

Princeton University  
COS429 Computer Vision  
**Problem Set 4: Cloning Yourself**



Figure 1: SFMedu: Structure from Motion (SfM) and Stereo Reconstruction.

This assignment is focus on 3D reconstruction. Structure from Motion (SfM) and Stereo Matching is one of the most popular pipeline for 3D reconstruction. In this assignment, we will study a full pipeline of such a 3D reconstruction system.

**Problem 1** *Run the SFMedu system*

Download the code we provide. And run the SFMedu system by running the SFMedu2.m script in MATLAB. Make sure the it runs successfully and produce a dense.ply point cloud file. Install MeshLab from <http://meshlab.sourceforge.net>, and open the point cloud in MeshLab and rotate around the model. Take a screen capture of the point cloud and put it in the report as the answer to this question.

**Problem 2** *List the Major Steps*

Read the code carefully, especially SFMedu2.m. Summerize the major steps for SfM and dense stereo matching, and write it down in the report as answer to this questions.

**Problem 3** *Principal Point*

What is the assumption for the principal points in this system? Under what condition will this assumption get violated?

**Problem 4** *Data Structure*

Graph is a data structure that we used in the code for bundle adjustment. It has 5 major fields. What are their names? What is the data in each field? How the data is organized? Why we need these 5 fields?

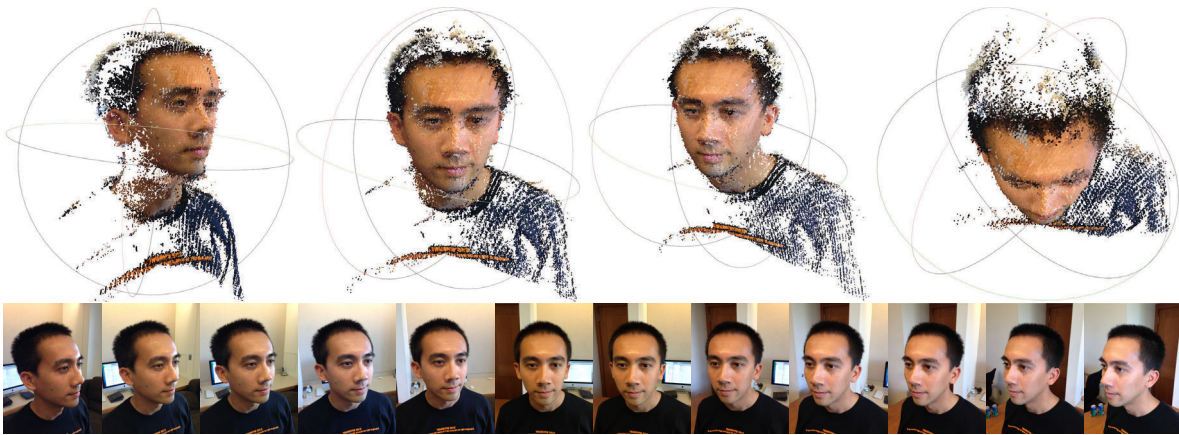


Figure 2: 3D Reconstruction of Jianxiang's head by SFMedu. The bottom row contains the images used for reconstruction. [Jianxiang: Oh no! The outliers of matching propagation messed up my hair!]

### Problem 5 *Reconstruct Yourself*

Ask your friends to take several pictures (at least 5 pictures) of your head (or an interesting object or scene, don't pick a boring or ugly object :p ), run the system on your photos. If it fails, figure out why it fails, and change the code or retake good pictures until it works. Submit the input images, as well as a screen capture of your reconstruction in the report. Explain all the changes or things you have to do to make it work. Which step do you think is the most unstable one? Explain why. List suggestion for a normal user without any computer vision background (such as your friends), about how to take good pictures to make it work.

In Figure 2, we used 12 pictures taken by my iPhone 5 to reconstructe my head, as an example.

### Problem 6 *Compute the Reprojection Error*

Write a function `printReprojectionError(graph)` that takes a graph (as defined in the SFMedu system), and print out the current reprojection error, so that you can insert the function into many steps for debugging purpose, to check if the reconstruciton errors are getting smaller. A call to this function is at Line 160 of `SFMedu2.m`.

Submit the function as code, and also include the source code of your function in the report PDF file.

### Problem 7 *Visualize the Reprojection Points*

Write a function `visualizeReprojection(graph,frames)` that takes a graph (as defined in the SFMedu system), and draw the 3D keypoint point cloud projected onto each image, as well as their observed location. Figure 3 shows an example of the output for your function. You are required to write a function to produce the same kind of visualization. Each observed keypoint is represented by a red x, and its reprojection of its 3D estimated location is represented by a green +, and these two points are connected by a blue line. For the 3D points that are



Figure 3: Result produced by function `visualizeReprojection(graph,frames)`.

projected to the image but not observed from the image, it should be shown as a yellow o. A call to this function is at Line 162 of `SFMedu2.m`.

Submit the function as code, and also include the source code of your function in the report PDF file.

### Problem 8 *Levenberg-Marquardt*

In `bundleAdjustment.m`, we use matlab function `lsqnonlin` to minimize the objective function using Levenberg-Marquardt algorithm:

```
[vec,resnorm,residuals,exitflag] =
lsqnonlin(@(x) reprojectionResidual(graph.ObsIdx,graph.ObsVal,px,py,f,x), [Mot(:);
Str(:)], [], [], options);
```

Please read all the code and figure out what is the objective function we are optimizing for. Write down the math equation of this objective function in your report.

The Matlab implementation `lsqnonlin` of Levenberg-Marquardt is not suitable for very large scale problem. Read the document of `lsqnonlin` and explain why this statement is true in your report.

### Problem 9 *Motion Adjustment Only*

Bundle adjustment typically optimize the Structure (the 3D location of the points) and the Motion (the camera parameters) together, as the above line of code. But we can also fix the Structure, to adjust the Motion only, i.e. not allow the 3D points to move, but only changing the rotation and location of the cameras to optimize the objective function. With some good initialization, this can be used as a fine adjustment to solve camera resectioning task as well ([http://en.wikipedia.org/wiki/Camera\\_resectioning](http://en.wikipedia.org/wiki/Camera_resectioning)). Fill in Line 34 in `bundleAdjustment.m`.

Please write down the math equation for the objective function, and also change the above line of code to do this. Submit the code and also paste the code in the report PDF file.

### **Problem 10** *Structure Adjustment Only*

In the same way of thinking, we can also fix the Motion, to adjust Structure only, i.e. not allow the cameras to move, but only allows the 3D points to change their locations. Similarly, with some good initialization, this can be used as a fine adjustment to do “triangulation” for reconstructing the 3D point locations. Fill in Line 30 in `bundleAdjustment.m`.

Please write down the math equation for the objective function, and also change the above line of code to do this. Submit the code and also paste the code in the report PDF file.

Although this can be used to do triangulation, there is certain drawback. What is that? Answer this in the report.

### **Problem 11** *Why to Estimate the Essential Matrix?*

Given that our goal is to minimize the objective function, why not just do the bundle adjustment directly? Why we need to go through all the trouble to estimate the essential matrix? Is there other way to bypass the estimation of essential matrix? Some smart trick that let the user take pictures in a certain way? Provide your answer in the report.

### **Extra Credit [Optional, but highly recommended]** *Smarter Graph Merging*

The current reconstruction is merging the sequence sequentially, by adding one camera into the mergedGraph in each step and do bundle adjustment again. A better graph merging will help to make the system more robust. For example, we can merge the graphs that have the maximal number of keypoint correspondences, instead of sequentially picking one.

### **Extra Credit [Optional]** *Run on Large Scale*

Run the system with a lot of images (> 50), figure out the speed bottleneck, and see how to make it faster. Hint: possible idea include the above smarter graph merging, as well as no bundle adjustment in each step. This is also an open research topic (e.g. <http://vision.soic.indiana.edu/projects/disco/>).

**Extra Credit [Optional]** *Rectifying Images for Stereo Matching*

After SFMedu compute the camera poses, we can use the camera extrinsics parameters to rectify the image. And then we can use stereo matching to generate a disparity map and visualize it in 3D. In another word, instead of using the `denseMatch.m` in our system that does uncalibrated matching propagation, we can rectify the images and do a typical stereo matching algorithm and then convert the disparity back to the original image space to obtain a 3D point cloud. <http://www.mathworks.com/discovery/stereo-vision.html>

**Extra Credit [Optional]** *Different Focal Length*

Currently, we assume that all pictures have the same focal length and resolution. Change the code to allow images with different focal length and resolution.

**Extra Credit [Optional]** *More Serious Bundle Adjustment Package*

Ceres solver <http://ceres-solver.org> is a standard package for bundle adjustment (the other popular package is sba <http://users.ics.forth.gr/~lourakis/sba/>). What is the advantage of Ceres over the Matlab `lsqnonlin`? Replace our bundle adjustment code by ceres or sba.

**Extra Credit [Optional]** *Intrinsics*

Change the code to optimize for the full intrinsics matrix.

**Extra Credit [Optional]** *Radial Distortion*

Change the code to handle radial distortion as well (a fixed radial distortion or one that estimate the radial distortion).

**What to submit:** You need to submit two files: one PDF file for the report that contains your name, Princeton NetID, all the pictures taken and text to answer the questions; one ZIP file (not RAR or any other format) that contains all source code for your system, and a “SFMedu2.m” file that takes no parameter as input and run directly in Matlab to generate the results reported in your PDF file. Both the PDF and ZIP file should be named using your Princeton NetID underscore `cos429ps4`. As an example using my account, they should be named “`xj_cos429ps4.pdf`” and “`xj_cos429ps4.zip`”. To **verify your result** and **detect plagiarism** to make sure there is no cheating, we will use an automatic program to run your code and compare your code with other students’ (including both this year and all previous years) and public available implementations (e.g. from Google, Bing, Github). Therefore, please follow the file format to make our grading job easier. Failure to follow these rules will result in losing your grade.