Project 4
Grading Policy

Sung Soo Hwang

Perspective Transformation

- findHomography() function

◆ findHomography() [1/3]

```
Mat cv::findHomography ( InputArray  srcPoints,
                         InputArray  dstPoints,
                         int         method = 0 ,
```

Parameters

srcPoints	Coordinates of the points in the original plane, a matrix of the type CV_32FC2 or vector<Point2f> .
dstPoints	Coordinates of the points in the target plane, a matrix of the type CV_32FC2 or a vector<Point2f> .
method	Method used to compute a homography matrix. The following methods are possible: • **0** - a regular method using all the points, i.e., the least squares method • **RANSAC** - RANSAC-based robust method • **LMEDS** - Least-Median robust method • **RHO** - PROSAC-based robust method

Perspective Transformation

- findHomography() function

◆ findHomography() [1/3]

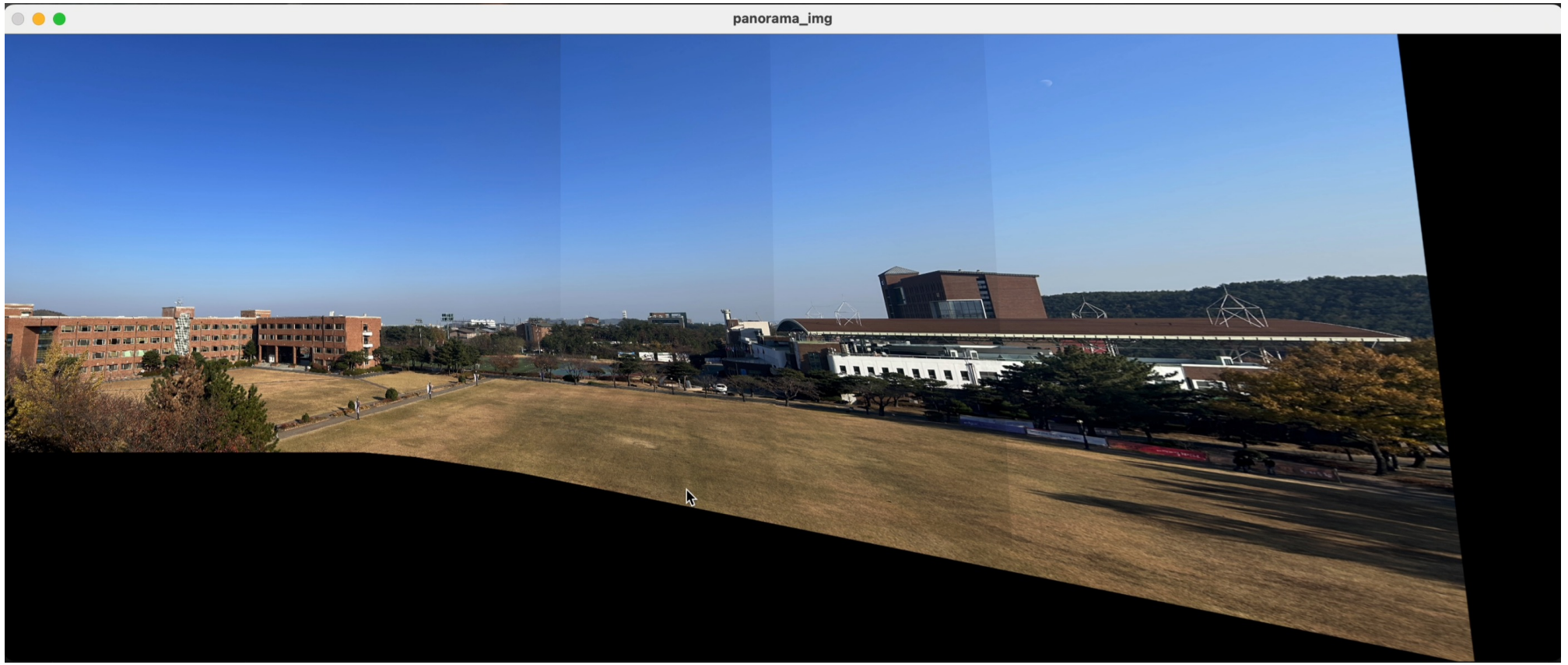
```
Mat cv::findHomography ( InputArray   srcPoints,
                         InputArray   dstPoints,
                         int          method = 0 ,
```

- This function is used to find the homography matrix between two sets of points. A homography is a transformation (a 3x3 matrix) that maps the points in one image to the corresponding points in another image.

- The function calculates the best homography that maps the points in one image to the corresponding points in another image, often using methods like RANSAC or Least Median to handle outliers and provide a robust solution.

Purpose of Project 4

- Given 4 images, generate a panoramic image



Grading policy

- Finish project 4 using functions provided in the lecture only: 10 pts
  - Including findHomgraphy()
- Finish project 4 using extra functions not provided in the lecture
  - -1 pt whenever you use the function
- Fail to finish project 4 on time: 0 pt