SOE VR Lab

Instruction for Photogrammetry

Main Editor: Hanyuan Xiao Version: 1.0 [07/19/2018]

NOTE For users in RPI SOE VR Lab, skip *Preparation Hardware and Software* part. Everything has been purchased and meet the requirements.

Preparation in Hardware and Software

Items listed below are the lowest requirement for photogrammetry. Optional items are labeled or noted. For more options, read documentation *Equipment Purchasing Wishlist*.

Hardware

- A graphics PC
    - Windows OS (compatible with software listed below)
    - CPU: Intel 7th i5 or better
    - GPU: NVIDIA GeForce 1060 or better
    - RAM: at least 32 GB
    - Hard drive storage: at least 200 GB free for use
- Camera
    - Sports camera is not recommended
    - SLR camera (resolution is as high as possible, use JPG format)
    - 360-degree panorama camera is recommended (8K at least)
- SD card (at least 32 GB)
- VR system
    - HTC Vive (Pro)
    - (optional) Oculus Rift (or Go)
    - (optional) Samsung HMD Odyssey
- Tripods for camera and VR cameras
- (optional) Drone

Software

- Model generation
    - RealityCapture
    - (optional) Agisoft Photoscan
    - (optional) Autodesk Recap bundle
- Post-edition
    - Autodesk Maya
    - (optional) ImageMagick
    - (optional) Modo
    - (optional) Photoshop
    - (optional) IrfanView

- VR view
    - Steam & SteamVR
    - Oculus

Start Generation

- Select a place to take pictures
    - A bad place will not generate an desired model and it will be a waste of time. Here is a very detailed instruction published by Unity, and some indispensable requirements are listed below.
        - Do not choose a too large scene for photogrammetry. The algorithm needs to match pixels in different photos and short movement in reality will not have scene far away move. The result can only be blank.
        - If you really want to bring some large scene into photogrammetry such as *Mars* in SteamVR workshop, you need several more steps in post-edition which are not listed in this doc.
    - Outdoor scene always involve wind, lighting, and weathers.
        - Choose cloudy day, which does not have specular lighting effect.
        - Do not choose windy day, which does not move anything in the scene during photo capturing. It is OK if you can ensure nothing can be moved through whole scene. (Remember you have to ensure the camera is not moving as well)
        - Do not choose rainy or snowy day, which will bring the worst effect that you can imagine.
    - Make sure all photos at one location outdoor are captured in a relatively short time.
        - Any change in sunlight, overall light intensity, or change in objects in the scene can make algorithm be difficult to match pixels.
- Take images
    - Why is a drone recommended?
        - Here is a video that generates a large scene with only about 130 images.
        - A drone can scan terrain and generate ground surfaces more accurately.
        - You can bring UV map taken by drone to model generation software for more accurate model.
    - To build high-resolution model that is competitive with reality, a set of photos needs to be taken at every time you walk 1 foot long (approximately 30 centimeters). Never imagine the algorithm can match pixels with large difference.
        - Note that high-resolution output requires large size of data and high-quality mode in software which will cost much longer time.
    - Take 360° pictures.
        - For example, use high-resolution 360° camera.
        - Or take at least 8 pictures at every 45° at one spot to cover a 360° scene.
    - Generally, do not be worried about ground because it is closed to the height you put the camera and many images can cover texture and geometry accurate enough.
    - However, if you want to cover higher altitude objects (such as wall of tall buildings), you need to adjust the angle of camera.
    - Do not be worried about ISO and change in light intensity. Just use Auto mode of camera would be fine. Because ...
        - We want to simulate function and adjustment of eyes and change in light can be easily offset.

- According to experiment, most algorithms can transit between different lighting environments smoothly.

There are many optional softwares for Photogrammetry. The table below shows the comparison among the most widely used ones.

	Usage	Difficulty to use	Price	When to use
VisualSfM (CMVS/PMVS)	Efficient: quickly generate sparse point cloud Partial pipeline: Cannot generate dense mesh	Very easy (With UI and tutorial video)	Open Source	If your goal is to see whether all images can be aligned, VisualSfM can test image data as fast as possible. (necessary and efficient when dataset is large and/or images are dim) If your goal is to generate a final dense model, try others. OS: (Recommend) Windows 10
OpenMVG+OpenMVS	Average efficiency Easy and trivial under stable condition of OS and hardware settings Full pipeline	Difficult (No UI -- all in terminal; Debug even after successful installation is troublesome; Require knowledge of Linux OS, hardware debugging; See tutorial for OpenMVG and OpenMVS)	Open Source	Best pipeline for small or medium size scenes (outdoor/indoor) Quality is good enough but mesh is complicated -- output file may be overly large OS: (Recommend) Linux
COLMAP	Average efficiency Easy with UI and tutorial Full pipeline	Easy (after installation; with UI; see tutorial)	Open Source	Best pipeline for large or extremely large (landscape scanning) scenes Low detail in meshes but is able to identify and generate most from given dataset Mesh is noisy and round in geometry (both model and texture file can be unnecessarily extremely large) OS: (Recommend) Linux
Commercial Softwares (e.g. RealityCapture, Pixel4D)	Efficient Commercial softwares use the most basic algorithm to build models for small or medium scenes, but the result is often the most ideal ones. Full pipeline	Very easy (full pipeline in menu tab and tutorial)	Start from $99/3 mo	Most ideal for object modeling and small/medium scene generation. OS: (Often specified) Windows 10

Note: To compare benchmarks of most popular algorithms, read the paper *Tanks and Temples: Benchmarking Large-Scale Scene Reconstruction (Knapitsch, Park, Zhou, & Koltun, 2017)*.

Option 1 RealityCapture from CapturingReality

Follow the workflow in algorithm. (Many softwares have clear procedure listing in menu tab.)

- Align images
    - Alignment of images aims at identifying pixel pairs in different images.
    - This step should not take long time. For example, a 900 image set costs 5 minutes to finish.
    - For some softwares such as RealityCapture, you do not have options for accuracy at this step.
    - For other softwares such as Agisoft Photoscan, choose wisely for the accuracy. High or Very High can take up to 30 hours in this step for large image set.
    - Do not believe estimated time remaining in any software, the time depends on images that are input serially. It means as new photo being scanned and calculated, the time will always be recalculated based on coefficients of that image.
    - We recommend open source software VisualSFM by Wu that is free and efficient.
- Build dense cloud
- Build mesh
- Simplify
- Colorize and apply texture
- Export

Check the output model in any software. Go to Post-edition step if you …

- Need to delete noisy data (such as unnecessary meshes).
- Find some meshes are totally wrong.
- Need to clip out objects or surfaces that images do not cover but are rendered inaccurately.

Option 2 OpenMVG+OpenMVS

The tutorials for installation and usage (see links in the table above) are feasible and clear. The following instruction is aim at identifying potential errors and debugging.

- What should be run to use Structure-from-Motion from OpenMVG?
    - After compiling the repository to local directory, cd to `openMVG_Build/software/SfM/` and run
      `python SfM_SequentialPipeline.py [full path image directory] [resulting directory]`
    - This will generate two folders `matches` and `reconstruction_sequential`. Cd to the later one and find a binary file named `sfm_data.bin`.
    - Cd to `openMVG_Build/Linux-x86_64-RELEASE/` and run
      `./openMVG_main_openMVG2openMVS -i sfm_data.bin -o scene.mvs`
    - Continue steps in OpenMVS
- Where to find scripts of OpenMVS
    - First make sure to finish building OpenMVS via CMake.
    - Cd to `openMVS_build/bin` and you will find the scripts shown in Usage page on GitHub.
- You need to copy original image dataset to a folder named `undistorted_images` under the directory `~/openMVS_build/bin/` to ensure the use of OpenMVS.
- Any error like `Aborted: core dumped` or `segmentation fault`
    - Search the issue in OpenMVG/OpenMVS GitHub at first. Here are several errors I encountered and where to find the solution:
        - OpenMVS RefineMesh Segmentation Fault (r/ core dumped)
        - Can't use SfM issue
        - OpenMVG The input SfM_Data file "..." can't be read

    - There should not be other error when running if installation is finished and successful. Otherwise, it can be any incompatibility with hardwares such as shortage of RAM.
    - Post any issue on GitHub if any issue remains. Make sure to label information of hardwares (e.g. CPU, GPU, RAM, OS, etc.).

Option 3 VisualSfM

Structure-from-Motion can be easily used after downloading Changchang Wu's version. Remember to integrate the execution of Yasutaka Furukawa's PMVS/CMVS tool chain to fulfill dense reconstruction which is not with VisualSfM. Read the instruction from the link posted in the table above.

Because PMVS uses cluster dense reconstruction, the model will be generated in many files but will be combined at the last step. The number of clusters and running time can be checked from log window of VisualSfM.

Option 4 COLMAP

Installation of COLMAP requires nVidia CUDA. CUDA 8.0 is recommended since it is more stable than version 9.0. Note that sometimes nVidia graphics card driver goes counter with CUDA when installing (rarely). My solution is to uninstall the original GPU driver and install older version like 365.

It may require to disable X server when installing CUDA, there are many ways to disable it in Linux from Internet such as this I used.

Post-edition

- Use 3D model edition softwares such as Autodesk Maya.
- Sometimes, models generated are very large (≥ 700 MB) or texture files are extremely large (≥ 20 MB). Use meshlab to clean up the model. There are many functions to clean up such as remove duplicated vertex or remove faces from non manifold edges in Filters>Cleaning and Repairing from menu bar. Also refer to some tutorial videos on Youtube such as this.

Input into SteamVR and Publish

- Be sure that an HMD is plugged to computer before starting SteamVR. Otherwise, you will see an error popped.
- Follow the official tutorial of Steam. Some steps may be confusing and thus being clarified below.

Clarification:

- At the very beginning
    - Start SteamVR. Small SteamVR window will pop up usually at the lower right corner of screen.
- Create your SteamVR Addon
    - For a brand-new project, select "Create Empty Addon". To add a version of same scene or object, select the project in the list. Click "Launch Tools".
- Bring your model in with Model Editor
    - If importing OBJ file from RC or Agisoft Photoscan, choose "Z Up (Valve)".
- Bring your model into Hammer

    - It is important to remember that the scale of scene can only be changed by scales of whole scene. Change in size of figure has no effect.
    - There is no way (I can found so far) to specify length of object in the scene, so you need to adjust the scale of x, y and z simultaneously for many times. Generally, if you want everything to be on-scale as in real world, try from “x=45, y=45, z=45”.
- Add a light
    - This step is necessary when you want to bring colored objects into the scene, such as *beach ball* grabbed from SteamVR menu (available when viewing final model).
    - If no light is added, scene can still be seen as it should be, but any objects created in SteamVR will be totally black and thus there will be no shadow casted.
    - Types of light can be picked from *Entity Class* as “info_player_start”. Here is a short tutorial about how to put a light.
        - **If generating outdoor scene that has sun or other worldly baked lights, choose “light_environment”.** Note that for this light, location where you put the light does not matter. However, intensity and orientation need to be adjusted.
        - **If generating indoor scene that has light bulbs (i.e. point light), choose “light_spot”.** After add one into the scene, you will see the visual effect of the light. This type of light is seldom used, because you usually do not want to build a haunted house. (i.e. Everything that is not inside the affected region of light will be totally black.) The orientation, intensity, location and maximum affected region should all be taken into account now.
        - **If generating indoor scene that has ambient effect (mostly ubiquitous), choose “light_environment” or other ambient light (you can try to see visual effect) and pick the most accurate type consistent with real world.** Again, the imported scene will not be affected unless you build any object in SteamVR.
    - In most cases, “light_environment” is put at least. You can add other light sources besides.
- Placing Teleport area
    - So far there is no way to draw polygon with any shape you want. Polygon must be outlined along the side or diagonal of square.
    - To merge many polygons into one as teleport area, draw many separated polygons firstly. Then, select all of them and choose merge from the horizontal bar at nearly the top.
- Submit to the Steam Workshop
    - To publish a project, you need to submit this agreement at first.
    - You need to add at least $5 to your account to publish a project open to the public. The prepaid deposit is only an authorization.

The End