COLMAP-Free 3D Gaussian Splatting

Reviewer: Veronika Treumova

Archeologist: Veronika Treumova

Hacker: Mateus Barbosa

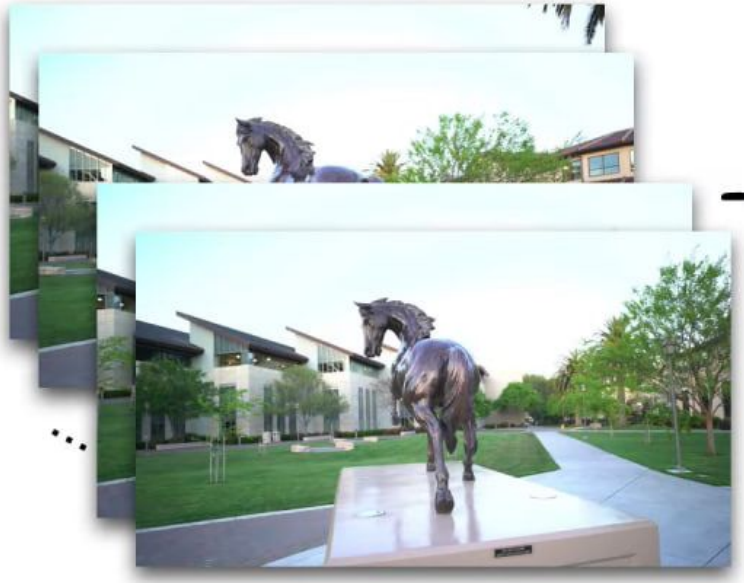
PhD Student: Vitor Pereira Matias

Reviewer

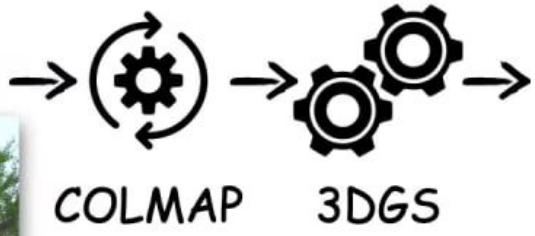
Veronika Treumova

3D Gaussian Splatting with COLMAP Preprocessing

Input



Video: sequência de imagens



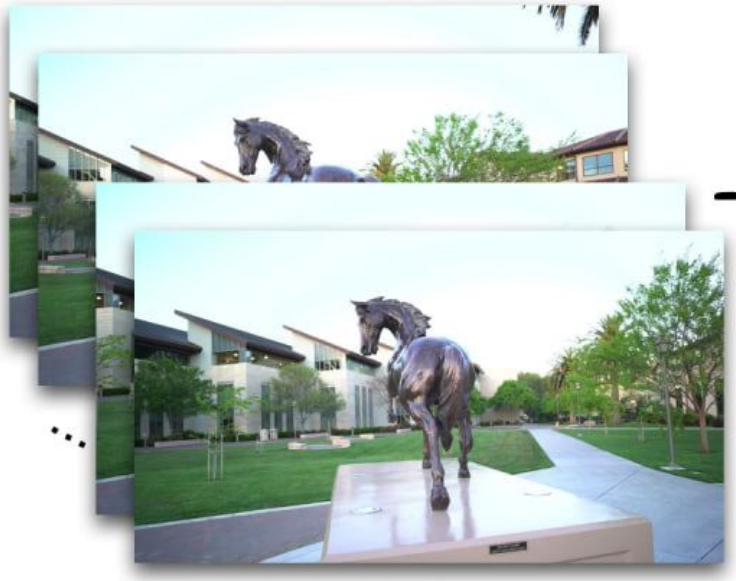
Novel View Synthesis

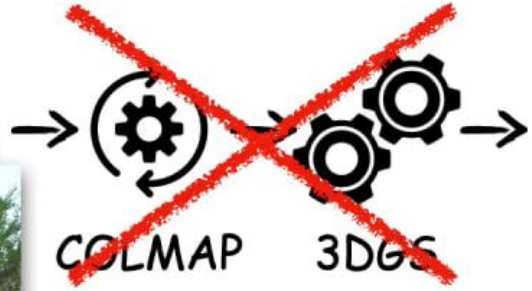
Problem

- initialization step for training NeRF is to first **prepare the camera poses** for each input image
- achieved by running the Structure-from-Motion (SfM) library **COLMAP**
- this pre-processing is
  - time-consuming
  - can fail due to its sensitivity to feature extraction errors and difficulties in handling textureless or repetitive regions

3D Gaussian Splatting without COLMAP Preprocessing

Input

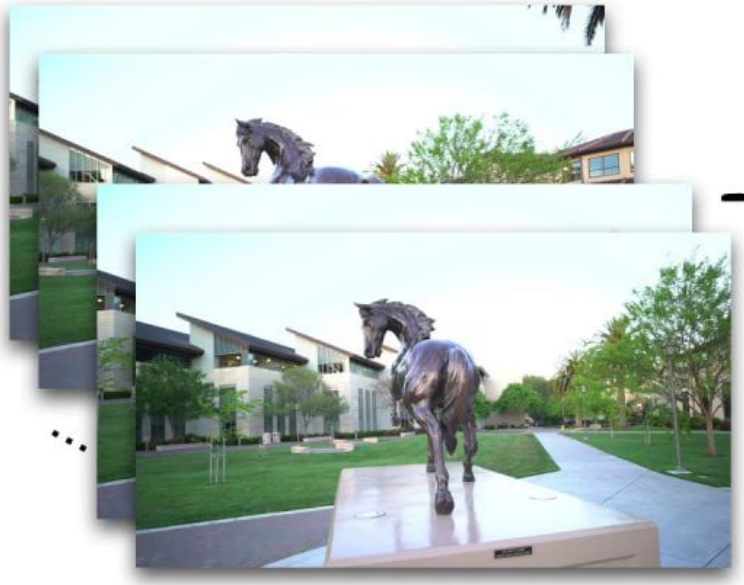




Novel View Synthesis

3D Gaussian Splatting without COLMAP Preprocessing

Input



→ CF-3DGS →

a sequence of unposed images along with camera intrinsic

Recover camera poses and reconstruct the photo-realistic scene

Solution

COLMAP-Free 3D Gaussian Splatting
(CF-3DGS)

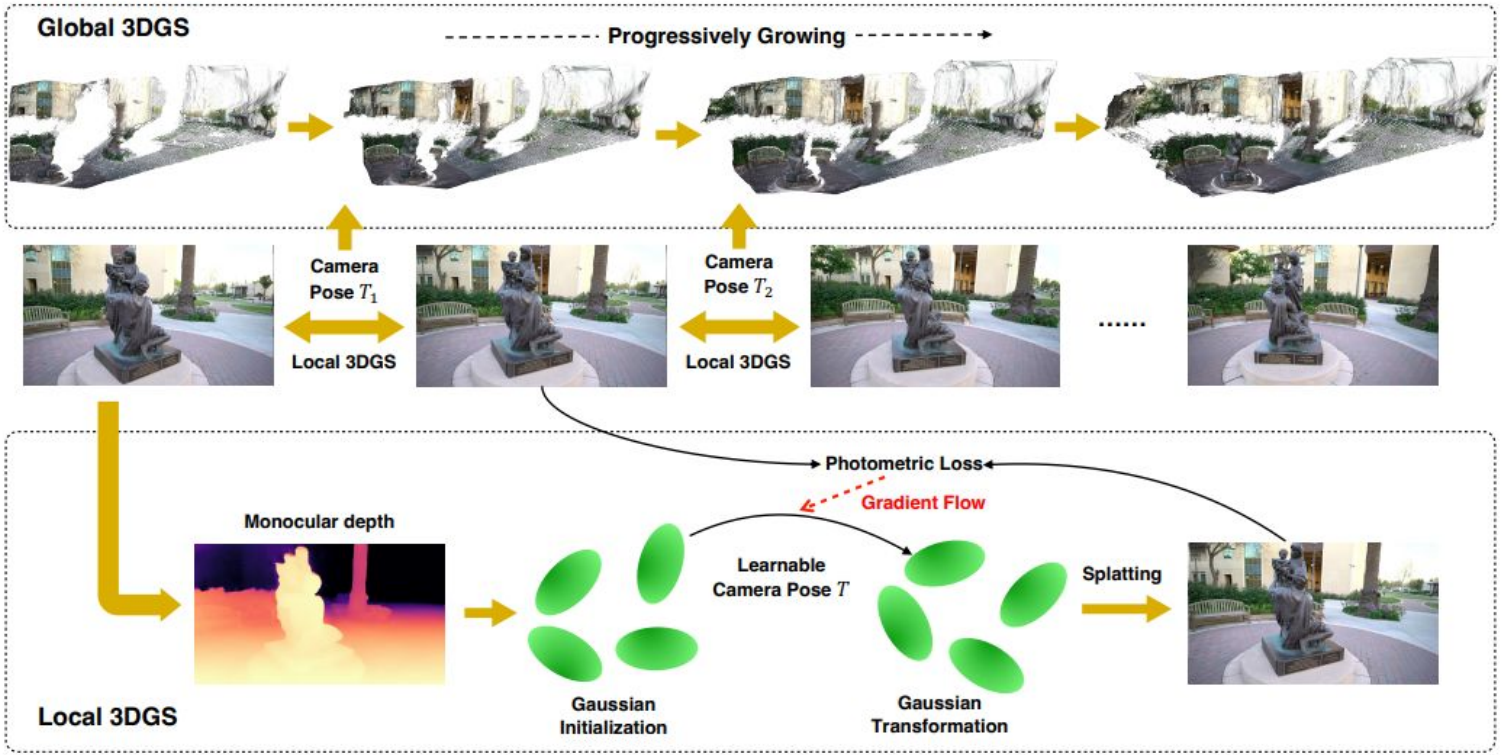
Authors propose to build the 3D Gaussians of the scene in a continuous manner, “growing” one frame at a time as the camera moves

key ingredients:

temporal continuity from video

explicit point cloud representation

Method


Global 3DGS
Progressively Growing
Camera Pose T_1
Local 3DGS
Camera Pose T_2
Local 3DGS
.....
Photometric Loss
Gradient Flow
Monocular depth
Learnable Camera Pose T
Splatting
Local 3DGS
Gaussian Initialization
Gaussian Transformation

Initialization from a single view

Given a frame I_t at timestep t:

1) Utilize an off-the-shelf monocular depth network, to generate the monocular depth
2) Initialize 3DGS with points lifted from monocular depth, leveraging camera intrinsic and identity camera pose
3) Learn a set of 3D Gaussian G_t with all attributes to minimize the photometric loss between the rendered image and the current frame

$$G_t^* = \arg \min_{c_t, r_t, s_t, \alpha_t} \mathcal{L}_{rgb}(\mathcal{R}(G_t), I_t),$$

$$\mathcal{L}_{rgb} = (1 - \lambda)\mathcal{L}_1 + \lambda \mathcal{L}_{\text{D-SSIM}}$$

Pose Estimation by 3D Gaussian Transformation

we transform the pre-trained 3D Gaussian $G_t{}^*$ by a learnable SE-3 affine transformation T_t into frame $t+1$, denoted as $G_{t+1} = T_t \odot G_t$. The transformation T_t is optimized by minimizing the photometric loss between the rendered image and the next frame I_{t+1}

$$T_t{}^* = \arg\min_{T_t} \mathcal{L}_{rgb}(\mathcal{R}(T_t \odot G_t), I_{t+1}),$$

Pose Estimation by 3D Gaussian Transformation

This optimization is not difficult as the **explicit point cloud representation** allows to directly apply an affine transformation on it which cannot be achieved with NeRFs, and the two frames are close (**temporal continuity**) which makes the transformation relatively small.

Global 3DGS with Progressively Growing

1) Starting from the t-th frame I_t, we first initialize a set of 3D Gaussian points with the camera pose set as orthogonal
2) Utilizing the local 3DGS, we estimate the relative camera pose between frames I_t and I_{t+1}
3) Following this, the global 3DGS updates the set of 3D Gaussian points, along with all attributes, over N iterations, using the estimated relative pose and the two observed frames as inputs
4) The next frame I_{t+2} becomes available, this process is repeated: we estimate the relative pose between I_t and I_{t+2}, and subsequently infer the relative pose between I_t and I_{t+2}

Global 3DGS with Progressively Growing

- To update the global 3DGS to cover the new view, they densify the Gassians that are ”under-reconstruction” as new frames arrive
- They determine the candidates for densification by the **average magnitude of viewspace position gradients**
- In addition, instead of stopping the densification in the middle of the training stage, they keep growing the 3D Gaussian points until the end of the input sequence

Strengths

- method performs significantly better than previous approaches without pre-computed camera poses
- effectiveness and robustness of approach on challenging scenes like 360◦ videos
- thanks to the advantages of Gaussian splatting, approach achieves rapid training and inference speeds.

Weaknesses

- optimizes camera pose and 3DGS jointly in a sequential manner, thereby restricting its application primarily to video streams or **ordered** image collections

Paper aprovado😇😇😇

Archeologist

Veronika Treumova

Foundations

- Gaussian Splatting
- …
-

Posterior research

Concurrent work: Look Gauss, No Pose: Novel View Synthesis using Gaussian Splatting without Accurate Pose Initialization (11/10/2024)

While they experiment with a similar initialization scheme for the trajectory, they use a different process to build a consistent Gaussian representation. Their method **does not require an iterative estimation of poses** and therefore has **lower runtime**, especially for long video sequences. Additionally, their method can be adjusted to work on **unordered** image collections.

Fig. 1. Our proposed approach enables fast, Gaussian Splatting based 3D reconstruction and photo-realistic novel-view synthesis while simultaneously estimating and refining the camera poses, as visualized on this reconstruction of the *horns* scene from *LLFF* [10]. This reconstruction was performed without camera pose information.

Comparison results

COMPARISON ON TANKS AND TEMPLES [15] AGAINST STATE-OF-THE-ART METHODS FOR JOINT 3D RECONSTRUCTION AND POSE ESTIMATION. * DENOTES CONCURRENT WORK.

Method	Novel View Synthesis			Pose Estimation		
	PSNR↑	SSIM↑	LPIPS↓	RPE_t↓	RPE_r↓	ATE↓
Nerf-- [8]	22.50	0.59	0.54	1.735	0.477	0.123
SCNerf [33]	23.76	0.65	0.48	1.890	0.489	0.129
BARF [7]	23.42	0.61	0.54	1.046	0.441	0.078
Nope-Nerf [9]	26.34	0.74	0.39	0.080	0.038	0.006
CF-3DGS* [6]	31.28	0.93	0.09	0.041	0.069	0.004
Ours	31.24	0.92	0.12	0.075	0.069	0.009

similar visual fidelity as CF-3DGS, but about 4× faster

Hacker

Mateus Barbosa

Depth

Depth is supposed to be obtained from:

- MiDaS for the Tanks and Temples dataset
- ZoeDepth for the CO3D dataset
- DepthAnything for any custom dataset

However, in practice it always uses MiDaS.

Intrinsics

While the paper relies on given intrinsics, it allows for custom datasets without any given intrinsics.
Then it assumes a fixed intrinsic parameter:

- FoV of all scenes is set to 79º
- make the principle points to the image center.

```
else:
    # use some hardcoded values
    fov = 79.0
    FoVx = fov * math.pi / 180
    intr_mat = np.eye(3)
    intr_mat[0, 0] = fov2focal(FoVx, width)
    intr_mat[1, 1] = fov2focal(FoVx, width)
    intr_mat[0, 2] = width / 2
    intr_mat[1, 2] = height / 2
```

Testing with their dataset and parameters

Ground Truth

CF-3DGS

Ground Truth

CF-3DGS

Ground Truth

CF-3DGS

CF-3DGS (ran by them)

CF-3DGS(ran by me)

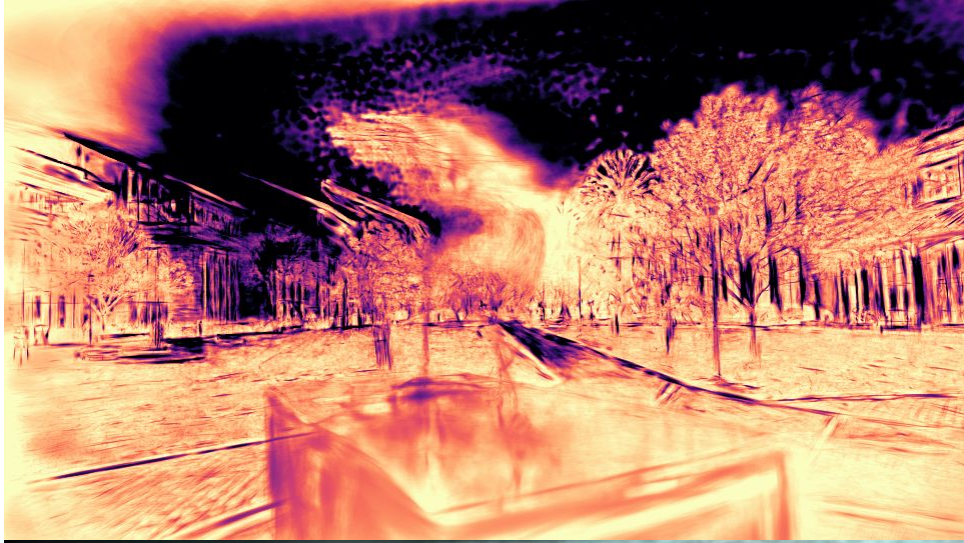

Sports Hall of Fame

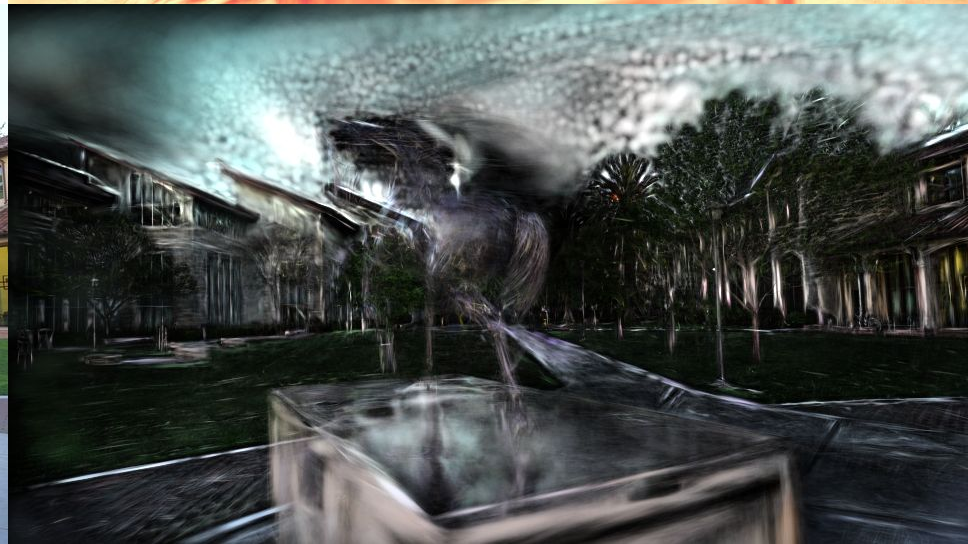

Sports Hall of Fame


Frank's Tannery

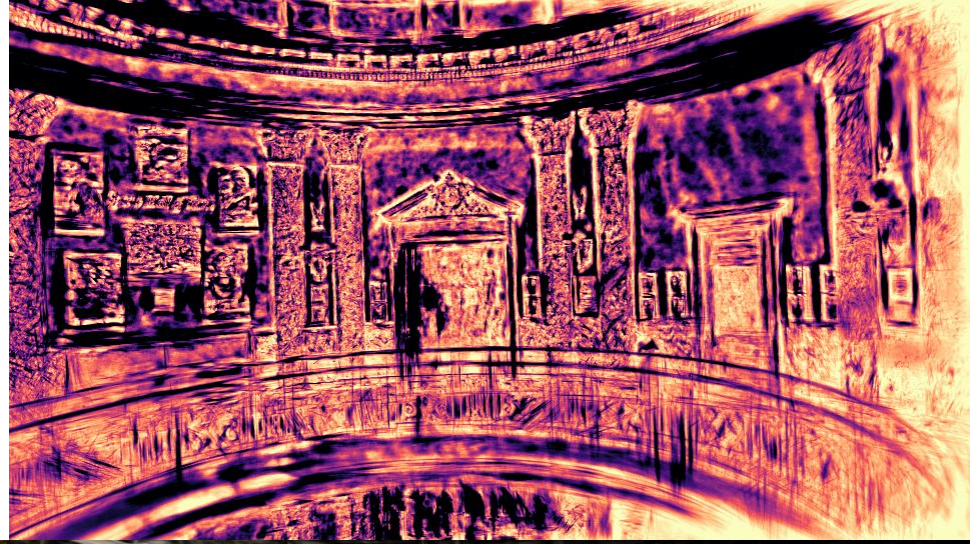
Some issues

Some issues





Some issues



	Scenes	Type	Seq. length	Frame rate	Max. rotation (deg)
Tanks and Temples	Church	indoor	400	30	37.3
	Barn	outdoor	150	10	47.5
	Museum	indoor	100	10	76.2
	Family	outdoor	200	30	35.4
	Horse	outdoor	120	20	39.0
	Ballroom	indoor	150	20	30.3
	Francis	outdoor	150	10	47.5
	Ignatius	outdoor	120	20	26.0
CO3D-V2	34_1403_4393	indoor	202	30	180.0
	106_12648_23157	outdoor	202	30	180.0
	110_13051_23361	indoor	202	30	71.6
	219_23121_48537	indoor	202	30	180.0
	245_26182_52130	indoor	202	30	180.0
	247_26441_50907	indoor	202	30	180.0
	407_54965_106262	indoor	202	30	180.0
	415_57112_110099	outdoor	202	30	180.0
	415_57121_110109	outdoor	202	30	180.0
	429_60388_117059	outdoor	202	30	180.0

Table 8. **Details of selected sequences.** We downsample several videos to a lower frame rate. FPS denotes frame per second. *Max rotation* denotes the maximum relative rotation angle between any two frames in a sequence. Our method can handle dramatic camera motion (large maximum rotation angle).

Intrinsics ablation study

Method	PSNR	SSIM	LPIPS	RPE_t	RPE_r	ATE
Heuristic Intrinsic	30.90	0.92	0.09	0.044	0.072	0.004
G.T. Intrinsic	**31.28**	**0.93**	**0.09**	**0.041**	**0.069**	**0.004**

Table 11. Ablation study of camera intrinsic on Tanks and Temples.

Testing in the custom dataset mode, i.e., without previously-estimated intrinsics

Ground Truth

CF-3DGS

Testing in the custom dataset mode, i.e., without previously-estimated intrinsics

CF-3DGS (tanks-optimized mode)

CF-3DGS (custom mode without intrinsics)

com intrínsecos / sem intrínsecos

CF-3DGS (custom mode with intrinsics)

CF-3DGS (custom mode without intrinsics)

PhD Student

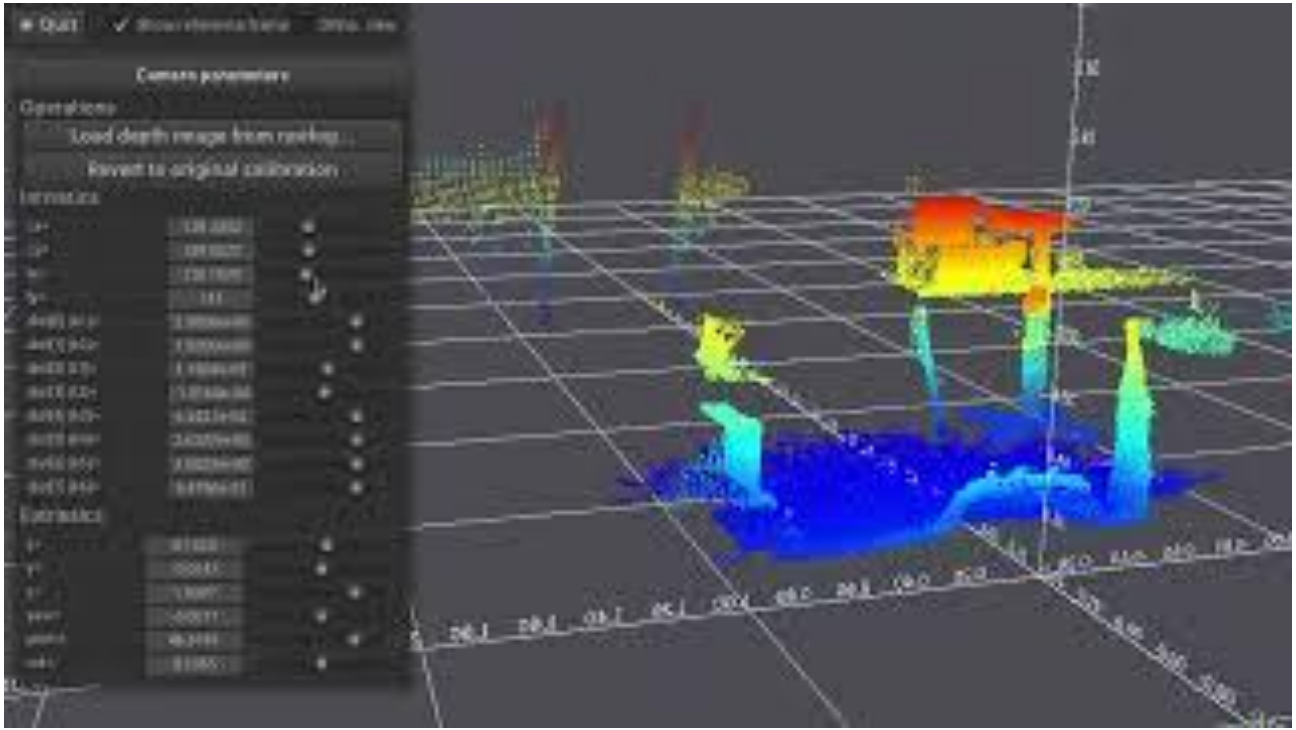
Vitor Pereira Matias

RGBsplat

- No Pose
- No Intrinsics

- Only RGB images as input
- Similar to flowmap

How colmap uses K?



How flowmap infers K?

- Tries n different values of K
- Choses the one that returns the smallest loss

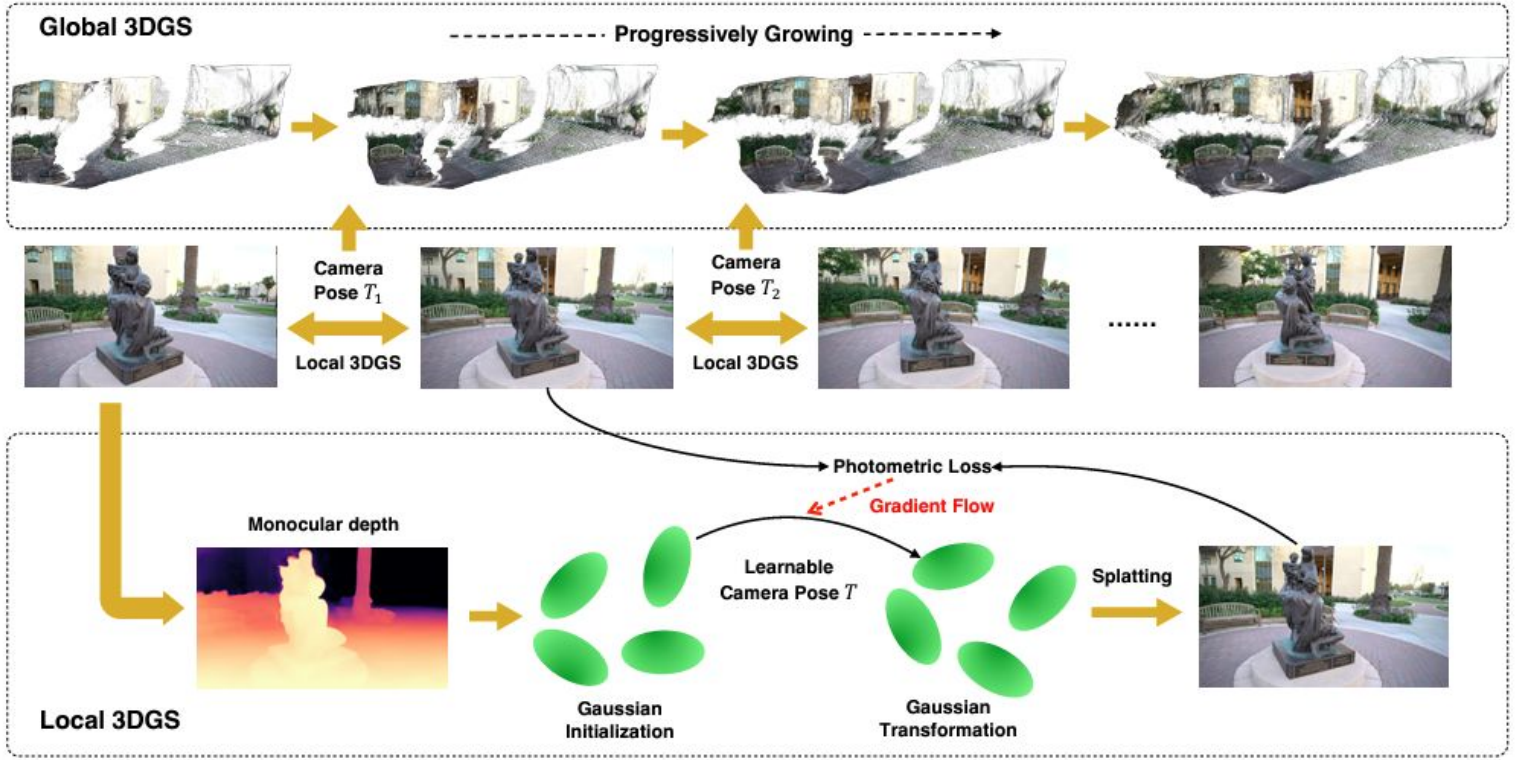
$$\begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}$$

$f_x = f_y$

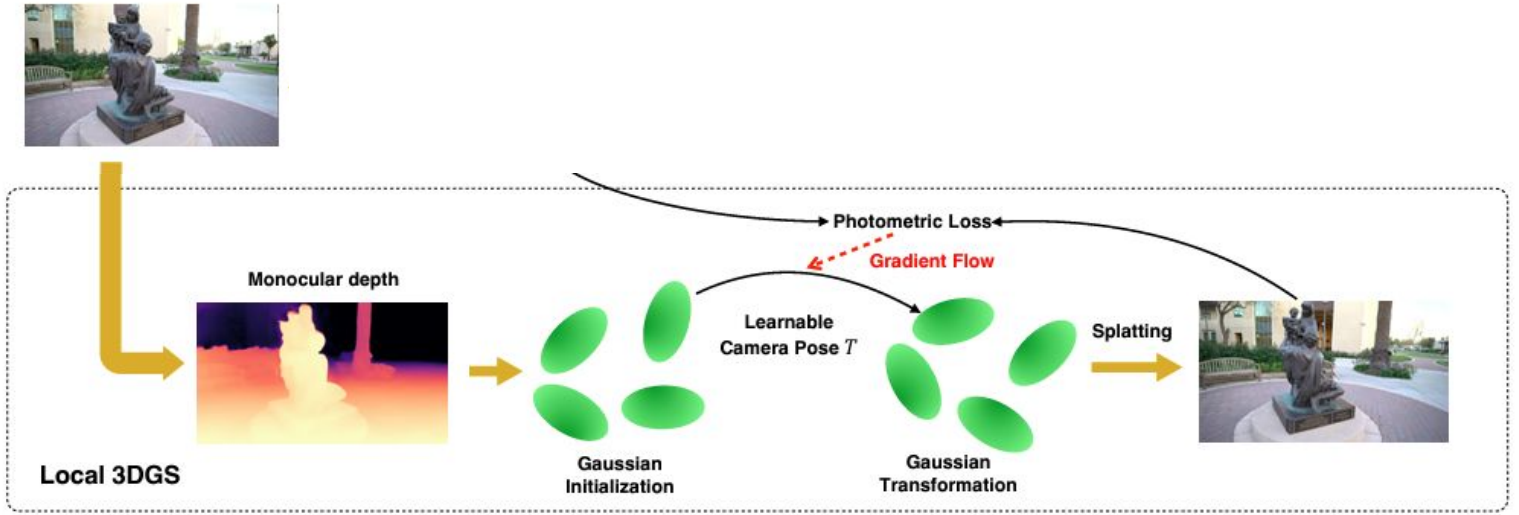
c_x, c_y given by image center = 0.5

- All images have the same K

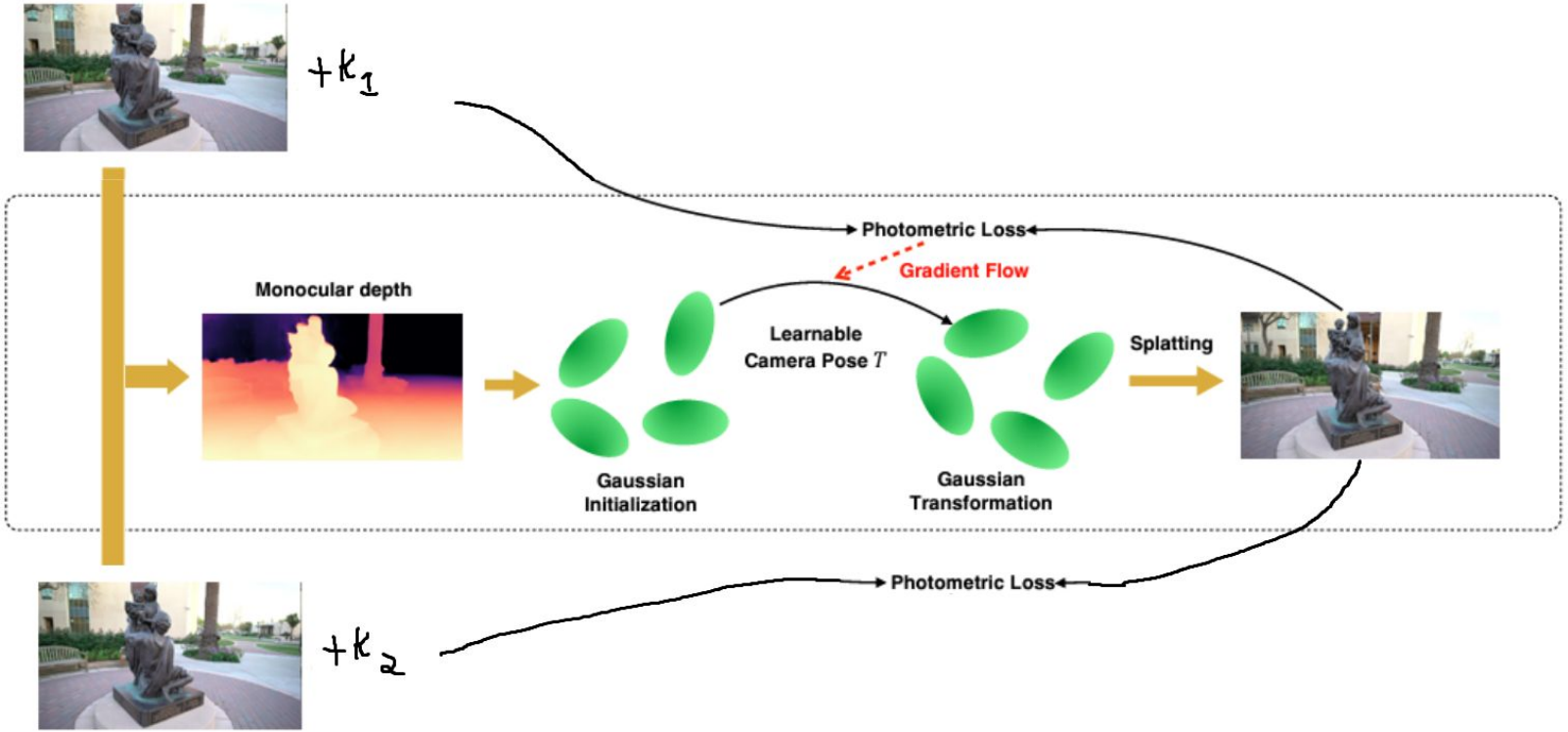
Colmap


Global 3DGS
Progressively Growing
Camera Pose T_1
Local 3DGS
Camera Pose T_2
Local 3DGS
.....
Photometric Loss
Gradient Flow
Monocular depth
Learnable Camera Pose T
Splatting
Local 3DGS
Gaussian Initialization
Gaussian Transformation

Colmap core


Photometric Loss
Gradient Flow
Monocular depth
Learnable
Camera Pose T
Splatting
Local 3DGS
Gaussian
Initialization
Gaussian
Transformation

RGBsplat


+K1
Photometric Loss
Gradient Flow
Monocular depth
Learnable
Camera Pose T
Splatting
Gaussian
Initialization
Gaussian
Transformation
Photometric Loss
+K2

RGBsplat

- Which loss to use?
  - Flowcam/map pose loss

$$\mathcal{L}_{\text{pose}} = \frac{1}{N-1} \sum_{t=1}^{N-1} \left\| \mathbf{V}_t - (\pi(\mathbf{P}_t^{-1} \cdot \mathbf{P}_{t+1} \cdot \mathbf{X}_{t+1}) - \mathbf{uv}) \right\|_2^2,$$

  - Colmap Photometric loss

$$\mathcal{L}_{rgb} = (1-\lambda)\mathcal{L}_1 + \lambda \mathcal{L}_{\text{D-SSIM}}$$

- Then, as done by flowmap

$$\mathbf{K} = \sum_k w_k \mathbf{K}_k \qquad\qquad w_k = \frac{\exp(-\mathcal{L}_k)}{\sum_l \exp(-\mathcal{L}_l)}$$

RGBsplat in short

- No pose, no intrinsics
- Pose are estimated using COLMAP-Free 3DGS method
- Intrinsics are estimated using FlowMap (or flowcam) method
    - For each intrinsic K in a given range
        - For each image, reconstruct it using 3DGS
            - Calculate loss
    - Compare all losses using softmax as done by flowmap
    - Chose K