HOMEWORK 18

Camera Calibration

Student Name: Rohit Das **Student ID**: 61047086s

Objective: Calculate the equivalent distance of the real world in 1 pixel of the digital camera for each:

- Focal length
- Object Displacement
- Object Distance

Algorithm Used: Traditional Algorithm

- Find the ROI (Region of Interest)
- Get Displacement pixel values, Displacement
- Compare with the Theoretical Equation

Screenshot of the Algorithm:

<u>Get Displacement Pixel values:</u>

```
int getDisplacementPixel(Mat& obj, Point2i obj_loc, Mat& img)
{
    imgSet img_dataset = getBlock(img, obj_loc, obj.rows, obj.cols);
    double loss = numeric_limits<double>::max();
    Point2i matchPoint;
    double search_range = 600.0;

    for (int i = 0; i < img_dataset.size(); ++i)
    {
        double dis = img_dataset[i].first.x - obj_loc.x;
        if (dis <= search_range)
        {
            Mat diff;
            absdiff(obj, img_dataset[i].second, diff);
            double current_loss = sum(diff)[0] + sum(diff)[1] + sum(diff)[2];
            if (current_loss < loss)
            {
                loss = current_loss;
                matchPoint = img_dataset[i].first;
            }
        }
    }

    return matchPoint.x - obj_loc.x;
}
```

Theoretical Formula

```
double theoreticalFOV(int f)
{
    return 2 * atan(sensor_width / 2 / f) * 180 / M_PI;
}

double mmPrePixel(int d, int p)
{
    return (double)d / p;
}

double measuredFOV(int img_width, int d, double mm_pre_pixel)
{
    return 2 * atan(img_width * mm_pre_pixel / 2 / d) * 180 / M_PI;
}
```

The output table:

Focal length(mm)	Object distance(mm)	Object actual displacement(mm)	Object displacement(pixel)	mm/pixel	FOV theoretical value(degree)	FOV measured value(degree)
18	600	1	7	0.142857	66.0477	58.1648
18	600	5	28	0.178571	66.0477	69.617
18	600	10	56	0.178571	66.0477	69.617
18	600	20	106	0.188679	66.0477	72.6014
18	1200	1	2	0.5	66.0477	88.4516
18	1200	5	16	0.3125	66.0477	62.6271
18	1200	10	30	0.333333	66.0477	65.9582
18	1200	20	56	0.357143	66.0477	69.617
18	1800	1	3	0.333333	66.0477	46.7859
18	1800	5	9	0.555556	66.0477	71.5823
18	1800	10	19	0.526316	66.0477	68.6694
18	1800	20	45	0.444444	66.0477	59.9518
53	600	1	22	0.045455	24.8973	20.0714
53	600	5	64	0.078125	24.8973	33.836

53	600	10	134	0.074627	24.8973	32.4022
53	600	20	275	0.072727	24.8973	31.6191
53	1200	1	9	0.111111	24.8973	24.4097
53	1200	5	44	0.113636	24.8973	24.9473
53	1200	10	78	0.128205	24.8973	28.0264
53	1200	20	162	0.123457	24.8973	27.0271
53	1800	1	12	0.083333	24.8973	12.3449
53	1800	5	37	0.135135	24.8973	19.8942
53	1800	10	59	0.169492	24.8973	24.8107
53	1800	20	102	0.196078	24.8973	28.5537
135	600	1	61	0.016393	9.90651	7.30391
135	600	5	164	0.030488	9.90651	13.5386
135	600	10	268	0.037313	9.90651	16.5315
135	600	20	510	0.039216	9.90651	17.3617
135	1200	1	37	0.027027	9.90651	6.02341
135	1200	5	62	0.080645	9.90651	17.844
135	1200	10	151	0.066225	9.90651	14.6919
135	1200	20	317	0.063092	9.90651	14.0038
135	1800	1	3	0.333333	9.90651	46.7859
135	1800	5	32	0.15625	9.90651	22.9258
135	1800	10	88	0.113636	9.90651	16.7784
135	1800	20	212	0.09434	9.90651	13.9602