Boqian Yu 03708925

1 Exercise 1

`test_ex2.cpp` tests the camera classes (`PinholeCamera`, `ExtendedUnifiedCamera`, `DoubleSphereCamera` and `KannalaBrandt4Camera`) with 21 × 21 points to see if the `project` and `unproject` functions are correctly implemented by comparing the original coordinates with coordinates after `project` and `unproject`.

2 Exercise 2

Curve-fitting uses a **CostFunction** to define the error of each observation. Moreover, robust curve-fitting can deal with outliers by introducing a **LossFunction** to reduce the influence of residual blocks with high residuals (usually corresponding to outliers), thus can give more robust results.

3 Exercise 3

Command line parameters:
`--show-gui`: Specify if the GUI should be shown (default) or not.
`--dataset-path`: Specify the dataset path, which is a requirement to run the program.
`--cam-model`: Specify the camera model to be used. Possible values are `pinhole`, `ds`, `eucm`, `kb4`, and the default value is `ds`.
Calibration results on different camera models:
The results of calibration on different camera models are listed in Table.1 with the following criteria: **Final cost** (remaining error after calibration), **Iterations** (number of iterations taken before convergence), **Running time** (time consumed for the whole calibration process)
Among these criteria, the final cost is the most appropriate measure to determine how well the camera models fit the lenses, because it directly reflects the quality of the calibration. It can be seen that the pinhole model yields much worse results than the other three models, and that the other three models can all give similar (and acceptable) results.

Table 1: Calibration results on different camera models

Measures	pinhole	ds	eucm	kb4
Final cost	156573.5	162.7482	162.7604	161.9844
Iterations	16	15	7	8
Running time	1.015199	0.874160	0.493556	0.558785