# Omnidirectional Cameras Calibration and Stereo 3D Reconstruction

## A new idea for GSoC 2015

**Baisheng Lai**

## Abstract

This proposal is about omnidirectional cameras calibration and their stereo 3D reconstruction, which are new ideas for GSoC 2015 of Opencv. Omnidirectional cameras consist of fish eye cameras and catadioptric cameras, both can be modeled by C Mei's unified spherical model. Opencv new can only calibrate fish-eye cameras so such a unified model can be useful. The conventional way to reconstruct 3D scene from omnidirectional stereo image pair is to rectify them to perspective ones but need to use large images to contain all details, or will lose some details due to the large distortion. In this project, first, I plan to implement omnidirectional cameras calibration tools based on C Mei's model. Second, I will use a longitude-latitude mapping to get the rectified images which contains all the details while not increasing image size no matter how large the fields of view is. This project will be helpful for Opencv since catadioptric cameras can also be calibrated and all omnidirectional cameras can do stereo 3D reconstruction easier.

## Project Goals

This project has three goals.

1. Implement unified sphere model [1] and the calibration tools for this model so that fish-eye cameras and catadioptric cameras can all be calibrated.
2. Implement stereo rectification method based on longitude-latitude mapping [2] so that all details in images will be kept no matter how large the field of view is. 3D reconstruction from disparity map should also be modified.
3. The corner extraction in omnidirectional calibration is almost manual due to the large distortion. Opencv now has no manual or automatic corner extraction tool for omnidirectional cameras. They will be implemented.

## Implementation Details

### 1. Implementation for the first goal

The main work of this project is to achieve the first goal. A calibration toolbox consists several parts.

1) Corner extraction.
2) Geometric relations. Like projecting image to 3D, projecting 3D to image, composing R and T. It is the key work of calibration toolbox and the camera model is embed in it. The target is to get the back projection error.
3) An optimization method, like LM algorithm, which is used to optimize parameters by minimizing back projection error.

In a calibration toolbox, most manual inputs are in the first part, others can be fully automatic. The corner extraction for omnidirectional cameras is not available in Opencv. Users must first extract corners out of Opencv and input them. I will implement a manual and automatic corner extraction tools so that users can do all calibration procedures in Opencv.

The second part is the most important in the whole project. Design a new calibration toolbox is too heavy. Fortunately, the structure of existing Opencv calibration code can be used for reference, especially the fisheye related code. Some code in Opencv calibration can be directly used and others should be modified according to the camera model. Some advanced functions may be ignored to make the code clean and readable. They can be added afterwards. I will read the code of Opencv fish-eye calibration code first and see what can be re-used and be familiar with the calibration code in Opencv. The authors of [1] also provide Matlab calibration toolbox [3]

For the third part, the LM algorithm is somehow a standard optimization algorithm, its code is also in Opencv calibration.cpp file. I will first use a more friendly code in [5] and then use the code in calibration.cpp in Opencv.

By the way, a new namespace named "omnidirectional" like "fisheye" in Opencv will be created.

**2. Implementation for the second goal**

The longitude-latitude rectification method has been implemented by me. Please go to this page [https://github.com/jiuerbujie/stereo_matching](https://github.com/jiuerbujie/stereo_matching). The results in the **Background for This Project** section is generated by this code.

**3. Implementation for the third goal**

First, a manual corner extraction tool like in many calibration toolbox will be implemented. It is based on corner detection in a small window so it will not be hard. After this, I will try to implement an automatic so semi-automatic corner extraction tool. The toolbox in [4] has an automatic one for corner extraction but the source code is unavailable. I may contact the author to ask which method is used and implement it. I also find a research paper [7] which describes another method and can also be implemented.

## Benefits

1. Since Opencv does not have tools to calibrate catadioptric cameras, the unified sphere model in this project will help to fill in the gap.
2. To apply stereo matching, the existing Opencv rectification function can only get perspective image. Some scene will be lost or large black region must be introduced (see **Background for This Project** for details). The longitude-latitude mapping allow users to reconstruct the **whole** 3D scene captured no matter how large field of view omnidirectional cameras have.
3. The man-machine interaction tool will help users save time and make users more comfortable during calibration.

## Timeline

The roughly timeline is given below.

| **Part 1** | **Calibration study and reference code reading** |
|---|---|
| Mar 27 to Apr 10 | Review of omnidirectional camera calibration |
| Apr 11 to 18 | Opencv calibration code reading |
| Apr 18 to 26 | Other reference code reading |
| **Part 2** | **Calibration tools coding** |
| May 15 to May 27 | Geometric mapping coding |
| May 28 to Jun 10 | Calibration main function |
| Jun 11 to Jun 20 | Debugging |

| | |
|---|---|
| Jun 21 to Jun 25 | Image Rectification |
| Jun 26 to Jul 03 | Middle term evaluation |
| **Part 2** | **Advanced features** |
| Jul 4 to Jul 24 | Manual and automatic corner extraction tool |
| Jul 25 to Aug 5 | Add more features for calibration |
| Aug 6 to Aug 14 | Overall debugging |
| Aug 15 to Aug 20 | Writing documents |
| Aug 21 to Aug 28 | Final evaluation |

## About Me

My name is Baisheng Lai, I am a 2nd year PhD student at Machine Vision based Navigation Lab of Zhejiang University, China. My research fields and interests including calibration, image processing and sparse representation. When I first go into Computer Vision, I learn the details about camera calibration, most on omnidirectional cameras and do some research in calibration.

I have a research paper titled "omnidirectional camera calibration from a low-rank textured image", which is accepted by ICIP 2014 (unfortunately, it is removed because of no-show). This paper proposed a self-calibration method for omnidirectional camera with a unified sphere model and is based on sparse representation. Since then, I know the details of unified sphere model and camera calibration.

I also publish a paper [6] on Opt Express which is about image deconvolution.

I am quite familiar with C++ and Opencv. Most of my lab's projects I participated are based on Opencv. Many of them used stereo and I am responsible for camera calibration to 3D reconstruction.

In last summer, I participated a CSDN Camp, a Chinese version of GSoC. I implemented some online machine learning algorithms based on Spark Steaming. Refers to the code by https://github.com/jiuerbujie/online-learning-Spark

If you are interested in the technical details of this project, please have a look at the following section.

## Backgrounds for This Project

### 1. Omnidirectional camera calibration

Camera calibration is a basic task in machine vision. The goal of camera calibration is to get camera parameters, including internal parameters such as focal length and optical centers as well as external parameters which determine the location and pose between multiple cameras.

Omnidirectional cameras are extensively used because of its large field of view, which means more information about a scene can be attend, by introducing large distortion, especially near the boundary of images. Omnidirectional cameras consist several types of cameras, such as fish-eye cameras and catadioptric cameras. Here are two examples of them:

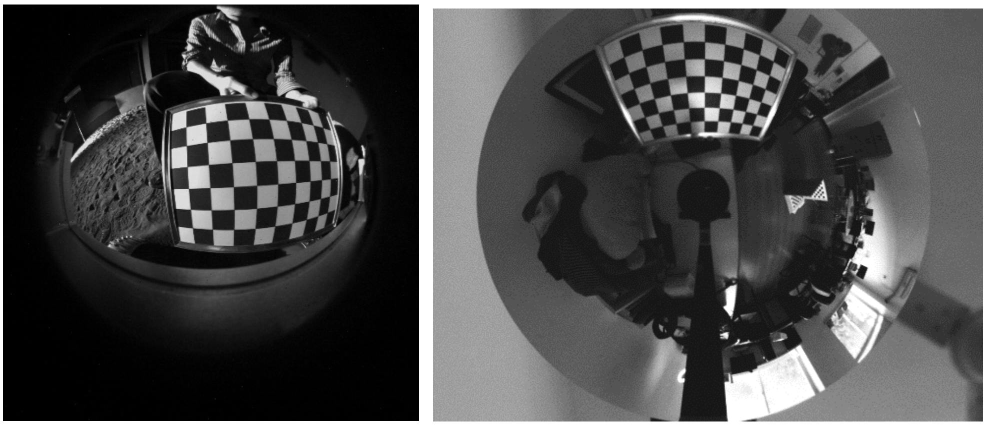

Fig 1 Images of fish-eye cameras (left) and catadioptric cameras (right)

Catadioptric cameras can be further divided into hyperbolic, parabolic and elliptical cameras, all have different imaging processes, making omnidirectional camera calibration a complex task. Fortunately, there is a unified sphere model [1] proposed by C'Mei which can be used to present fish-eye and all catadioptric cameras, even including perspective camera. It introduces a parameter $\xi$ to control the types of cameras. For instance, $\xi > 1$ is for fish-eye cameras, $\xi < 1$ is for catadioptric cameras. By making use of the unified sphere model, one can use the same code to work with all omnidirectional cameras.

A description of C'Mei model is given below:

Given a 3D point $X_w$ in world coordinate, first transform it to camera coordinate.

$$\mathbf{X}_c = \mathbf{R}\mathbf{X}_\mathbf{w} + \mathbf{T}$$

Next, map $X_c$ to sphere.

$$\mathbf{X}_\mathrm{s} = \frac{\mathbf{X}_\mathbf{c}}{||\mathbf{X}_\mathbf{c}||} = (X_s, Y_s, Z_s)$$

Then transform $\mathbf{X}_\mathrm{s}$ to a new reference frame centered at $O_q = (0,0,-\xi)$

$$\mathrm{x} = \left(\frac{X_s}{Z_s + \xi}, \frac{Y_s}{Z_s + \xi}, 1\right)$$

Next, add lens distortion

$$\tilde{x} = x + \tilde{d}$$

Finally, with an intrinsic matrix $\mathbf{K}$, the pixel location of this 3D point is

$$\mathrm{p} = \mathbf{K}\tilde{\mathrm{x}}$$

Note $\xi$ is also a parameter for calibration. The unified sphere model is given in Fig 2.

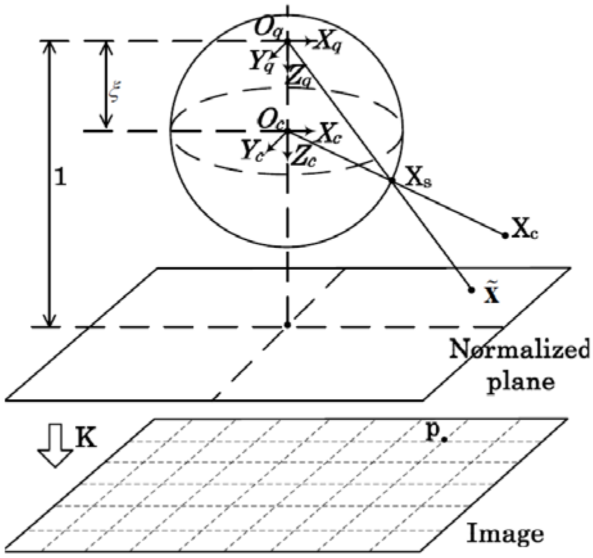


Fig 2. The unified sphere model

## 2. Stereo rectification of omnidirectional camera

To get 3D from one pair of stereo images, one should rectify them by camera parameters from calibration. The rectification method is also related with camera model. So the rectification should be also implemented.

The conventional rectification result is a perspective image so that horizontal lines in 3D space are horizontal on image, which restrict stereo matching on a single line. See Fig 3.

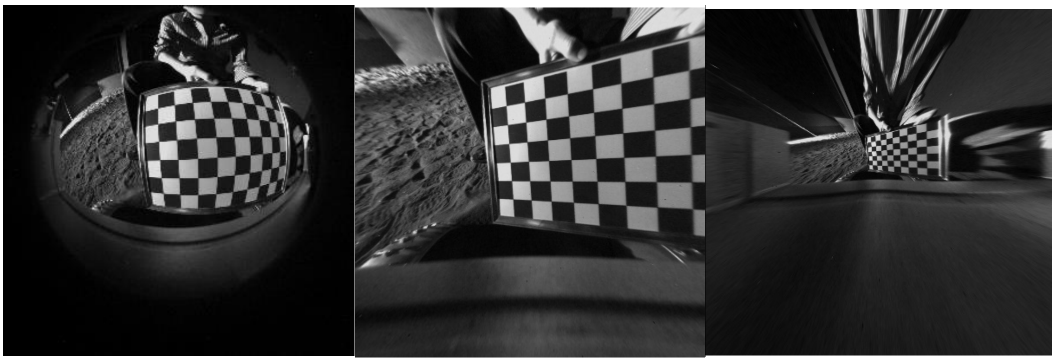

Fig 3. One fish-eye image (left), its perspective rectified image (center), 4 times scaled perspective rectified image (right)

It can be observed that perspective rectified image lost some boundary scene. One can scale the size of image to contain more scene. However, it becomes more difficulty when field of view is larger, when field of view is larger than 180, it is even impossible. The right figure of Fig 3 increase the image plane by 4 times, but still can preserve all details.

To deal with this, [2] propose a longitude-latitude mapping to rectify images. The main idea is that if one project 3D points on sphere, then horizontal lines in 3D space locate on latitude lines on sphere. So one can map the points on sphere to rectified image by their longitude and latitude, so that each pixel on rectified image represents one longitude-latitude pair. See Fig 4. Note that every points on sphere have longitude-latitude pair, the longitude-latitude mapping rectification supports 360° of field of view.

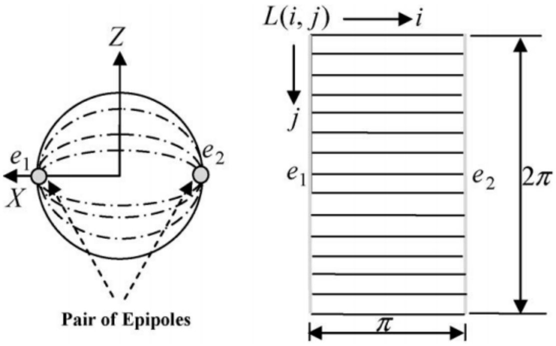


Fig 4. Explanation of longitude-latitude mapping

An example of longitude-latitude rectification for stereo images is given in Fig 5. Be aware that horizontal lines are mapped to latitude lines so that stereo matching can still be applied on single line, although it looks strange.

A new pair of fish-eye images and their disparity map is given in Fig 6.

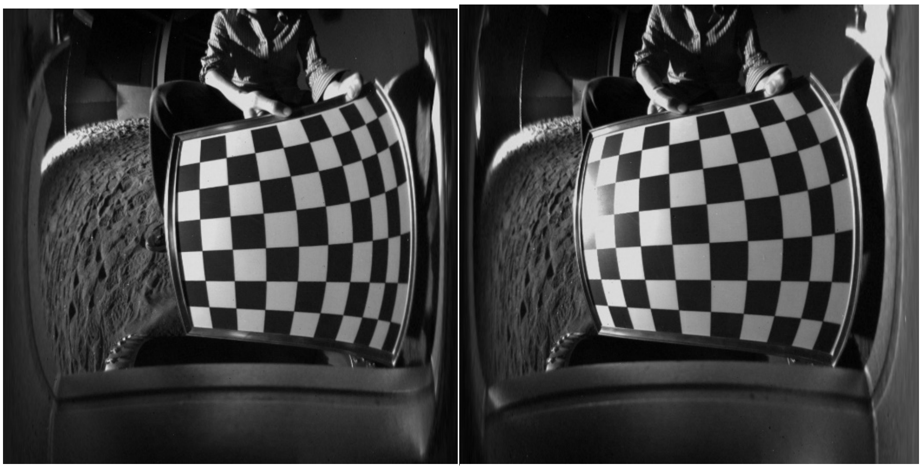

Fig 5. A pair of rectified images by longitude-latitude mapping

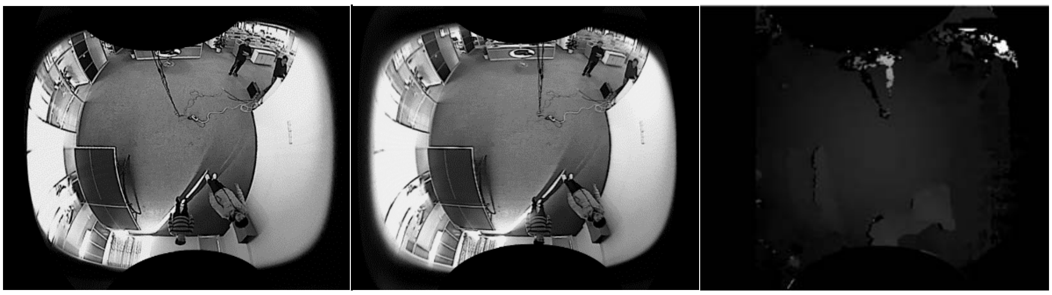

Fig 6. A pair of stereo rectified images and the disparity map

**Reference:**

[1] Mei C, Rives P. Single view point omnidirectional camera calibration from planar grids[C]//Robotics and Automation, 2007 IEEE International Conference on. IEEE, 2007: 3945-3950.

[2] Li S. Binocular spherical stereo[J]. Intelligent Transportation Systems, IEEE Transactions on, 2008, 9(4): 589-600.

[3] http://homepages.laas.fr/cmei/index.php/Toolbox

[4] https://sites.google.com/site/scarabotix/ocamcalib-toolbox

[5] http://apps.jcns.fz-juelich.de/doku/sc/lmfit

[6] Gong X, Lai B, Xiang Z. AL 0 sparse analysis prior for blind poissonian image deconvolution [J]. Optics express, 2014, 22(4): 3860-3865.
[7] Rufli M, Scaramuzza D, Siegwart R. Automatic detection of checkerboards on blurred and distorted images[C]//Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on. IEEE, 2008: 3121-3126.