Huan-Yang Chang
Team G: Excalibr
Teammates:
Man-ning Chen, Yiqing Cai, Sambuddha Sarkar, Siddharth Raina

ILR03
10/27 2016

Individual Progress

1. Movement matrix and corresponding images

For future usage in geometric calibration and light-field calibration, the relation of position and corresponding images was important. In the geometry calibration, we might use the combination of Tsai's and Zhang's methods to get a better accuracy[1][2] . In Tsai's method, the movement of two images is should be known for calculating the camera parameters. For light-field calibration, we have to collect the light-field data and the sphere camera's position (the sphere camera is located on the end effector of robot arm). With the data pair of lightness and position, we could use that to get the environment light field. Hence, the movement matrix and corresponding images were an important base for our future work.

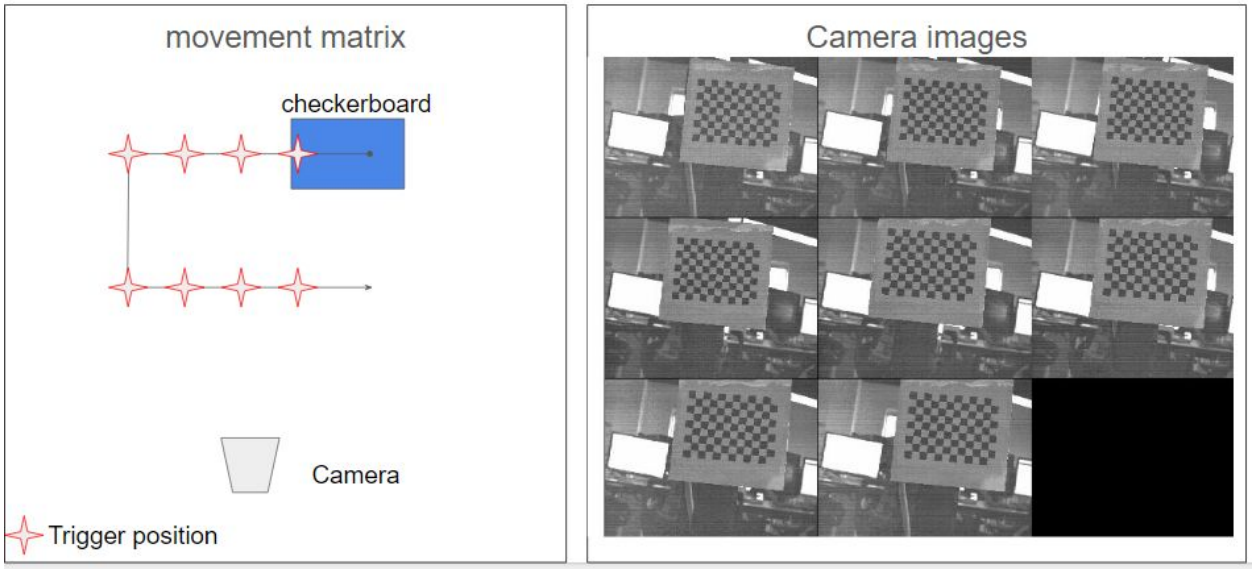

Fig 1. Movement matrix and corresponding images

[1] http://research.microsoft.com/en-us/um/people/zhang/calib/
[2] http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/DIAS1/

2. Geometry Calibration Pipeline draft

By studying the off-the-shelf calibration toolbox in Matlab(Caltech camera calibration toolbox)[3], This method was based on Zhang's method.

I summarized the four steps in their method that were image collection, key point detection, Initial estimate, and Nonlinear optimization. In implementing this toolbox ,the first two steps were not automatic, I had to move the camera to take different images in different angle and position, and it usually needed >12 images to get the enough equations to solve the least square equations . And then I had to set the origin of the checkerboard and manually adjusted for better corner detection. Hence, if we want to automatize the geometry process, we needed to figure out how these two steps work without human's operation. By consulting with our sponsor, we might use the robot arm to move the object and because we used 3D calibrated object, we probably don't need that much images to calculate the parameters. For origin setting of the images, we probably used the April tags(Fig 3) to define the origin and direction of the target.

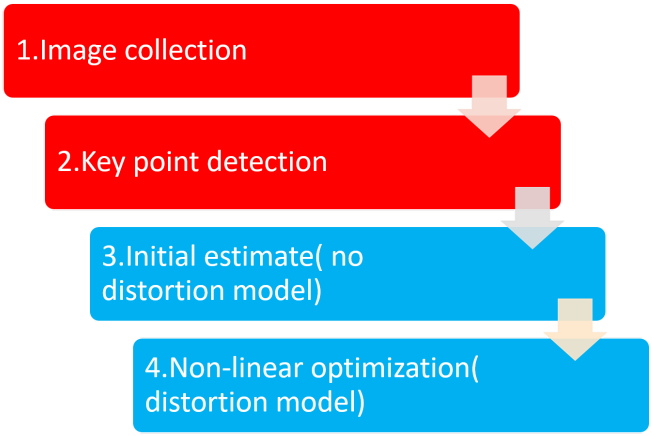

Fig 2. Pipeline in camera toolbox in Matlab

[3] https://www.vision.caltech.edu/bouguetj/calib_doc/

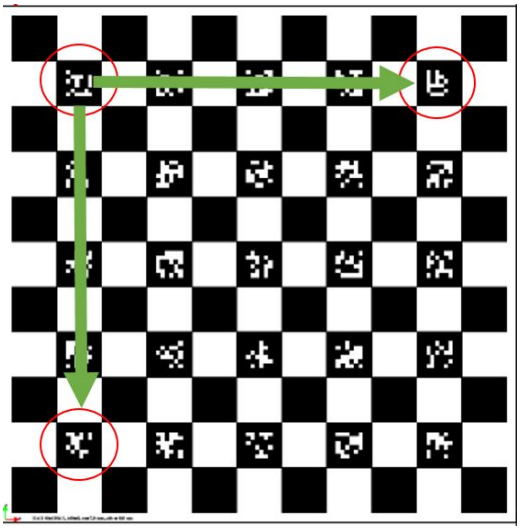

Fig 3. Using April tags to set origin and rough direction.

Challenges

1. Future total integration

For our progress now, we focused on solving the each subsystem level problems. However, in the spring semester, we have to combine everything together and try to make it work automatically in four calibration requirements we set. Hence, we probably will discuss to conduct each calibration earlier and try to integrate them into a system earlier.

2. Different OS system

The employee in Sponsor's company already worked on geometry calibration for a while and had some previous work. But this work was created in Linux that we were not familiar with. Hence, it made harder for me to understand the previous work and think how to integrate it into our project. Besides, the robot arm control system was based on Windows and the camera system was based on Linux system. Hence, we might need to decide either to combine this system together or build the communication platform.

Teamwork

Work was done in this week	Name
Movement planing and root cause analysis	Sam
Get movement matrix and corresponding images	Peter
Solve FPN	Mandy/ Cece
Implement in Photometric calibration	Mandy/ Cece/Sid
Geometry calibration pipeline draft	Peter

From this week, because the robot work in the linear actuator was almost done, so I split from robot part and started to work on the geometric calibration part. I start to understand the process of off-the-shelf calibration methods and consult the sponsor for help. Mandy ,Sid, and Cece were responsible for FPN and Photometric calibration. They tried to implement the method in the paper which sponsor provided and test the method in Matlab. From this week, They also started doing Photometric calibration by implement the method in this paper. I will also work with Sid in the geometric calibration together.

Future Plans

1. May change requirements of this semester from spring to fall

Because we found that photometric calibration would have dependency with the FPN calibration. And Sponsor also wanted us to conduct the photometric calibration now. Hence, we will discuss the new schedule and decide whether postpone the light-field calibration

2. Implement the geometry calibration program

We will use the geometry calibration program provided by the sponsor. First, we have to read and understand it’s structure and test it. Then we have to combine their code to our system and make the subsystem (geometric calibration system) work automatically.