GSplatLoc : Ultra-Precise Pose Optimization via 3D Gaussian Reprojection

https://spla-tam.github.io

Atticus Zhou, Atticus Zhou, Atticus Zhou, Atticus Zhou

July 19, 2024

Abstract

We present GSplatLoc, an innovative pose estimation method for RGB-D cameras that employs a volumetric representation of 3D Gaussians. This approach facilitates precise pose estimation by minimizing the loss based on the reprojection of 3D Gaussians from real depth maps captured from the estimated pose. Our method attains rotational errors close to zero and translational errors within 0.01mm, representing a substantial advancement in pose accuracy over existing point cloud registration algorithms, as well as explicit volumetric and implicit neural representation-based SLAM methods. Comprehensive evaluations demonstrate that GSplatLoc significantly improves pose estimation accuracy, which contributes to increased robustness and fidelity in real-time 3D scene reconstruction, setting a new standard for localization techniques in dense mapping SLAM.

1 Introduction

We present GSplatLoc, an innovative pose estimation method for RGB-D cameras that employs a volumetric representation of 3D Gaussians. This approach facilitates precise pose estimation by minimizing the loss based on the reprojection of 3D Gaussians from real depth maps captured from the estimated pose. Our method attains rotational errors close to zero and translational errors within 0.01mm, representing a substantial advancement in pose accuracy over existing point cloud registration algorithms, as well as explicit volumetric and implicit neural representation-based SLAM methods. Comprehensive evaluations demonstrate that GSplatLoc significantly improves pose estimation accuracy, which contributes to increased robustness and fidelity in real-time 3D scene reconstruction, setting a new standard for localization techniques in dense mapping SLAM.

2 Related Work

We present GSplatLoc, an innovative pose estimation method for RGB-D cameras that employs a volumetric representation of 3D Gaussians. This approach facilitates precise pose estimation by minimizing the loss based on the reprojection of 3D Gaussians from real depth maps captured from the estimated pose. Our method attains rotational errors close to zero and translational errors within 0.01mm, representing a substantial advancement in pose accuracy over existing point cloud registration algorithms, as well as explicit volumetric and implicit neural representation-based SLAM methods. Comprehensive evaluations demonstrate that GSplatLoc significantly improves pose estimation accuracy, which contributes to increased robustness and fidelity in real-time 3D scene reconstruction, setting a new standard for localization techniques in dense mapping SLAM.

3 Method

3.1 Gaussian Splatting

Depth-only Gaussian splatting is a highly effective method for modeling 3D scenes and producing depth maps. In our methodology, we initiate 3D Gaussians from a dense point cloud acquired via a RGB-D camera. Let $\mathcal{G} = \{G_i\}_{i=1}^N$ be a set of N 3D Gaussians, where each Gaussian G_i is defined as follows:

$$G_i = (\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, o_i)$$

where $\boldsymbol{\mu}_i \in \mathbb{R}^3$ is the 3D mean, $\boldsymbol{\Sigma}_i \in \mathbb{R}^{3\times 3}$ denotes the 3D covariance matrix, and $o_i \in \mathbb{R}$ signifies the opacity. Initially, we uniformly set $o_i = 1$ for all Gaussians to ensure they are fully opaque.

The projection of a 3D Gaussian onto the 2D image plane is computed as:

$$\mu_I = \pi(P(T_{wc}\mu_{\text{homogeneous}}))$$

where $T_{wc} \in SE(3)$ denotes the world-to-camera transformation, $P \in \mathbb{R}^{4\times 4}$ represents the projection matrix, and $\pi : \mathbb{R}^4 \to \mathbb{R}^2$ maps to pixel coordinates.

The 2D covariance Σ_I of a projected Gaussian is given by:

$$\Sigma_I = JR_{wc}\Sigma R_{wc}^T J^T$$

where R_{wc} is the rotation component of T_{wc}, and J denotes the Jacobian of the projection.

To generate the depth map, we employ a front-to-back compositing strategy. For each pixel p, its depth value d_p is computed as:

$$d_p = \sum_i w_i z_i$$

where z_i represents the depth of the i-th Gaussian's mean, and w_i is the weight derived from the 2D Gaussian distribution:

$$w_i = \exp\left(-\frac{1}{2}(x_p - \mu_{I,i})^T \Sigma_{I,i}^{-1}(x_p - \mu_{I,i})\right)$$

Here, x_p is the 2D coordinate of pixel p, $\mu_{I,i}$ and $\Sigma_{I,i}$ denote the projected mean and covariance of the i-th Gaussian.

This approach enables efficient depth map generation by leveraging the dense point cloud captured by the depth camera, without requiring colour information.

3.2 Depth Reprojection

Depth at a pixel i is represented by combining contributions from multiple Gaussian elements, each associated with a certain depth and confidence. Depth D_i can be expressed as[1]:

$$D_i = \frac{\sum_{n\leq N} d_n \cdot c_n \cdot \alpha_n \cdot T_n}{\sum_{n\leq N} c_n \cdot \alpha_n \cdot T_n}$$

d_n is the depth value from the n-th Gaussian, c_n is the confidence or weight of the n-th Gaussian,α_n is the opacity calculated from Gaussian parameters, T_n is the product of transparencies from all Gaussians in front of the n-th Gaussian.

The reprojection method utilizes the alignment of 2D Gaussian projections with observed depth data from an RGB-D camera. This involves adjusting the parameters of the Gaussians to minimize the discrepancy between the projected depth and the observed depth. The offset Δ_n and the covariance matrix Σ' are crucial for calculating the Gaussian weights α_n and their impact on reprojection accuracy.

3.3 Camera Tracking

We define the camera pose as

$$\mathbf{T}_{cw} = \begin{pmatrix} \mathbf{R}_{cw} & \mathbf{t}_{cw} \\ \mathbf{0} & 1 \end{pmatrix} \in SE(3)$$

where $\mathbf{T}_{cw}$ represents the camera-to-world transformation matrix. Notably, we parameterize the rotation $\mathbf{R}_{cw} \in SO(3)$ using a quaternion $\mathbf{q}_{cw}$. This choice of parameterization is motivated by several key advantages that quaternions offer in the context of camera pose estimation and optimization. Quaternions provide a compact and efficient representation, requiring only four parameters, while maintaining numerical stability and avoiding singularities such as gimbal lock. Their continuous and non-redundant nature is particularly advantageous for gradient-based optimization algorithms, allowing for unconstrained optimization and simplifying the optimization landscape.

Based on these considerations, we design our optimization variables to separately optimize the normalized quaternion and the translation. The loss function is designed to ensure accurate depth estimations and edge alignment, incorporating both depth magnitude and contour accuracy. It can be defined as:

$$L = \lambda_1 \cdot L_{\text{depth}} + \lambda_2 \cdot L_{\text{contour}}$$

where L_{depth} represents the L1 loss for depth accuracy, and L_{contour} focuses on the alignment of depth contours or edges. Specifically:

$$L_{\text{depth}} = \sum_{i\in M} |D_i^{\text{predicted}} - D_i^{\text{observed}}|$$

$$L_{\text{contour}} = \sum_{j\in M} |\nabla D_j^{\text{predicted}} - \nabla D_j^{\text{observed}}|$$

Here, M denotes the reprojection mask, indicating which pixels are valid for reprojection. Both L_{depth} and L_{contour} are computed only over the masked regions. λ_1 and λ_2 are weights that balance the two parts of the loss function, tailored to the specific requirements of the application.

The optimization objective can be formulated as:

$$\min_{\mathbf{q}_{cw},\mathbf{t}_{cw}} L + \lambda_q \|\mathbf{q}_{cw}\|_2^2 + \lambda_t \|\mathbf{t}_{cw}\|_2^2$$

where λ_q and λ_t are regularization terms for the quaternion and translation parameters, respectively.

We employ the Adam optimizer for both quaternion and translation optimization, with different learning rates and weight decay values for each. The learning rates are set to $5{\times}10^{-}4$ for quaternion optimization and $10^{-}3$ for translation optimization, based on experimental results. The weight decay values are set to $10^{-}3$ for both quaternion and translation parameters, serving as regularization to prevent overfitting.

4 Experiments

We present GSplatLoc, an innovative pose estimation method for RGB-D cameras that employs a volumetric representation of 3D Gaussians. This approach facilitates precise pose estimation by minimizing the loss based on the reprojection of 3D Gaussians from real depth maps captured from the estimated pose. Our method attains rotational errors close to zero and translational errors within 0.01mm, representing a substantial advancement in pose accuracy over existing point cloud registration algorithms, as well as explicit volumetric and implicit neural representation-based SLAM methods. Comprehensive evaluations demonstrate that GSplatLoc significantly improves pose estimation accuracy, which contributes to increased robustness and fidelity in real-time 3D scene reconstruction, setting a new standard for localization techniques in dense mapping SLAM.

5 Conclusion

We present GSplatLoc, an innovative pose estimation method for RGB-D cameras that employs a volumetric representation of 3D Gaussians. This approach facilitates precise pose estimation by minimizing the loss based on the reprojection of 3D Gaussians from real depth maps captured from the estimated pose. Our method attains rotational errors close to zero and translational errors within 0.01mm, representing a substantial advancement in pose accuracy over existing point cloud registration algorithms, as well as explicit volumetric and implicit neural representation-based SLAM methods. Comprehensive evaluations demonstrate that GSplatLoc significantly improves pose estimation accuracy, which contributes to increased robustness and fidelity in real-time 3D scene reconstruction, setting a new standard for localization techniques in dense mapping SLAM.

[1] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, pp. 1–14, 2023, doi: 10.1145/3592433.