ROSPlat: A Plug-and-Play Visualizer for 3D Gaussian Splatting in ROS2

Shady Gmira, s.gmira@student.rug.nl
University of Groningen - Faculty Science and Engineering - Nijenborgh 9, Groningen, Netherlands

Abstract: *We introduce ROSplat, the first real-time visualizer for 3D Gaussian Splatting fully integrated with the ROS2 ecosystem. ROSplat can visualize 3D Gaussian rendering, IMU data, and image streams via native ROS2 communication at real-time speed. Designed for scalability and ease of use, ROSplat supports adaptive buffer updates and different sorting and rendering backends, allowing smooth rendering performance even with millions of splats. Being tailored specifically for high-performance and incremental map update, ROSplat simplifies development and debugging of 3D Gaussian-based odometry, mapping, and other vision-based robotics tasks.*

Past Implementation and Problem Statement

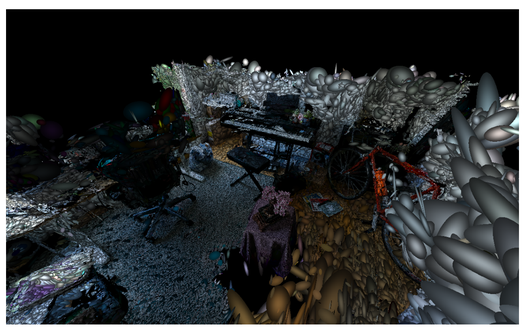
- 3D Gaussian Splatting (3DGS) enables **photorealistic scene reconstruction** using Gaussian primitives which are trained on a set of images from a specific scene[1].
- No existing solution offered a unified system for visualizing 3DGS in real-time within the ROS2 ecosystem, leading to fragmented workflows and difficult debugging.
- ROSplat addresses these limitations by integrating directly with ROS2 topics, real-time visualization of Gaussian splats during training and inference stages.
- This tight coupling with ROS2 enables seamless use in robotics pipelines.
- 3DGS renders a pixel color as:

$$C = \sum_i \alpha_i \, \mathcal{G}_i(\mathbf{u}) \, \mathbf{c}_i \prod_{j=1}^{i-1} (1 - \alpha_j) \,,$$

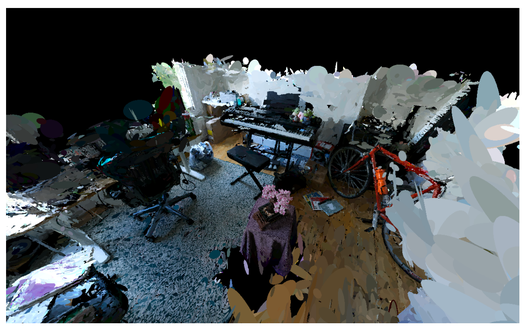
where α_i is the opacity, $\mathbf{c}_i$ the color, and $\mathcal{G}_i(\mathbf{u})$ the 2D projected Gaussian of the i-th primitive.

Key Features

- High-performance 3D Gaussian rendering using **OpenGL** and Shader Storage Buffer Objects (**SSBOs**), an advanced GPU technique enabling real-time updates and visualization of millions of splats with minimal latency.
- Supports dynamic updates from **live ROS2** topics or loading a static scene from a **PLY** file.
- Intuitive ImGui-based GUI with camera controls, rendering options, and shader toggling.
- Multi-modal visualization: RGB frames, IMU data, Gaussian splats.

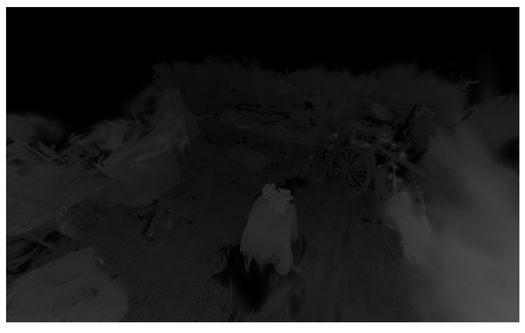


Gaussian Ball visualization.



Flat Ball rendering.

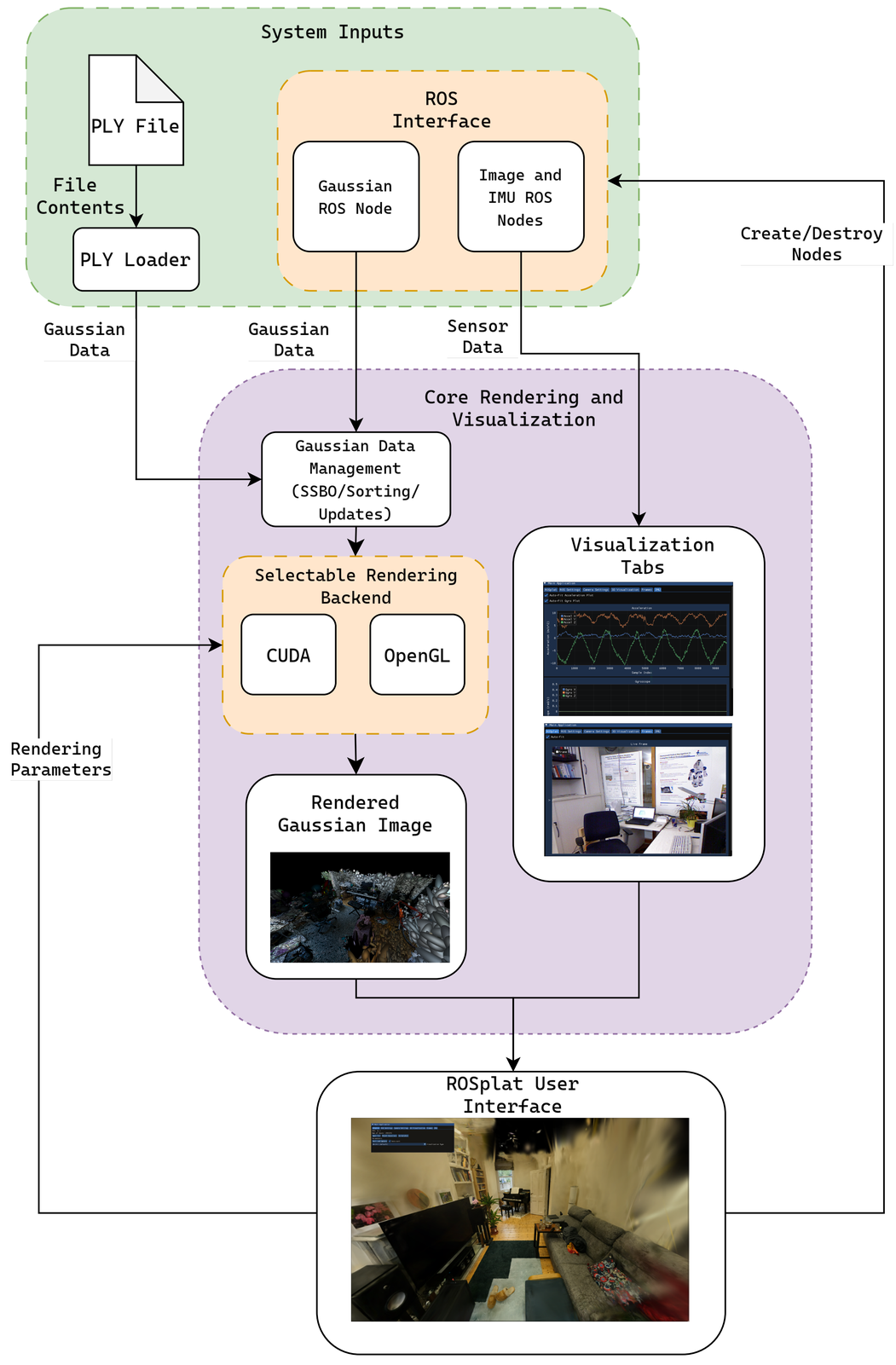
Billboard rendering style.



Depth-based shading mode.

Applications

- **Workflow for algorithm development with ROSplat:**
  1. Develop or train a 3DGS-based mapping or SLAM algorithm on a separate robot or simulation.
  2. Publish the resulting Gaussian splats, camera poses, and optional IMU data as ROS2 topics.
  3. Run ROSplat on a remote development machine or local robot.
  4. ROSplat subscribes to the published topics and renders the Gaussian splats in real time.
  5. Developers can inspect the scene interactively, validate performance, and debug geometry and pose accuracy.



System architecture showing ROS and PLY integration, rendering pipeline, and GUI interaction.

References

References

[1] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4), July 2023. https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/

Acknowledgments

Special thanks to Qihao Yuan and Kailai Li from ASIG for their guidance and encouragement, as well as the constructive feedback that helped shape this work.