Module: 3D Vision
Project: 3D recovery of urban scenes (Session 5)

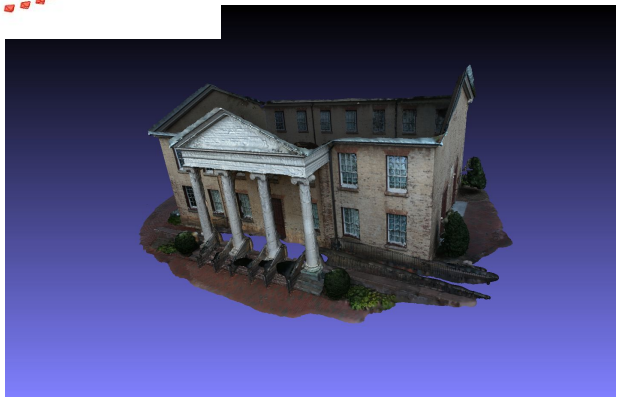
Original Lab: Marc Perez (marc.perez.quintana@upc.edu)

Goal

3D Reconstruction from uncalibrated images with a stratified method, by applying a Structure from Motion (SfM) pipeline in order to achieve a 3D sparse reconstruction. Dense 3D Reconstruction with Multi-View Stereo.

Sparse Reconstruction (Point cloud)

Images to 3d

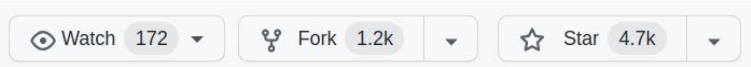


Dense reconstruction (mesh)

Colmap

https://github.com/colmap/colmap

- Opensource (BSD license)
- Widely used



Cited by 2892

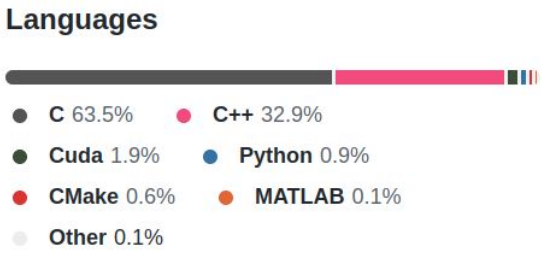
```
@inproceedings{schoenberger2016sfm,
    author={Sch\"{o}nberger, Johannes Lutz and Frahm, Jan-Michael},
    title={Structure-from-Motion Revisited},
    booktitle={Conference on Computer Vision and Pattern Recognition (CVPR)},
    year={2016},
}
```

Cited by 1139

```
@inproceedings{schoenberger2016mvs,
    author={Sch\"{o}nberger, Johannes Lutz and Zheng, Enliang and Pollefeys, Marc and Frahm,
    title={Pixelwise View Selection for Unstructured Multi-View Stereo},
    booktitle={European Conference on Computer Vision (ECCV)},
    year={2016},
}
```

- Documented
  https://colmap.github.io/
  https://demuc.de/tutorials/cvpr2017/
- Maintained

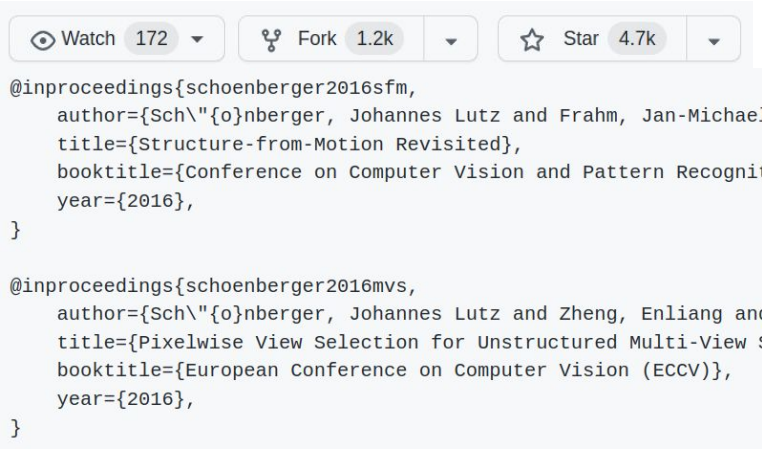
- Code in C/C++



Colmap

https://github.com/colmap/colmap

- Opensource (BSD license)
- Widely used



- Documented
  https://colmap.github.io/
  https://demuc.de/tutorials/cvpr2017/
- Maintained

- Code in python

Python functions provided
Adapted from the scripts at
https://github.com/colmap/colmap/tree/dev/scripts/python

Colmap installation instructions 

We will use Colmap, a state-of-the-art open source solution for rigid Structure from Motion.

You can find the code here: https://github.com/colmap/colmap
You can download pre-built binaries as explained here: https://github.com/colmap/colmap#getting-started

Since many of you had issues installing pycolmap in Windows, pycolmap is no longer used in the lab, therefore it is no longer required to build Colmap from source.

Mandatory Tasks

- 1. 3D mesh reconstruction from a set of images from the Gerrard Hall dataset. **(1.5)**
- 2. Step-by-step 3D mesh reconstruction on the CASTLE dataset (lab 4) following the notebook provided, explain each step. **(6.0)**
- 3. Configure the reconstruction to improve the results. **(1.0)**
- 4. Reconstruct a 3D mesh from images captured by you. **(1.5)**

Optional Tasks

- Read the paper on Structure from Motion [1] and comment its strengths and weaknesses **(+1.0)**
- Read the paper on Multi-View Stereo [2] and comment its strengths and weaknesses **(+1.0)**
- Read the NeRF [3] paper or another related paper from recent years and comment how it compares to the approach in the previous papers. **(+1.0)**

- Organized notebook/report and good programming practices and mathematical notation **(+1.0)**

[1] Schönberger, J.L. and Frahm, J.M., 2016. Structure-from-motion revisited. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4104-4113).

[2] Schönberger, J.L., Zheng, E., Frahm, J.M. and Pollefeys, M., 2016. Pixelwise view selection for unstructured multi-view stereo. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14 (pp. 501-518). Springer International Publishing.

[3] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R. and Ng, R., 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM, 65(1), pp.99-106.

Assignment

- Code is provided in python in a jupyter notebook.
- Deliver before 9 AM of next Thursday, February 9.

Deliverables

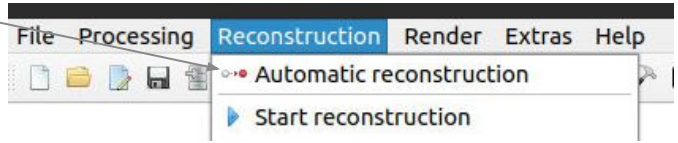
- **Jupyter notebook:** ready to run.
  - Document your code and decisions on markdown.
  - Be clear of what information is assumed/required for each algorithm/operation.
  - Understand the equations do not just reproduce them from the slides.
- **Report:**
  - Short report with in depth analysis, insights, problems and comments.
  - Do not paste code in report. I am interested in analysis and justification.
  - You can use the notebook as a report **IF, AND ONLY IF,** you format the notebook appropriately.
  - 

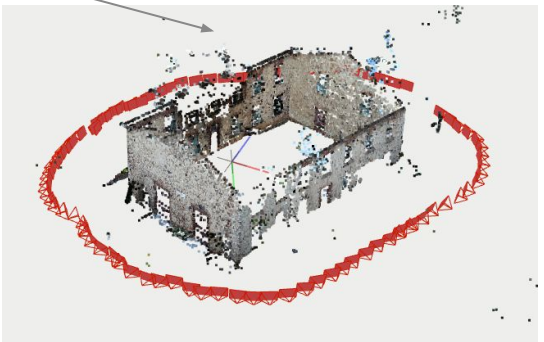
Please include some **feedback** on the lab. Was it too easy/hard? The points were correctly distributed according to difficulty? The installation process was easy? Something was not clear?
- 

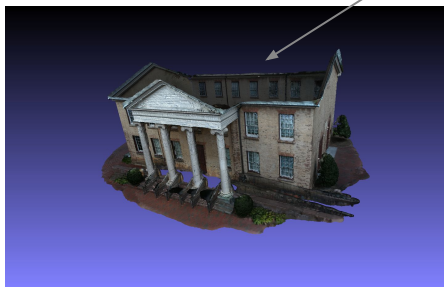
Meshes:
  - The **3D meshes** generated in all exercises or videos showing the 3D structure of the mesh. (link to Google Drive for example, if they take too much space)

1. 3D mesh reconstruction from a set of images from the Gerrard Hall dataset **(1.5)**

- Install Colmap: https://colmap.github.io/install.html
- Read the documentation: https://colmap.github.io/, https://demuc.de/tutorials/cvpr2017/
- Download the Gerrard Hall dataset: https://demuc.de/colmap/datasets/
- Run the automatic reconstruction on the Gerrard Hall dataset
- Visualize the sparse reconstruction on Colmap
- Visualize the dense reconstruction and remove noisy points (for example on Meshlab: https://www.meshlab.net/)







2. Analyze reconstructions using python. **(6.0)**

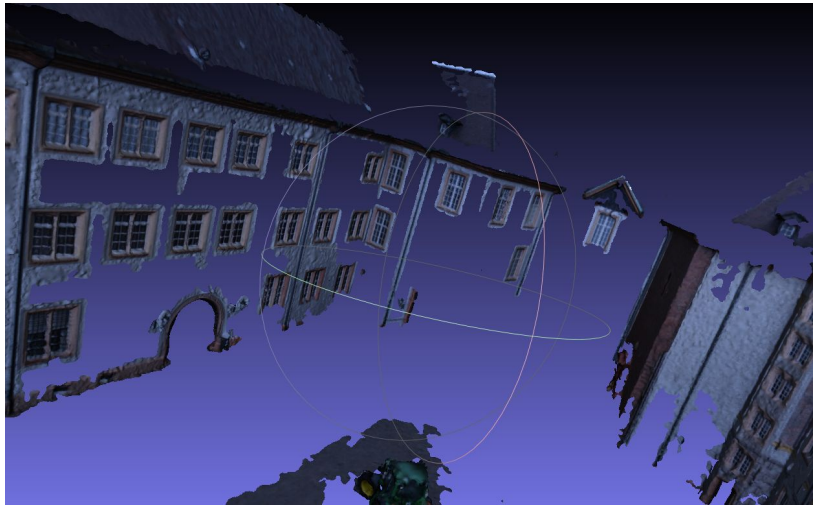
- 2.1 Run the notebook, using the Gerrard Hall reconstruction **(0.5)**
- 2.2 Plot the 3D points coloured according to the number of images and error. **(0.5)**
- 

2.3 Plot the 3D points that originated from a keypoint in the first image. Also plot the image with the keypoints **(1.0)**
- 

2.4 Create a visualization for the number of matches between all images. **(1.0)**
  - For example: https://seaborn.pydata.org/generated/seaborn.heatmap.html
- 2.5 Visualize the keypoints and matches between the two images used in lab 4 using Colmap, how it compares to the results from lab 4? **(1.0)**
- 2.6 Triangulate and visualize the 3D points from the keypoints extracted using Colmap on the two images used in lab 4, how it compares to the results from lab 4? **(1.0)**
- 2.7 Visualize the sparse reconstruction using the 2 images from lab 4, and the complete CASTLE dataset. Comment on the differences between techniques and number of images used. **(1.0)**

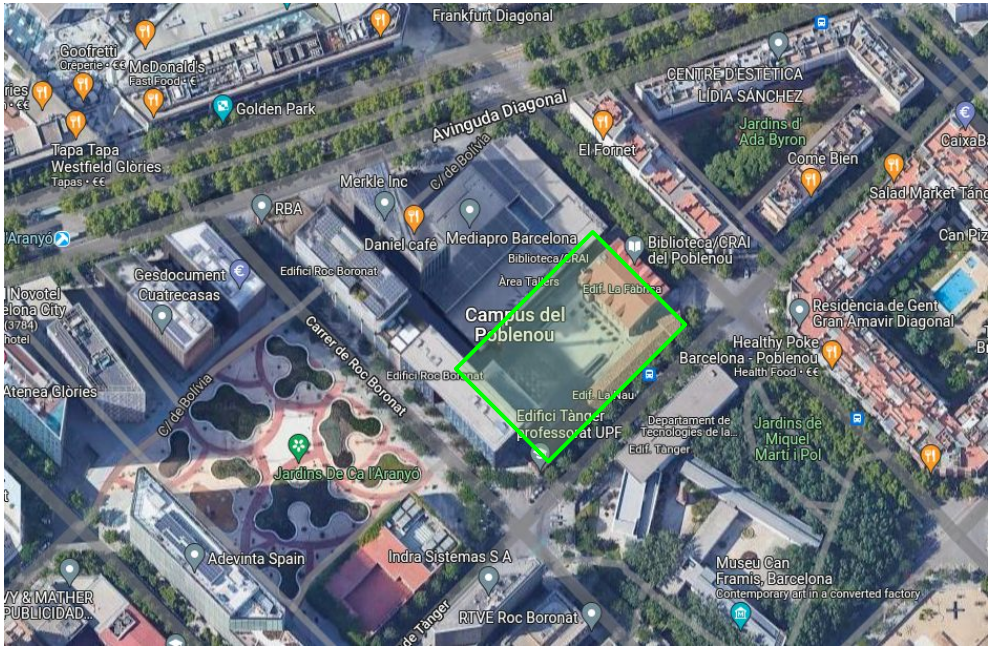
3. Configure the reconstruction to improve the results. **(1.0)**

- Configure the reconstruction to get the best possible mesh, explain the parameters changed and why you decided to change them. Remove noisy points with Meshlab or others.



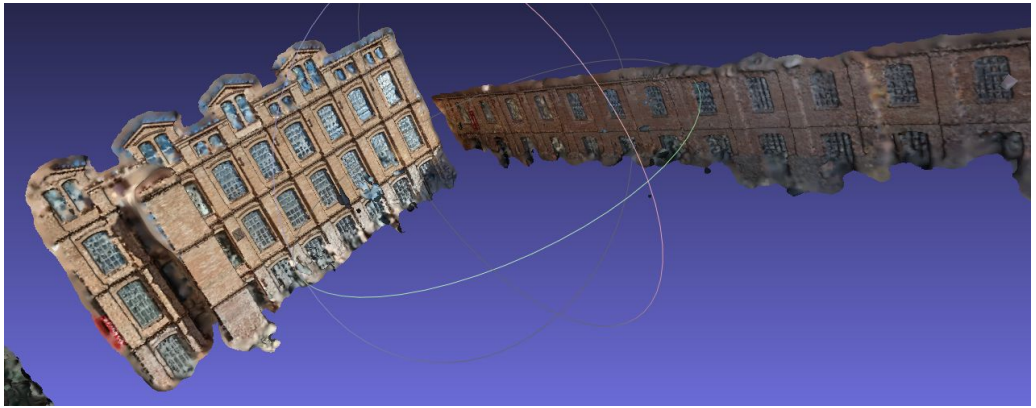
4. Reconstruct a 3D mesh from images captured by you. **(1.5)**

- Ideally from the same environment, to be able to compare between groups: [UPF campus](#)!



4. Reconstruct a 3D mesh from images captured by you. **(1.5)**

- Ideally from the same environment, to be able to compare between groups: UPF campus!
- Image capturing is important to get good results.



4. Reconstruct a 3D mesh from images captured by you. **(1.5)**

- What are the requirements for image capturing to get good 3D reconstructions?
- You might need to change the configuration to get a good reconstruction, comment on the changes you make.
- You can apply it to other scenes or objects.
- You can try with different cameras.
- You can try with images from the web.