

SLAM & 3D Gaussian Splatting

Literature Review

Shuqi XIAO

June 26, 2024

1 Overview

2 MonoGS

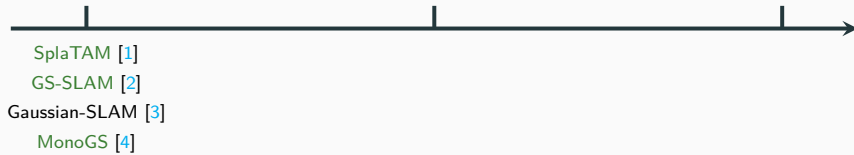
- Methodology

Overview

11/2023 - 12/2024

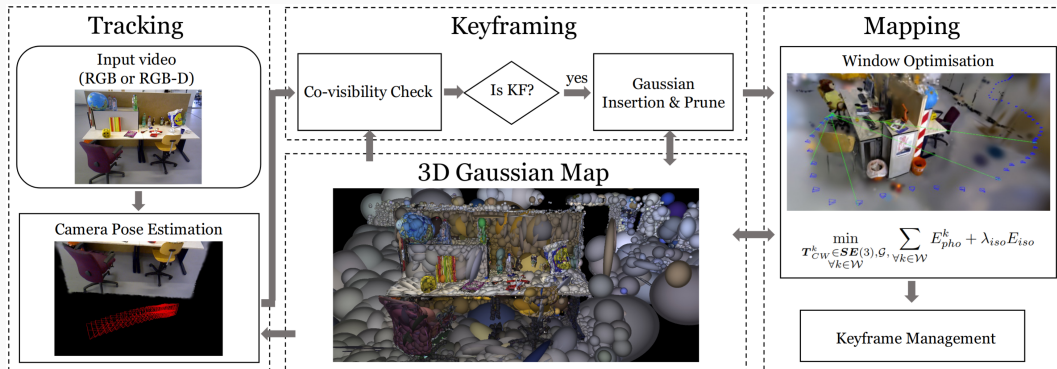
01/2024 - 04/2024

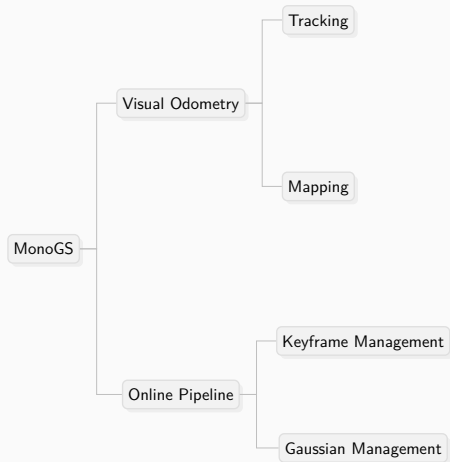
05/2024 - 06/2024

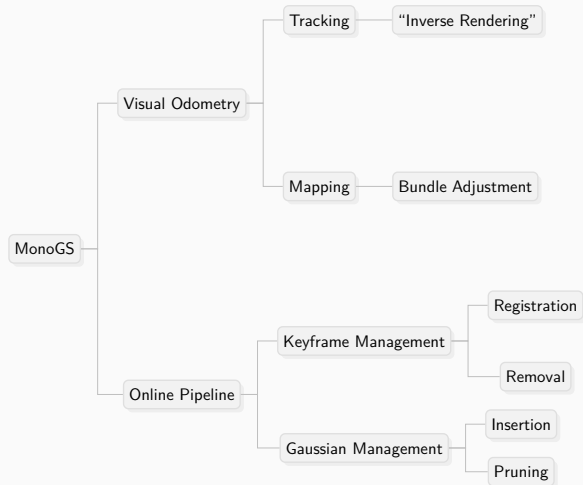


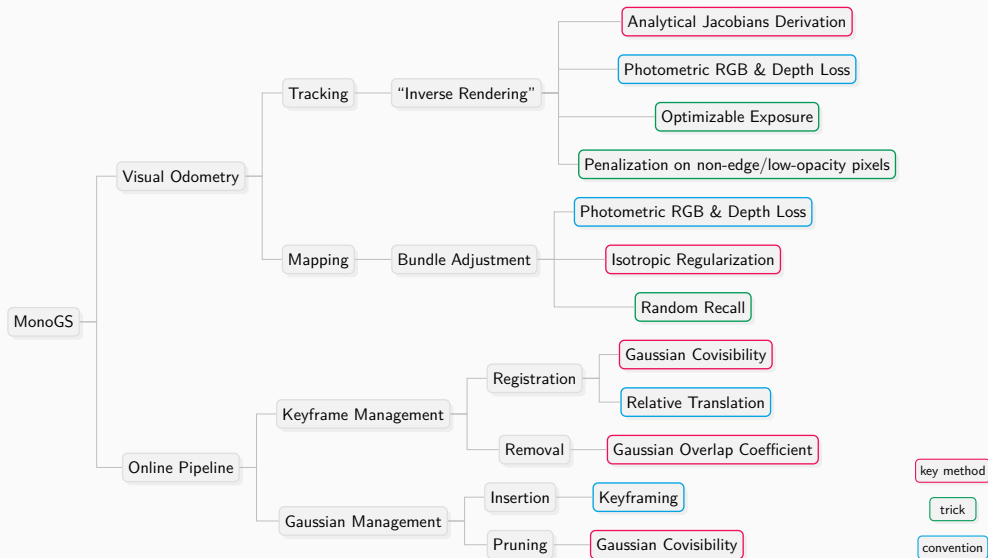
MonoGS

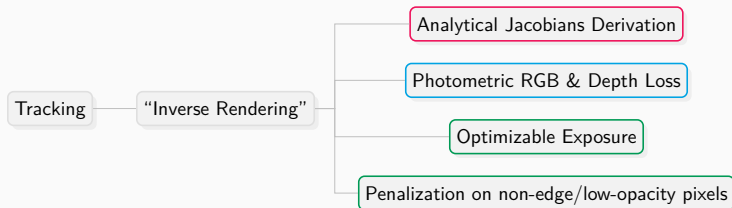






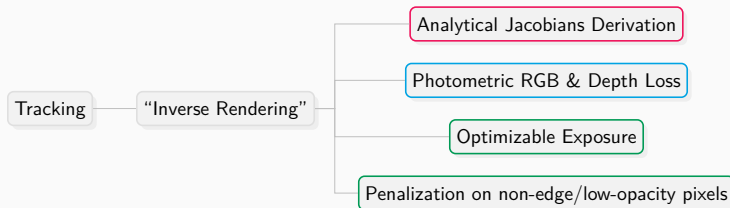






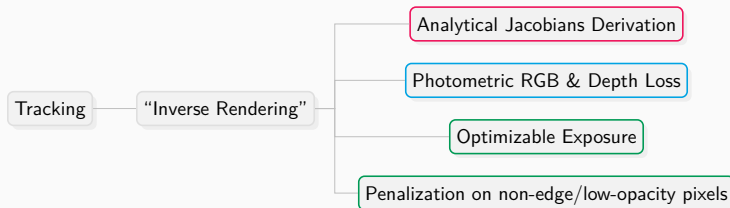
Track camera poses,

- through the extended differentiable rendering pipeline,



Track camera poses,

- through the extended differentiable rendering pipeline,
- by a direct optimization against fixed 3D Gaussians,



Track camera poses,

- through the extended differentiable rendering pipeline,
- by a direct optimization against fixed 3D Gaussians,
- with some tricks to be more adaptive to brightness and more robust to noise.

The projection from “ellipsoids” to “ellipses” in 3DGS,

$$\mathcal{N}(\mu_w, \Sigma_w) \xrightarrow{\pi} \mathcal{N}(\mu_i, \Sigma_i), \quad (1)$$

is achieved by,

$$\mu_i = \pi(\mathbf{T}_{cw} \cdot \mu_w) \quad (2)$$

$$\Sigma_i = \mathbf{J}_\pi \mathbf{R}_{cw} \Sigma_w \mathbf{R}_{cw}^T \mathbf{J}_\pi^T \quad (3)$$

The projection from “ellipsoids” to “ellipses” in 3DGS,

$$\mathcal{N}(\mu_w, \Sigma_w) \xrightarrow{\pi} \mathcal{N}(\mu_i, \Sigma_i), \quad (1)$$

is achieved by,

$$\mu_i = \pi \left(\mathbf{T}_{cw} \cdot \mu_w \right)$$

Diagram illustrating the projection of the 3D world mean μ_w to the 2D image mean μ_i . The world mean μ_w is a 3D vector in \mathbb{P}^3 . It is transformed by the camera pose \mathbf{T}_{cw} (an element of $\text{SE}(3)$) and then projected by π to the 2D image mean μ_i in \mathbb{P}^2 . The projection π is indicated by a green arrow.

$$\Sigma_i = \mathbf{J}_\pi \mathbf{R}_{cw} \Sigma_w \mathbf{R}_{cw}^T \mathbf{J}_\pi^T$$

Diagram illustrating the projection of the 3D world covariance Σ_w to the 2D image covariance Σ_i . The world covariance Σ_w is a 3×3 matrix in $\mathbb{R}^{3 \times 3}$. It is transformed by the rotation component \mathbf{R}_{cw} (an element of $\text{SO}(3)$) and the Jacobian of the linear approximation of π , \mathbf{J}_π (a 2×3 matrix in $\mathbb{R}^{2 \times 3}$), to the 2D image covariance Σ_i in $\mathbb{R}^{2 \times 2}$.

The chain rule,

$$\frac{\partial \mu_i}{\partial \mathbf{T}_{cw}} = \frac{\partial \mu_i}{\partial \mu_c} \frac{\partial \mu_c}{\partial \mathbf{T}_{cw}} \quad (4)$$

$$\frac{\partial \Sigma_i}{\partial \mathbf{T}_{cw}} = \frac{\partial \Sigma_i}{\partial \mathbf{J}_\pi} \frac{\partial \mathbf{J}_\pi}{\partial \mu_c} \frac{\partial \mu_c}{\partial \mathbf{T}_{cw}} + \frac{\partial \Sigma_i}{\partial \mathbf{R}_{cw}} \frac{\partial \mathbf{R}_{cw}}{\partial \mathbf{T}_{cw}} \quad (5)$$

The chain rule,

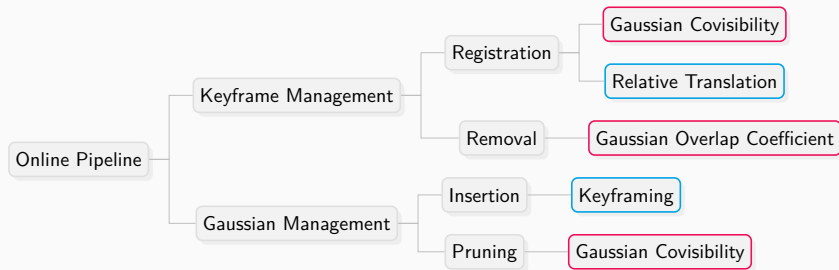
$$\frac{\partial \mu_i}{\partial \mathbf{T}_{cw}} = \frac{\partial \mu_i}{\partial \mu_c} \frac{\partial \mu_c}{\partial \mathbf{T}_{cw}} \quad (4)$$

$$\frac{\partial \Sigma_i}{\partial \mathbf{T}_{cw}} = \frac{\partial \Sigma_i}{\partial \mathbf{J}_\pi} \frac{\partial \mathbf{J}_\pi}{\partial \mu_c} \frac{\partial \mu_c}{\partial \mathbf{T}_{cw}} + \frac{\partial \Sigma_i}{\partial \mathbf{R}_{cw}} \frac{\partial \mathbf{R}_{cw}}{\partial \mathbf{T}_{cw}} \quad (5)$$

The Lie Algebra,

$$\frac{\partial \mu_c}{\partial \mathbf{T}_{cw}} = [\mathbf{I} \quad -\mu_c^\times] \quad (6)$$

$$\frac{\partial \mathbf{R}_{cw}}{\partial \mathbf{T}_{cw}} = \begin{bmatrix} \mathbf{0} & -\mathbf{R}_{cw}^\times(:, 1) \\ \mathbf{0} & -\mathbf{R}_{cw}^\times(:, 2) \\ \mathbf{0} & -\mathbf{R}_{cw}^\times(:, 3) \end{bmatrix} \quad (7)$$



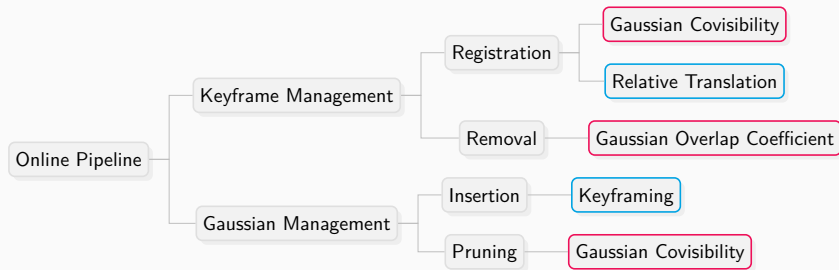
Keyframe Management:

- Classic keyframing strategies from DSO [5].

key method trick convention

(arXiv, 2016) DSO: Direct Sparse Odometry

(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM



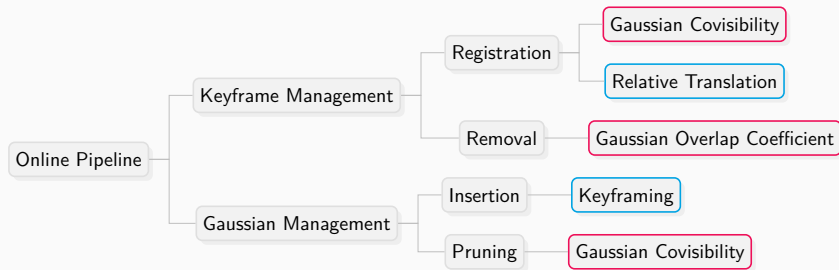
Keyframe Management:

- Classic keyframing strategies from DSO [5].
- Occlusion-aware Gaussian visibility is leveraged to construct covisibility and overlap metrics.

key method trick convention

(arXiv, 2016) DSO: Direct Sparse Odometry

(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM



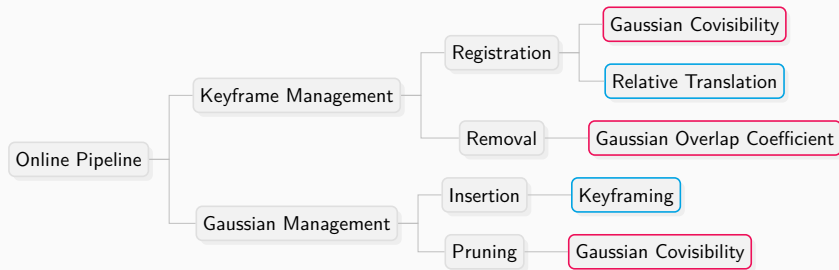
Gaussian Management:

- Insertion is triggered by keyframing and means Gaussian initialization.

key method trick convention

(arXiv, 2016) DSO: Direct Sparse Odometry

(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM



Gaussian Management:

- Insertion is triggered by keyframing and means Gaussian initialization.
- Pruning unstable/incorrect Gaussians by covisibility for better geometry in a monocular setting.

key method trick convention

(arXiv, 2016) DSO: Direct Sparse Odometry

(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM

1 What is keyframing?

1 What is keyframing?

- A strategy of selecting and utilizing a subset of frames that matters.

1 What is keyframing?

- A strategy of selecting and utilizing a subset of frames that matters.

2 Why do we need keyframing?

1 What is keyframing?

- A strategy of selecting and utilizing a subset of frames that matters.

2 Why do we need keyframing?

- Infeasible to optimize jointly on all frames online.

1 What is keyframing?

- A strategy of selecting and utilizing a subset of frames that matters.

2 Why do we need keyframing?

- Infeasible to optimize jointly on all frames online.
- Trade-off between efficiency and accuracy/robustness/...

1 What is keyframing?

- A strategy of selecting and utilizing a subset of frames that matters.

2 Why do we need keyframing?

- Infeasible to optimize jointly on all frames online.
- Trade-off between efficiency and accuracy/robustness/...

3 How should we select keyframes? (What is an ideal keyframe selection?)

1 What is keyframing?

- A strategy of selecting and utilizing a subset of frames that matters.

2 Why do we need keyframing?

- Infeasible to optimize jointly on all frames online.
- Trade-off between efficiency and accuracy/robustness/...

3 How should we select keyframes? (What is an ideal keyframe selection?)

- a small sliding window.

1 What is keyframing?

- A strategy of selecting and utilizing a subset of frames that matters.

2 Why do we need keyframing?

- Infeasible to optimize jointly on all frames online.
- Trade-off between efficiency and accuracy/robustness/...

3 How should we select keyframes? (What is an ideal keyframe selection?)

- a small sliding window.
- non-redundant and observing the same area.

1 What is keyframing?

- A strategy of selecting and utilizing a subset of frames that matters.

2 Why do we need keyframing?

- Infeasible to optimize jointly on all frames online.
- Trade-off between efficiency and accuracy/robustness/...

3 How should we select keyframes? (What is an ideal keyframe selection?)

- a small sliding window.
- non-redundant and observing the same area.
- spanning a wide baseline for better multi-view constraints.

If any of the following conditions is true...

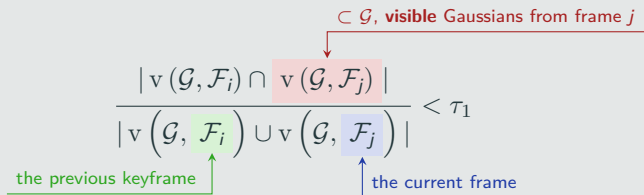
If any of the following conditions is true...

Small Gaussian Covisibility

Condition i, Keyframe Registration

Gaussian covisibility between the current frame and the previous keyframe drops below a threshold.

$$\frac{|\mathbf{v}(\mathcal{G}, \mathcal{F}_i) \cap \mathbf{v}(\mathcal{G}, \mathcal{F}_j)|}{|\mathbf{v}(\mathcal{G}, \mathcal{F}_i) \cup \mathbf{v}(\mathcal{G}, \mathcal{F}_j)|} < \tau_1$$



$\subset \mathcal{G}$, visible Gaussians from frame j

the previous keyframe

the current frame

If any of the following conditions is true...

Small Gaussian Covisibility

Condition i, Keyframe Registration

Gaussian covisibility between the current frame and the previous keyframe drops below a threshold.

$$\frac{|\mathbf{v}(\mathcal{G}, \mathcal{F}_i) \cap \mathbf{v}(\mathcal{G}, \mathcal{F}_j)|}{|\mathbf{v}(\mathcal{G}, \mathcal{F}_i) \cup \mathbf{v}(\mathcal{G}, \mathcal{F}_j)|} < \tau_1 \quad (8)$$

Large Relative Translation

Condition ii, Keyframe Registration

Translation from the previous keyframe w.r.t. to the median depth reaches a threshold.

$$\frac{\|\mathbf{t}_{\mathcal{F}_i \mathcal{F}_j}\|_2}{\bar{D}_{\mathcal{F}_i \mathcal{F}_j}} > \tau_2, \quad \bar{D}_{\mathcal{F}_i \mathcal{F}_j} = \frac{1}{2 \underset{\substack{\uparrow \\ \text{image height}}}{H} \underset{\substack{\uparrow \\ \text{image width}}}{W}} \sum_{\{\mathcal{F}_i, \mathcal{F}_j\}} \sum_{h=0}^H \sum_{w=0}^W \underset{\substack{\uparrow \\ \text{depth of pixel } (h, w)}}{d(h, w)}$$

If any of the following conditions is true...

Small Gaussian Covisibility

Condition i, Keyframe Registration

Gaussian covisibility between the current frame and the previous keyframe drops below a threshold.

$$\frac{|\mathbf{v}(\mathcal{G}, \mathcal{F}_i) \cap \mathbf{v}(\mathcal{G}, \mathcal{F}_j)|}{|\mathbf{v}(\mathcal{G}, \mathcal{F}_i) \cup \mathbf{v}(\mathcal{G}, \mathcal{F}_j)|} < \tau_1 \quad (8)$$

Large Relative Translation

Condition ii, Keyframe Registration

Translation from the previous keyframe w.r.t. to the median depth reaches a threshold.

$$\frac{\|\mathbf{t}_{\mathcal{F}_i \mathcal{F}_j}\|_2}{\bar{D}_{\mathcal{F}_i \mathcal{F}_j}} > \tau_2, \quad \bar{D}_{\mathcal{F}_i \mathcal{F}_j} = \frac{1}{2HW} \sum_{\{\mathcal{F}_i, \mathcal{F}_j\}} \sum_{h=0}^H \sum_{w=0}^W d(h, w) \quad (9)$$

If any of the following conditions is true,

Beyond Window Capacity

Condition i, Keyframe Removal

Remove the earliest keyframe out of the sliding window if the capacity is exceeded.

$$|\mathcal{W}| < \tau_3 \quad (10)$$

If any of the following conditions is true,

Beyond Window Capacity

Condition i, Keyframe Removal

Remove the earliest keyframe out of the sliding window if the capacity is exceeded.

$$|\mathcal{W}| < \tau_3 \quad (10)$$

Low Gaussian Overlap Coefficient

Condition ii, Keyframe Removal

Remove the previous keyframe if the “Gaussian overlap coefficient” between the previous frame and the new keyframe drops below a threshold.

$$\frac{|\mathbf{v}(\mathcal{G}, \mathcal{F}_i) \cap \mathbf{v}(\mathcal{G}, \mathcal{F}_j)|}{\min(|\mathbf{v}(\mathcal{G}, \mathcal{F}_i)|, |\mathbf{v}(\mathcal{G}, \mathcal{F}_j)|)} < \tau_4 \quad (11)$$

- Why do we need “Gaussian insertion”?

- Why do we need “Gaussian insertion”?
 - SLAM is for robotic exploration.

- Why do we need “Gaussian insertion”?
 - SLAM is for robotic exploration.
- When do we need “Gaussian insertion”?

- Why do we need “Gaussian insertion”?
 - SLAM is for robotic exploration.
- When do we need “Gaussian insertion”?

Keyframing

Condition i, Gaussian Insertion

Insertion is triggered for every new keyframe.

- How do we insert Gaussians?

In practice, “low”: 0.2σ ; “high”: 0.5σ , where σ is the standard deviation of the rendered depth map.
(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM

- How do we insert Gaussians?
 - Gaussian insertion is Gaussian **initialization**.

In practice, “low”: 0.2σ ; “high”: 0.5σ , where σ is the standard deviation of the rendered depth map.
(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM

- How do we insert Gaussians?
 - Gaussian insertion is Gaussian initialization.

If Depth Available

Gaussian Initialization

Back-project in a per-pixel, per-Gaussian approach.

- How do we insert Gaussians?
 - Gaussian insertion is Gaussian initialization.

If Depth Available

Gaussian Initialization

Back-project in a per-pixel, per-Gaussian approach.

If Depth Unavailable

Gaussian Initialization

Leverage the rendered depth map.

In practice, “low”: 0.2σ ; “high”: 0.5σ , where σ is the standard deviation of the rendered depth map.

(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM

- How do we insert Gaussians?
 - Gaussian insertion is Gaussian initialization.

If Depth Available

Gaussian Initialization

Back-project in a per-pixel, per-Gaussian approach.

If Depth Unavailable

Gaussian Initialization

Leverage the rendered depth map.

- **for pixels with depth:** use the rendered depth and assign a “low” covariance.

In practice, “low”: 0.2σ ; “high”: 0.5σ , where σ is the standard deviation of the rendered depth map.

(CVPR Highlight, 2024) [MonoGS: Gaussian Splatting SLAM](#)

- How do we insert Gaussians?
 - Gaussian insertion is Gaussian initialization.

If Depth Available

Gaussian Initialization

Back-project in a per-pixel, per-Gaussian approach.

If Depth Unavailable

Gaussian Initialization

Leverage the rendered depth map.

- for pixels with depth: use the rendered depth and assign a “low” covariance.
- for pixels w/o depth: use the median of rendered depth and assign a “high” covariance.

In practice, “low”: 0.2σ ; “high”: 0.5σ , where σ is the standard deviation of the rendered depth map.

(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM

- Why do we need “Gaussian Pruning” if depth unavailable?

If no pruning, although the majority of incorrect Gaussians vanish quickly in following optimization, there are some survivals.

In practice, the opacity threshold is 0.7.

In practice, the pruned Gaussians are inserted in the last 3 keyframes and unobserved by any other 3 keyframes in the sliding window.

(CVPR Highlight, 2024) [MonoGS: Gaussian Splatting SLAM](#)

- Why do we need “Gaussian Pruning” if **depth unavailable**?
 - Too many **incorrect/unstable** newly inserted Gaussians.

If no pruning, although the majority of incorrect Gaussians vanish quickly in following optimization, there are some survivals.

In practice, the opacity threshold is 0.7.

In practice, the pruned Gaussians are inserted in the last 3 keyframes and unobserved by any other 3 keyframes in the sliding window.

(CVPR Highlight, 2024) [MonoGS: Gaussian Splatting SLAM](#)

- Why do we need “Gaussian Pruning” if **depth unavailable**?
 - Too many incorrect/unstable newly inserted Gaussians.

Low Gaussian Opacity

Condition i, Gaussian Pruning

The Gaussians with a “low” opacity are pruned.

If no pruning, although the majority of incorrect Gaussians vanish quickly in following optimization, there are some survivals.

In practice, the opacity threshold is 0.7.

In practice, the pruned Gaussians are inserted in the last 3 keyframes and unobserved by any other 3 keyframes in the sliding window.

(CVPR Highlight, 2024) [MonoGS: Gaussian Splatting SLAM](#)

- Why do we need “Gaussian Pruning” if **depth unavailable**?
 - Too many incorrect/unstable newly inserted Gaussians.

Low Gaussian Opacity

Condition i, Gaussian Pruning

The Gaussians with a “low” opacity are pruned.

Low Gaussian Covisibility

Condition ii, Gaussian Pruning

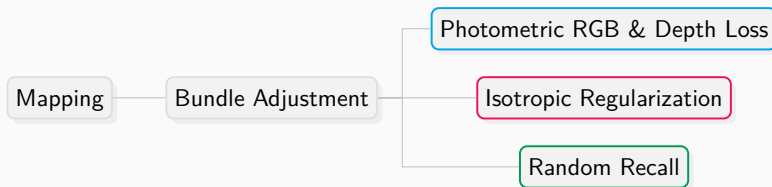
For “just” inserted Gaussians but unobserved by “some other” keyframes, are pruned out.

If no pruning, although the majority of incorrect Gaussians vanish quickly in following optimization, there are some survivals.

In practice, the opacity threshold is 0.7.

In practice, the pruned Gaussians are inserted in the last 3 keyframes and unobserved by any other 3 keyframes in the sliding window.

(CVPR Highlight, 2024) [MonoGS: Gaussian Splatting SLAM](#)

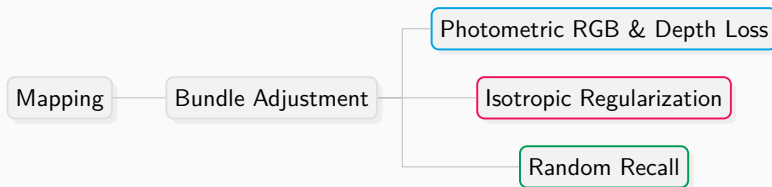


key method

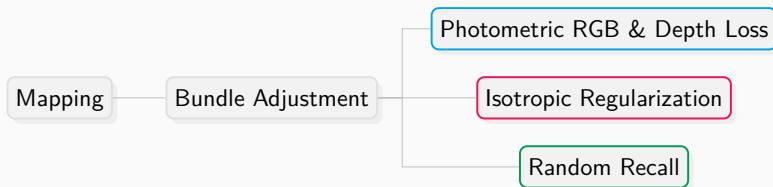
trick

convention

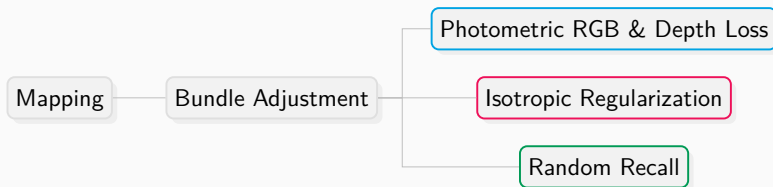
(CVPR Highlight, 2024) MonoGS: Gaussian Splatting SLAM



- Why do we need mapping in **3DGS** SLAM?



- Why do we need mapping in **3DGS** SLAM?
 - **Local Mapping**: Optimize newly inserted 3D Gaussians.



- Why do we need mapping in **3DGS** SLAM?
 - Local Mapping: Optimize newly inserted 3D Gaussians.
 - **Global Mapping**: Reconstruct a 3D-coherent structure.

Bundle Adjustment

$$\operatorname{argmin}_{\mathcal{G}, \{\mathbf{T}_{cw}(\mathcal{F}_k) | \mathcal{F}_k \in \mathcal{W}\}} \sum_{\mathcal{F}_k} \mathcal{L}_{pho}(\mathcal{F}_k) \quad (12)$$

keyframes in the sliding window
↓
 \mathcal{W}

Bundle Adjustment

$$\operatorname{argmin}_{\mathcal{G}, \{\mathbf{T}_{cw}(\mathcal{F}_k) | \mathcal{F}_k \in \mathcal{W}\}} \sum_{\mathcal{F}_k} \mathcal{L}_{pho}(\mathcal{F}_k) \quad (12)$$

keyframes in the sliding window
↓
 \mathcal{W}

Random Recall

A trick for global mapping

Besides \mathcal{W} , “some” randomly selected past keyframes are also leveraged in BA to avoid forgetting the global map.

- Why do we need “isotropic regularization”?

Isotropic Regularization

$$\mathcal{L}_{iso} = \sum_{i=1}^{|\mathcal{G}|} \|s_i - \bar{s}_i\|_1, \quad \text{where } \bar{s}_i = \frac{1}{3} (s_i^x + s_i^y + s_i^z). \quad (13)$$

- Why do we need “isotropic regularization”?
 - **Observation:** isotropic Gaussians behave better than anisotropic.

Isotropic Regularization

$$\mathcal{L}_{iso} = \sum_{i=1}^{|\mathcal{G}|} \|s_i - \bar{s}_i\|_1, \quad \text{where } \bar{s}_i = \frac{1}{3} (s_i^x + s_i^y + s_i^z). \quad (13)$$

- Why do we need “isotropic regularization”?
 - Observation: isotropic Gaussians behave better than anisotropic.
 - **Analysis:** no constraints on the elongation along the viewing ray direction, **even with depth**.

Isotropic Regularization

$$\mathcal{L}_{iso} = \sum_{i=1}^{|\mathcal{G}|} \|s_i - \bar{s}_i\|_1, \quad \text{where } \bar{s}_i = \frac{1}{3} (s_i^x + s_i^y + s_i^z). \quad (13)$$

The Overall Optimization for Mapping

$$\operatorname{argmin}_{\mathcal{G}, \{\mathbf{T}_{cw}(\mathcal{F}_k) | \mathcal{F}_k \in \mathcal{W}^+\}} \sum_{\mathcal{F}_k}^{\mathcal{W}^+} \mathcal{L}_{pho}(\mathcal{F}_k) + \lambda_{iso} \mathcal{L}_{iso} \quad (14)$$

Appendix

- [1] N. Keetha, J. Karhade, K. M. Jatavallabhula, et al., *SplaTAM: Splat, track & map 3d gaussians for dense RGB-d SLAM*, Apr. 16, 2024. arXiv: [2312.02126\[cs\]](https://arxiv.org/abs/2312.02126). [Online]. Available: <http://arxiv.org/abs/2312.02126> (visited on 05/20/2024) (cit. on p. iv).
- [2] C. Yan, D. Qu, D. Wang, et al., *GS-SLAM: Dense visual SLAM with 3d gaussian splatting*, Nov. 21, 2023. arXiv: [2311.11700\[cs\]](https://arxiv.org/abs/2311.11700). [Online]. Available: <http://arxiv.org/abs/2311.11700> (visited on 12/26/2023) (cit. on p. iv).
- [3] V. Yugay, Y. Li, T. Gevers, and M. R. Oswald, *Gaussian-SLAM: Photo-realistic dense SLAM with gaussian splatting*, Mar. 22, 2024. arXiv: [2312.10070\[cs\]](https://arxiv.org/abs/2312.10070). [Online]. Available: <http://arxiv.org/abs/2312.10070> (visited on 03/27/2024) (cit. on p. iv).
- [4] H. Matsuki, R. Murai, P. H. J. Kelly, and A. J. Davison, *Gaussian splatting SLAM*, Apr. 14, 2024. arXiv: [2312.06741\[cs\]](https://arxiv.org/abs/2312.06741). [Online]. Available: <http://arxiv.org/abs/2312.06741> (visited on 05/20/2024) (cit. on p. iv).
- [5] J. Engel, V. Koltun, and D. Cremers, “Direct sparse odometry,” in *arXiv:1607.02565*, Jul. 2016 (cit. on pp. xvii, xviii).