate that while consistently visible splats maintain low uncertainty, high-variance splats frequently appear close to scene boundaries, thin structures, and noisy geometry. This behavior implies that the method can be used as a tool for downstream filtering or dataset curation, as well as a rapid diagnostic for model quality.

Nevertheless, the strategy has a number of drawbacks:

- The contribution score is a heuristic that combines depth, screen-space radius, and opacity. Despite its effectiveness, it misses some intricate interactions and rendering effects.
- The method does not provide calibrated probabilistic confidence; it only measures statistical variance.
- Orbit sampling affects uncertainty; too few views can cause instabilities to go unnoticed, while too many views lengthen computation times.
- Anisotropic covariance from the underlying model is not included in the current pipeline, which only visualizes isotropic splats.

The pipeline offers a lightweight foundation for Gaussian splatting uncertainty analysis in spite of these limitations.

5 Future Work

This initial uncertainty analysis can be expanded in a number of ways:

- **Probabilistic modeling:** Bayesian or variational formulations can be used in place of heuristic scores to estimate posterior uncertainty in splat parameters.
- **Coupling semantics:** To find class-specific instabilities (such as uncertain edges of objects), combine semantic labels with per-splat uncertainty. Dynamic scenes: Measure the uncertainty in time-varying splats for moving objects or changing environments by extending to 4D Gaussian splatting.
- **Calibration:** Examine ways to produce interpretable confidence intervals by correlating uncertainty estimates with actual rendering error.