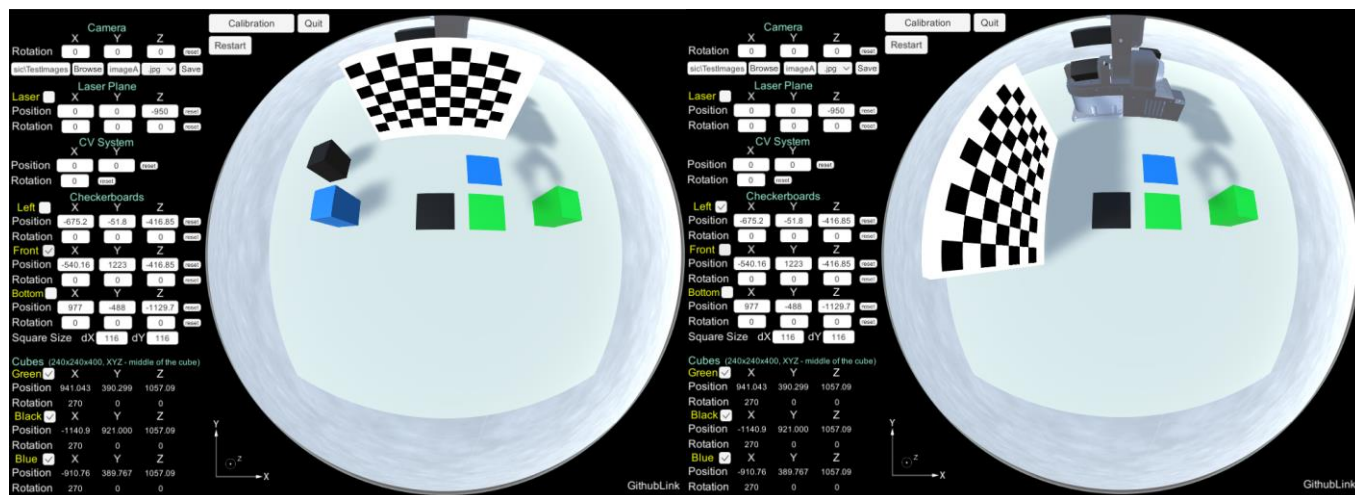


ПРОВЕРКА РЕЗУЛЬТАТОВ КАЛИБРОВКИ

В этом руководстве мы узнаем, как проводить проверку результатов внутренней калибровки со следующей конфигурацией системы компьютерного зрения:

- Камера не поворачивается;
- Нам нужно активировать мишень одну за другой и сделать их снимки.



Процедура калибровки

Прежде всего, вам необходимо откалибровать камеру, если вы еще этого не сделали — сначала перейдите к руководству по калибровке.

Подготовка

В этом руководстве есть несколько дополнительных файлов Matlab, которые должны быть включены в папку проекта:

- C_calib_data
- cam2world
- compose_rotation
- get_checkerboard_cornersUrban
- calibrate
- FindTransformMatrix

Основная программа

Сначала загрузим наши параметры калибровки и изображение в данные калибровки для дальнейшей работы.

```
clc
clear all
load('Omni_Calib_Results_116m.mat'); % Calib parameters
ocam_model = calib_data.ocam_model; % Calib parameters
i = calib_data.n_ima;
calib_data.L(i+1)={'TestImages/image.jpg'};
```

Далее нам нужно найти координаты пикселей нашей шахматной доски.

```
use_corner_find=1;
[callBack,Xp_abs_,Yp_abs_] = ...
    get_checkerboard_cornersUrban(i+1,use_corner_find,calib_data);
Xt = calib_data.Xt;
Yt = calib_data.Yt;
imagePoints = [Yp_abs_,Xp_abs_];
```

Далее мы можем выполнить процесс калибровки для нашего изображения. Идея здесь в том, что к уже существующим калибровочным изображениям мы добавили еще одно и проведем процесс калибровки для нашего изображения.

```
% first image extrinsic
[RRfin,ss]=calibrate(Xt, Yt, Xp_abs_, Yp_abs_, ocam_model);
RRfin_=FindTransformMatrix(Xp_abs_, Yp_abs_, Xt, Yt, ocam_model, RRfin);
```

После калибровки у нас есть матрица перехода между нашей камерой и предоставленной шахматной доской. Из матрицы перехода мы можем взять интересующее нас расстояние для использования (расстояние от камеры до шахматной доски).

```
% find distance
Y1 = RRfin_( ? )
```

Для второго изображения мы используем тот же код

```
%% second image
i = calib_data.n_ima;
calib_data.L(i+1)={'TestImages/image1.jpg'};
use_corner_find=1;
[callBack,Xp_abs_,Yp_abs_] = ...
    get_checkerboard_cornersUrban(i+1,use_corner_find,calib_data);
Xt = calib_data.Xt;
Yt = calib_data.Yt;
imagePoints = [Yp_abs_,Xp_abs_];
% first image extrinsic
[RRfin,ss]=calibrate(Xt, Yt, Xp_abs_, Yp_abs_, ocam_model);
RRfin_=FindTransformMatrix(Xp_abs_, Yp_abs_, Xt, Yt, ocam_model, RRfin);
% find distance
Y2 = RRfin_( ? )
```

Теперь мы можем сравнить значения:

	Левая мишень, мм	Вверхняя мишень, мм
Реальные значения	675.2	1223.0
Экспериментальные значения	672.4	1222.9
Погрешность	2.8	0.1