Dhruv Maheshwari 9001607595

CSCI 677 – Advanced Computer Vision

Assignment 3

This program is used to reconstruct 3D from 2D images.

COLMAP:

Feature extraction using SIFT

Images	Techniques			
	Simple Pinhole and Exhaustive (0.148 mins)	Simple Pinhole and Sequential (0.095 mins)	Simple Pinhole and Exhaustive (0.165 mins)	Simple Pinhole and Sequential (0.167 mins)
Img 1	12912	12912	12912	12912
Img 2	8612	8612	8612	8612
Img 3	8982	8982	8982	8982
Img 4	9532	9532	9532	9532
Img 5	10306	10306	10306	10306
Img 6	11756	11756	11756	11756
Img 7	11849	11849	11849	11849
Img 8	11145	11145	11145	11145
Img 9	10414	10414	10414	10414
Img 10	10849	10849	10849	10849
Img 11	-	-	11036	11036
Img 12	-	-	11257	11257
Img 13	-	-	10868	10868
Img 14	-	-	12434	12434
Img 15	-	-	8266	8266
Img 16	-	-	11440	11440
Img 17	-	-	11217	11217
Img 18	-	-	11504	11504
Img 19	-	-	12909	12909
Img 20	-	-	12005	12005

Feature Matching

Technique	Parameters
Simple Pinhole and Exhaustive (0.160 mins)	Block Size - 50
Simple Pinhole and Sequential (0.160 mins)	Overlap - 10
Simple Pinhole and Exhaustive (0.706 mins)	Block Size - 50
Simple Pinhole and Sequential (0.525 mins)	Overlap - 10

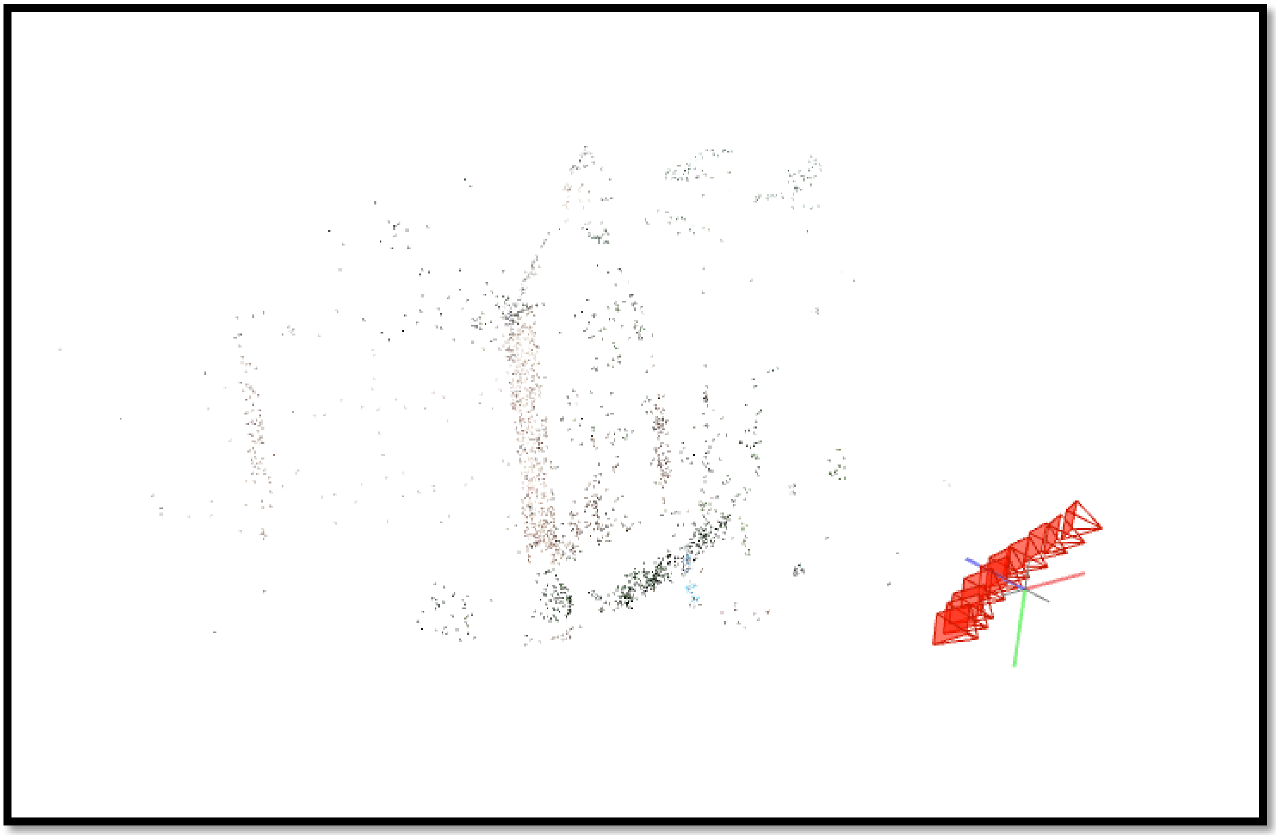

Figure 1: Simple Pinhole – Exhaustive (10 images)

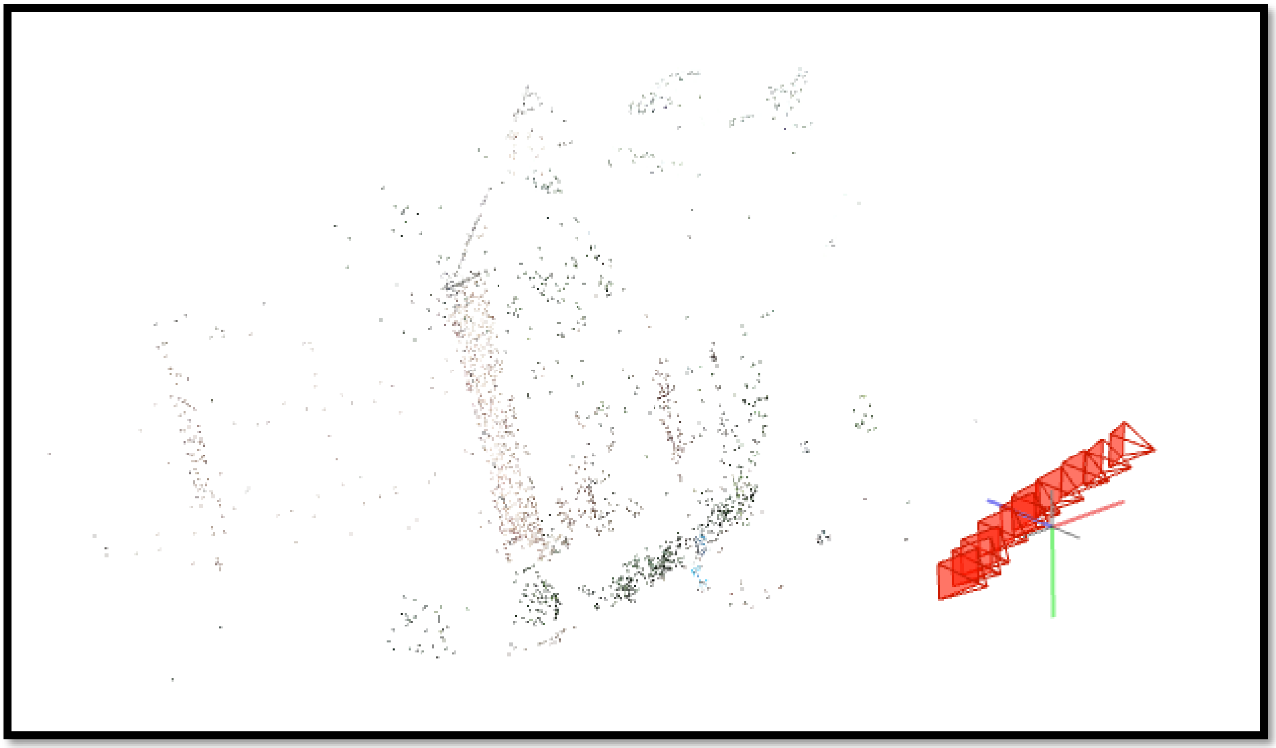

Figure 2: Simple Pinhole - Sequential (10 images)

Figure 3: Simple Pinhole - Exhaustive (20 images)

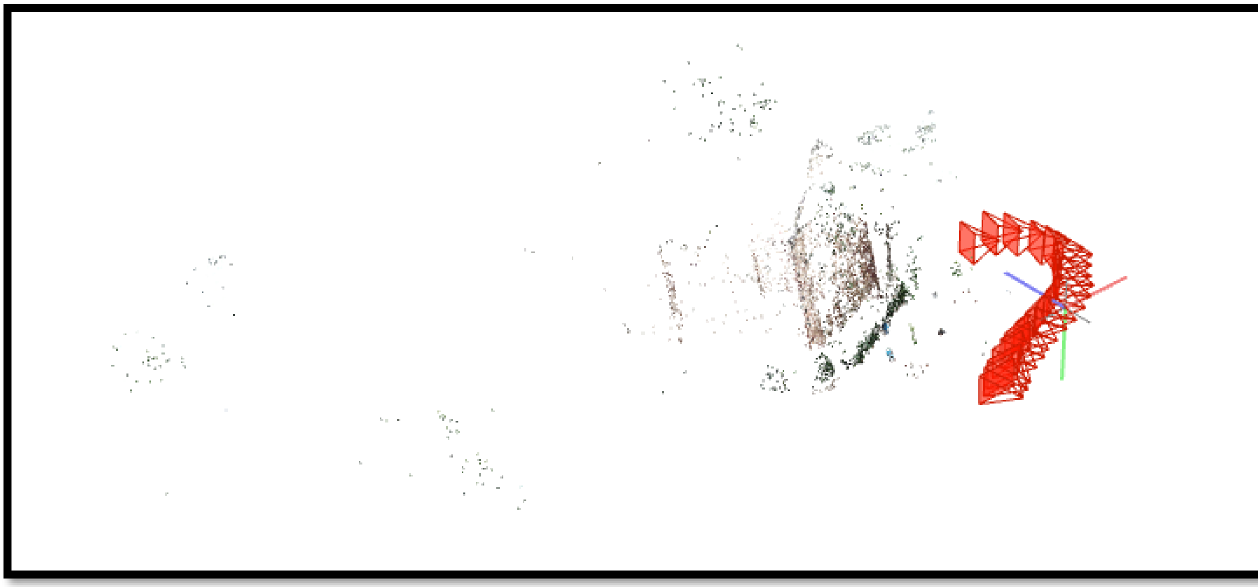

Figure 4: Simple Pinhole - Exhaustive (20 images)

Reconstruction and Reprojection error

Techniques	Parameters							
	Cameras	Images	Registered Images	Points	Observations	Mean Track length	Mean observations per image	Mean reprojection error
Simple Pinhole and Exhaustive	1	10	10	2704	10306	3.81139	1030.6	0.567371
Simple Pinhole and Sequential (0.042 mins)	1	10	10	2701	10299	3.81303	1029.9	0.566575
Simple Pinhole and Exhaustive (0.107 mins)	1	20	20	4730	18543	3.9203	927.15	0.59107
Simple Pinhole	1	20	20	4708	18468	3.92268	923.4	0.588509

and Sequential (0.115 mins)								

Higher the number of images, points and observations, better it is as it provides more detailed construction by providing a robust and accurate mapping between 3D model and a 2D model. Points observed in Exhaustive model are slightly greater than in Sequential model.

Mean track length - The average number of images in which a 3D point (track) is observed. A track refers to a 3D point's projection onto multiple images. Higher is better as it increases accuracy and position, thus leading to robust model. A sequential model here is better.

Mean Observations per image - The average number of 3D points observed in each image. This metric shows how many 3D points are visible in each image, which is a sign of the richness of data available from each viewpoint. Higher is preferred as each image contributes more towards building a model. Exhaustive model gives more.

Mean Reprojection error - The average distance between the 2D image points and their corresponding projected 3D points when the 3D model is projected back into the image plane. Lower is better as it shows 3D points are aligned closely as in corresponding 2D image. We can see that Sequential model gives a better result.

Hence, overall a Sequential model gives a better result.

GAUSSIAN SPLATTING

Code can be found here:
https://colab.research.google.com/drive/1V7cFc2cD1IfoHzRK61PBx85F2auNIRu1?usp=sharing

Code:

Made some changes in dataset_readers.py

Line 149-150

```
        train_cam_infos = [c for idx, c in enumerate(cam_infos) if (idx + 1) % 5 != 0 or idx == len(cam_infos) - 1]
        test_cam_infos = [c for idx, c in enumerate(cam_infos) if (idx + 1) % 5 == 0 and idx != len(cam_infos) - 1]
```

This was done to remove image #5 from dataset containing only 10 photos during training but to include all during testing. Similarly, #5, #10, #15 were removed for the dataset containing 20 photos.

Main code:

```
%cd /content
!git clone --recursive https://github.com/camenduru/gaussian-splatting
```

```
!pip install -q plyfile

%cd /content/gaussian-splatting
!pip install -q /content/gaussian-splatting/submodules/diff-gaussian-rasterization
!pip install -q /content/gaussian-splatting/submodules/simple-knn
from google.colab import drive
drive.mount('/content/drive')
!python train.py -s '/content/drive/MyDrive/workspace_10' -eval
!python render.py -m /content/gaussian-splatting/output/85ebcdd6-8
!python metrics.py -m /content/gaussian-splatting/output/85ebcdd6-8/
```

The output folder generated was rendered on SIBR Gaussian Viewer.

Dataset with 20 images

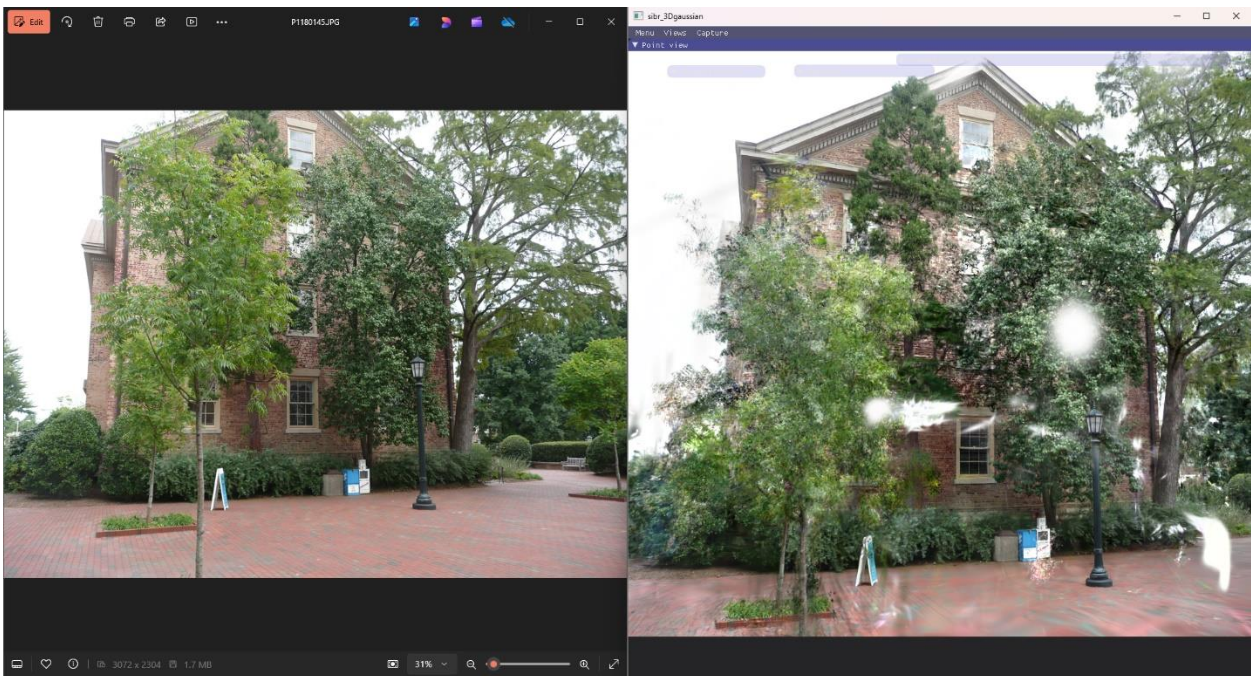

Figure 5: Comparing #5

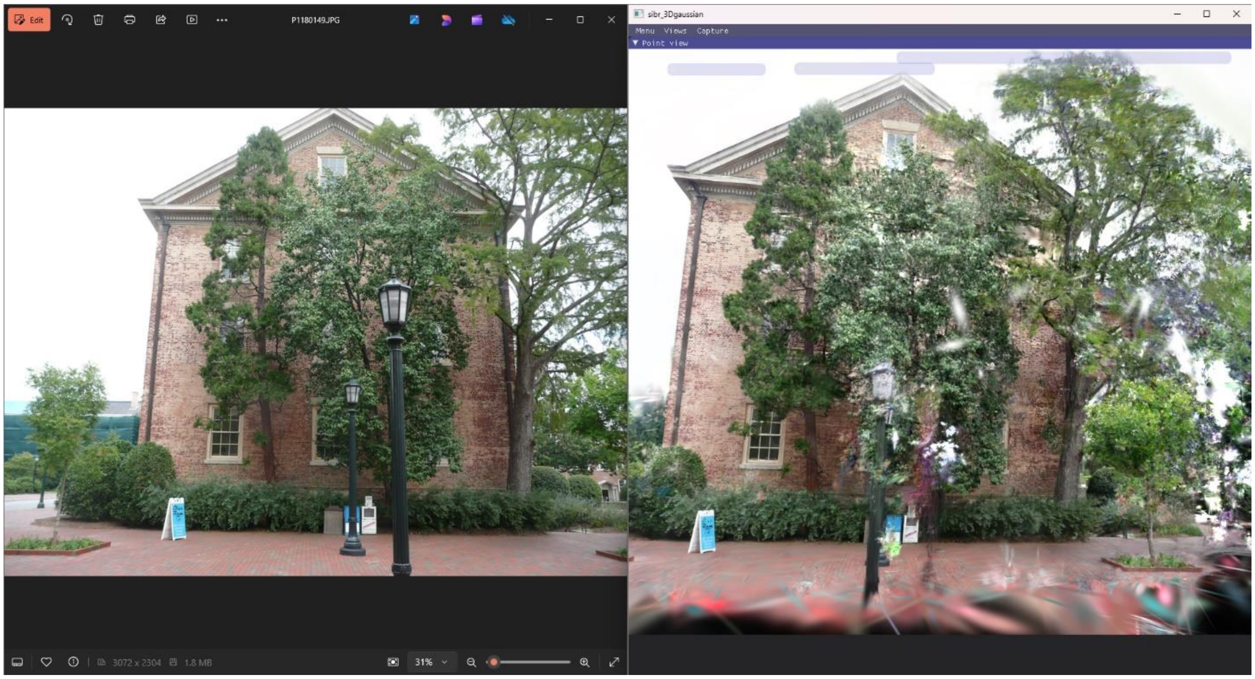

Figure 6: Comparing #10

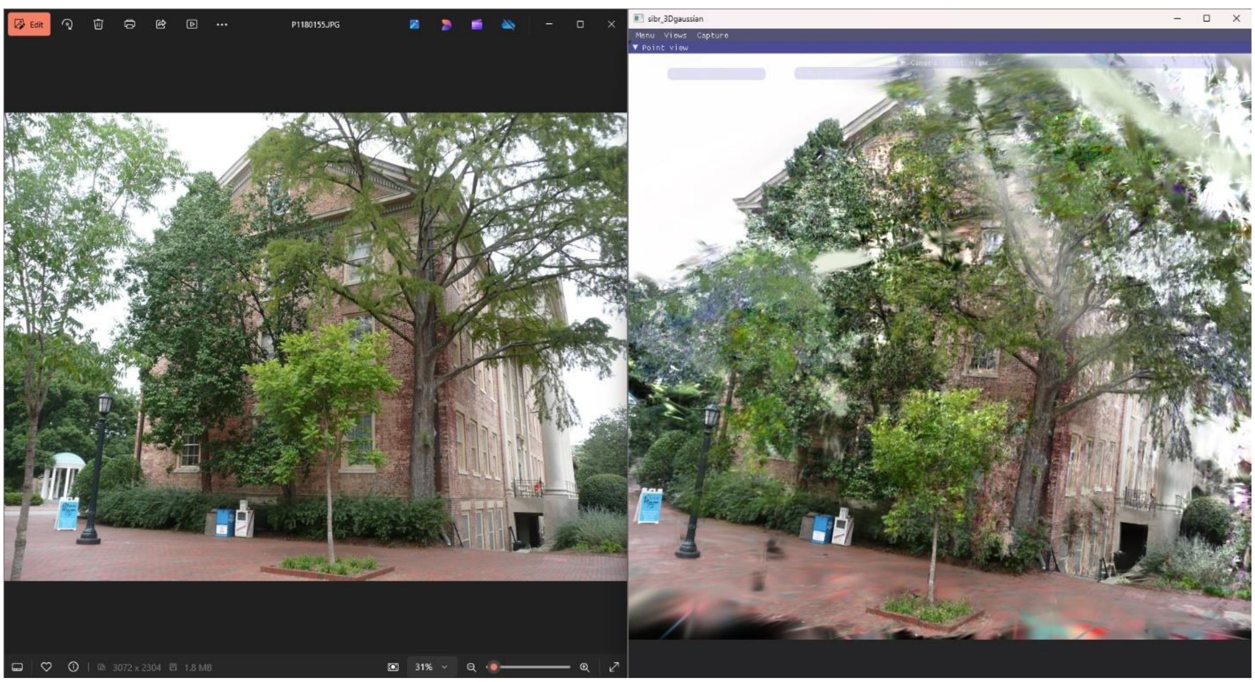

Figure 7: Comparing #15

Dataset with 10 images

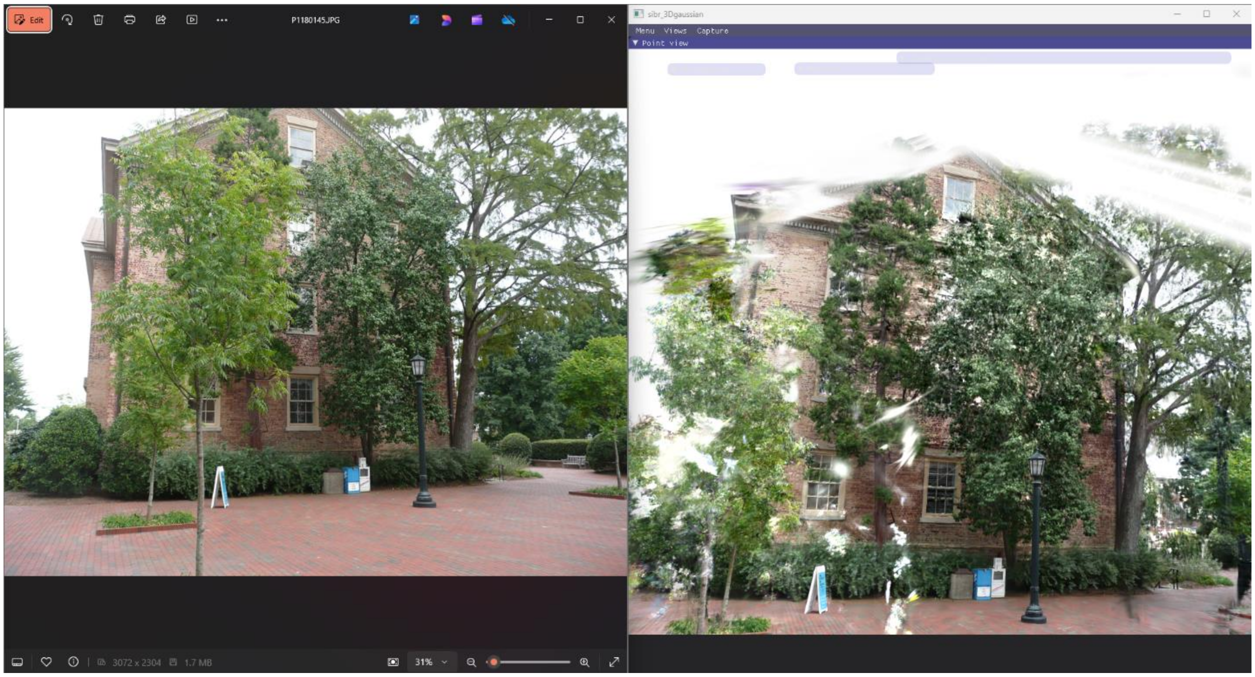

Figure 8: Comparing #5

Following screenshots show a comparison, left → 10 images, right → 20 images (without excluding images during training)

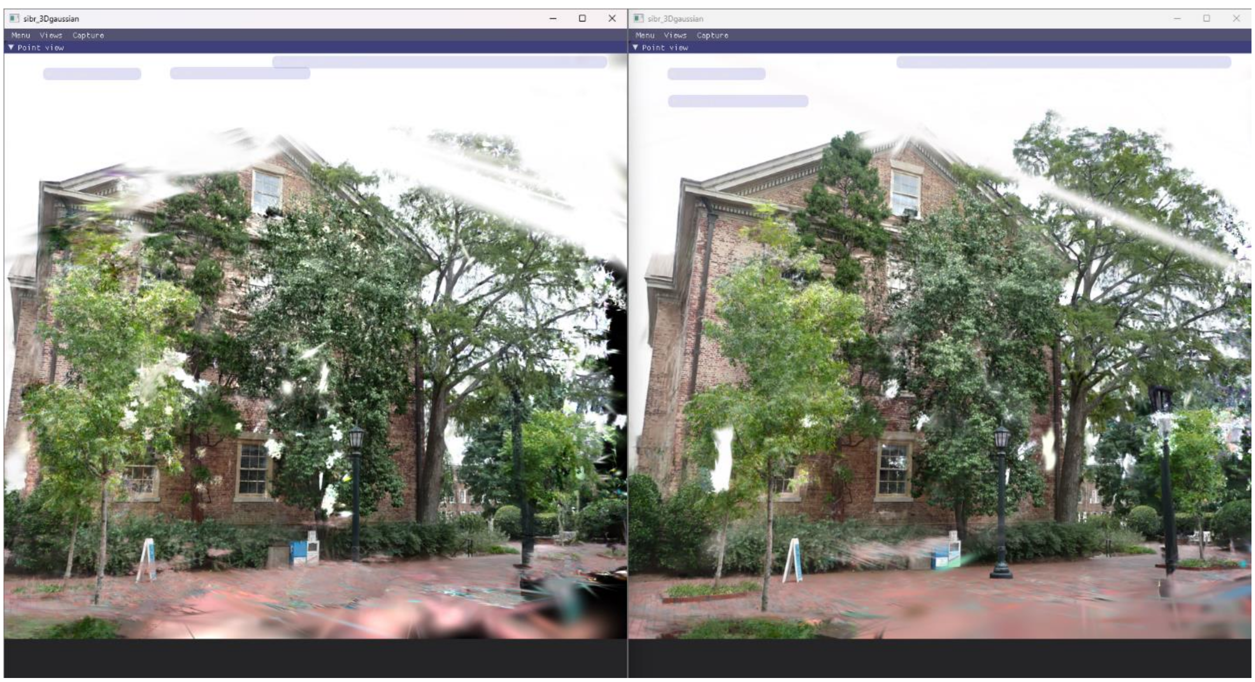

Figure 9: Comparing two models - half dataset (left), full dataset (right)

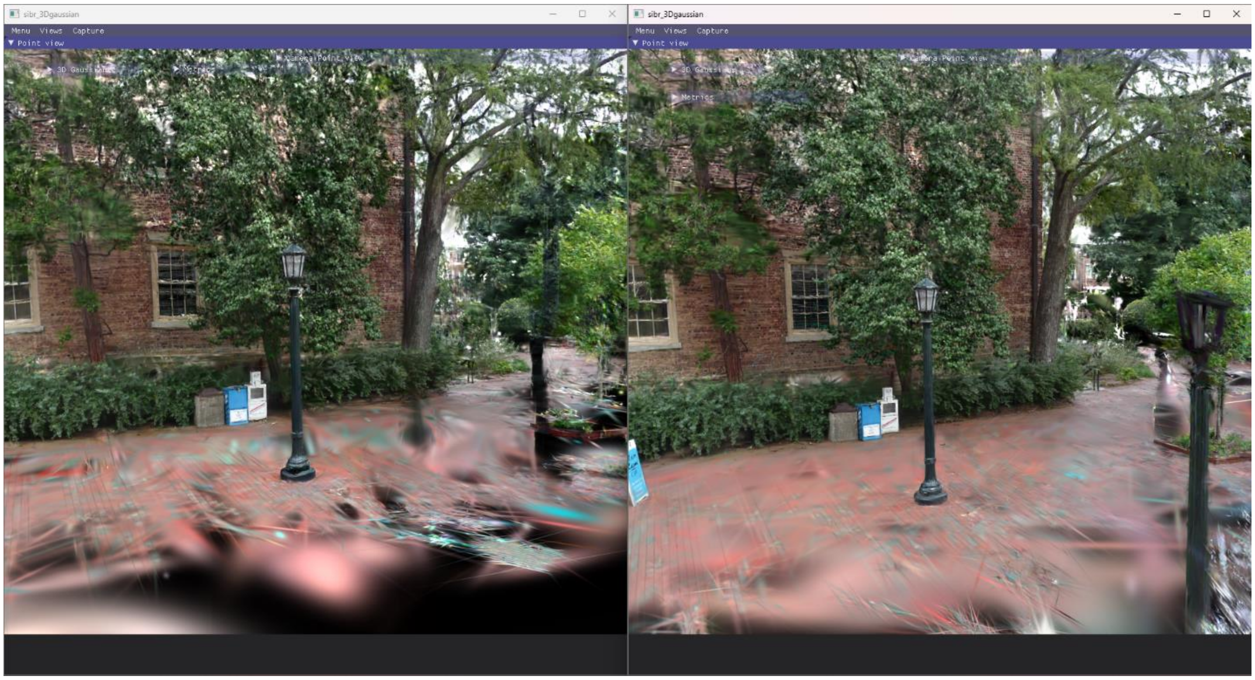

Figure 10: Comparing two models - half dataset (left), full dataset (right)

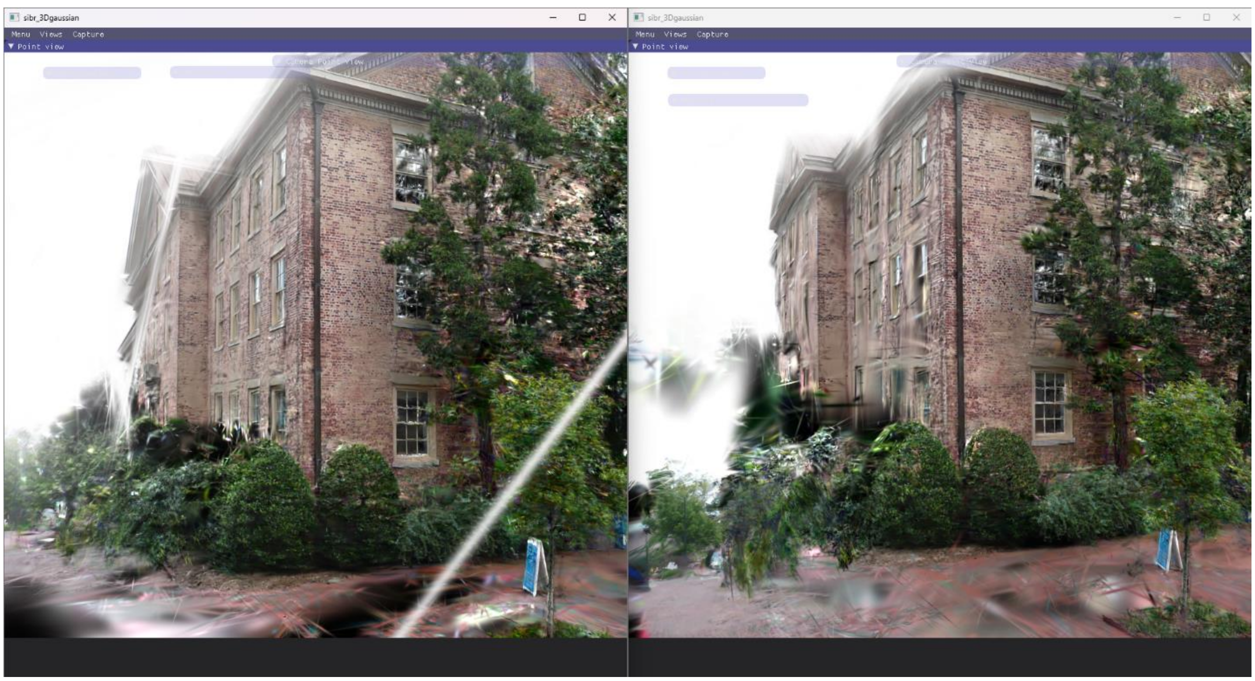

Figure 11: Comparing two models - half dataset (left), full dataset (right)

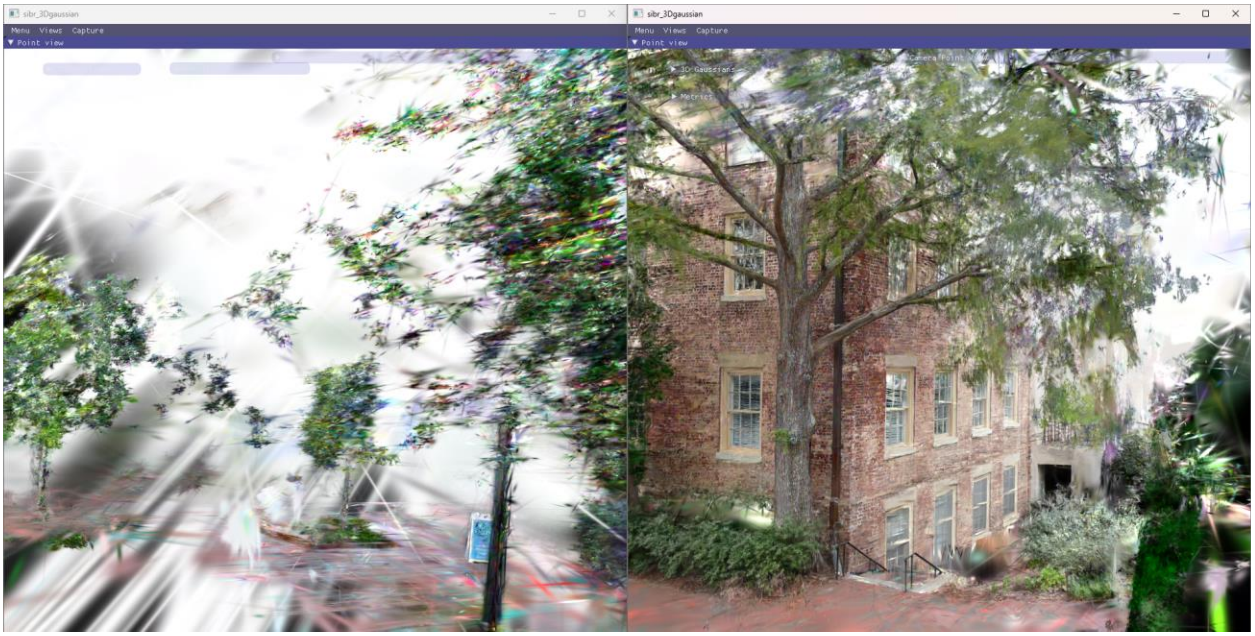

Figure 12: Comparing two models - half dataset (left), full dataset (right)

Metric results

Database with 20 images

SSIM : 0.4743837
PSNR : 15.4782104
LPIPS: 0.3955861

Database with 10 images

SSIM : 0.4283298
PSNR : 15.1196899
LPIPS: 0.4236086

Structural Similarity Index Measure – It refers to the perceptual quality of the image and works by comparing the structure in the image. Higher SSIM indicates that reconstructed model has better structural information and looks closer to original image.

Peak Signal-to-Noise Ratio - It compares pixel wise intensity difference between two images and sees how much noise is introduced. $PSNR=10\log10(MAX^2/MSE)$. A higher PSNR indicates a lower error which means a higher similarity.

Learned Perceptual Image Patch Similarity – It uses a pre-trained neural network (like VGG or AlexNet) to compare high-level perceptual features of image patches. A lower values indicates they are more perceptually similar when shown to a human.

Thus, the full image dataset having 20 images shows a better reconstructed 3D model than the half dataset.