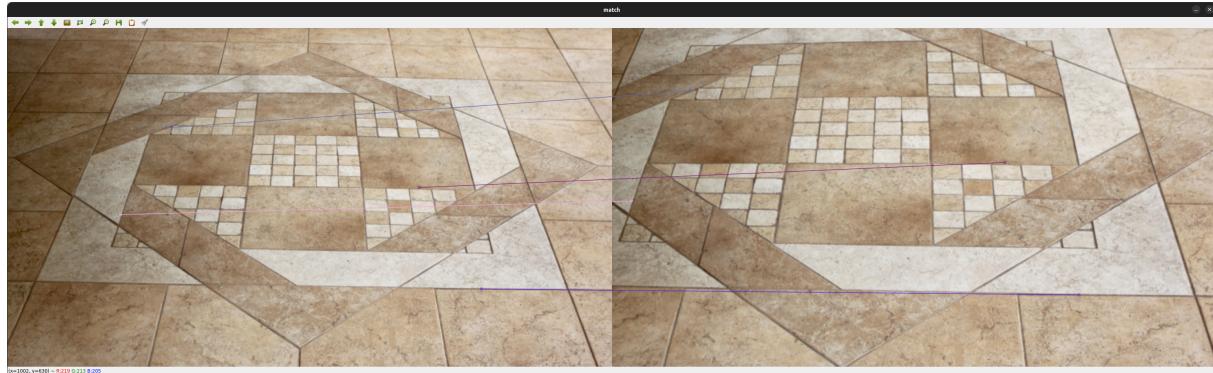


3dcv hw1

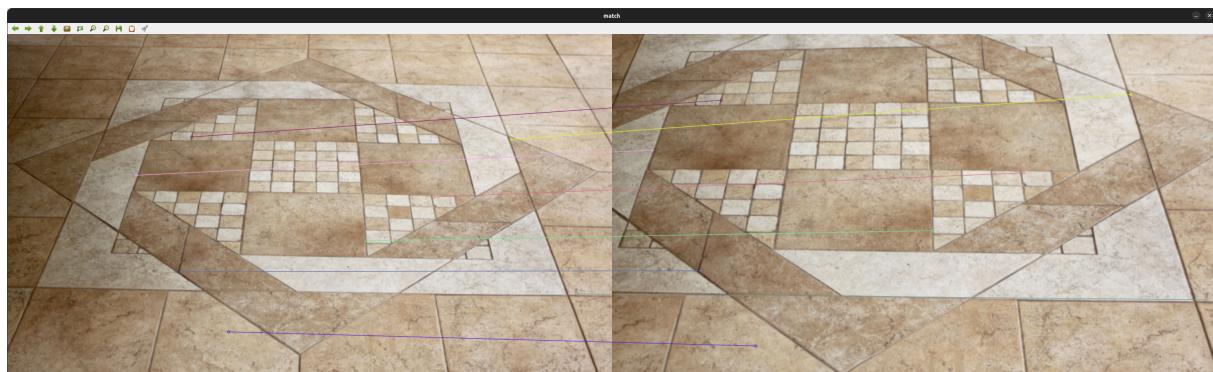
余承諺 r12922135

2023 / 10 / 2

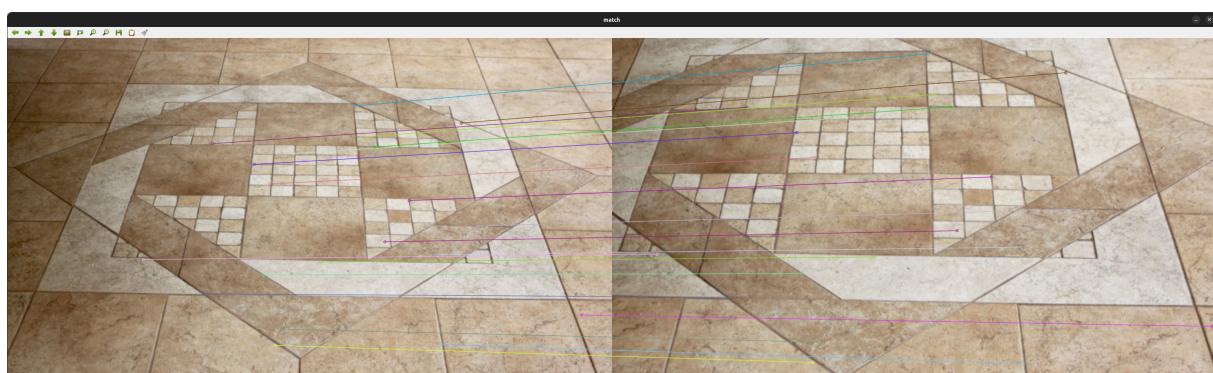
Problem 1 - Screenshots



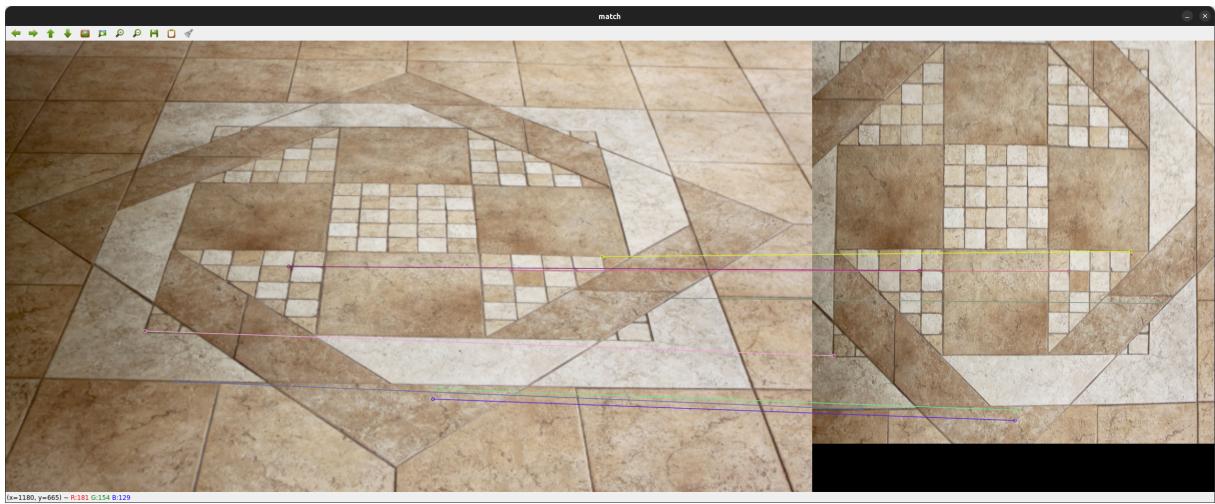
DLT from images/1-0.png to images/1-1.png with 4 correspondences.



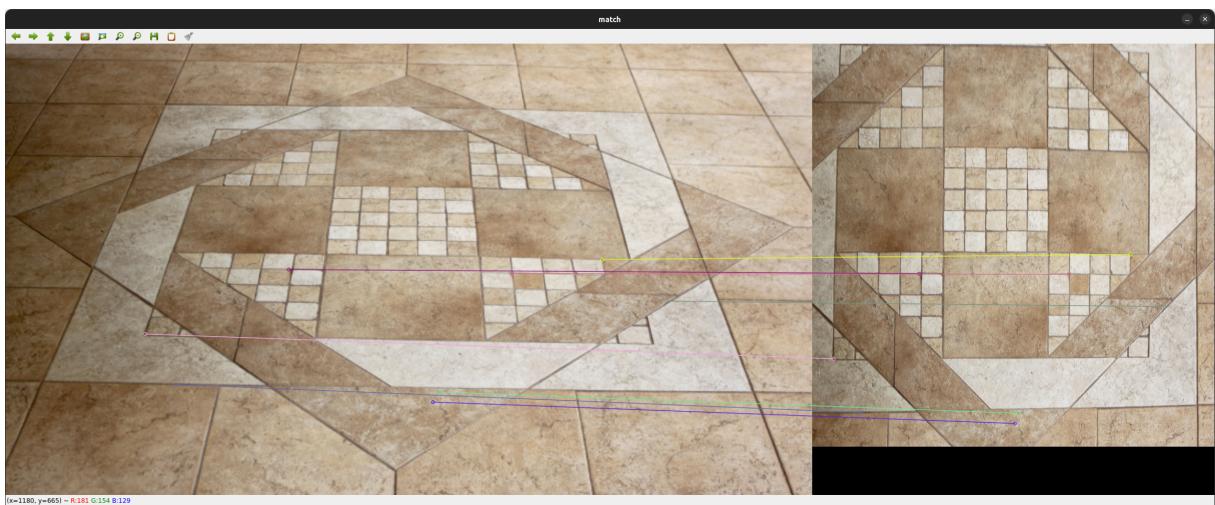
DLT from images/1-0.png to images/1-1.png with 8 correspondences.



DLT from images/1-0.png to images/1-1.png with 20 correspondences.



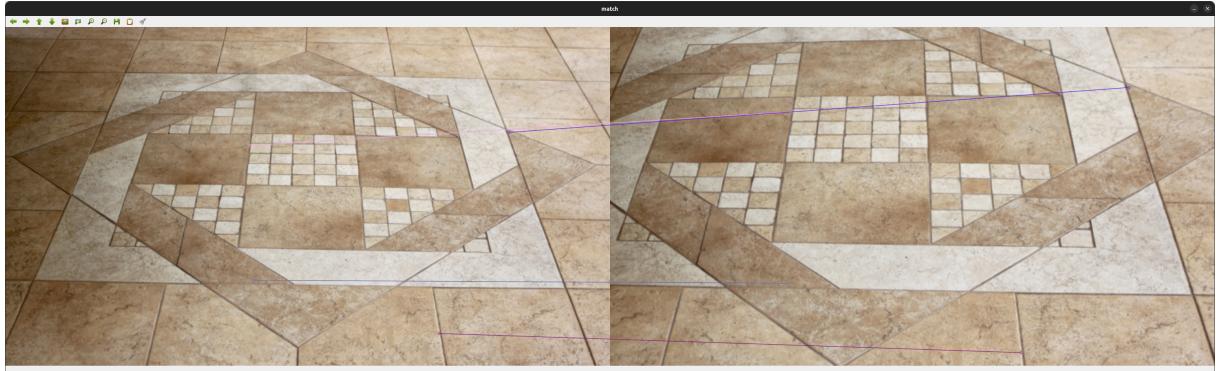
DLT from images/1-0.png to images/1-2.png with 4 correspondences.



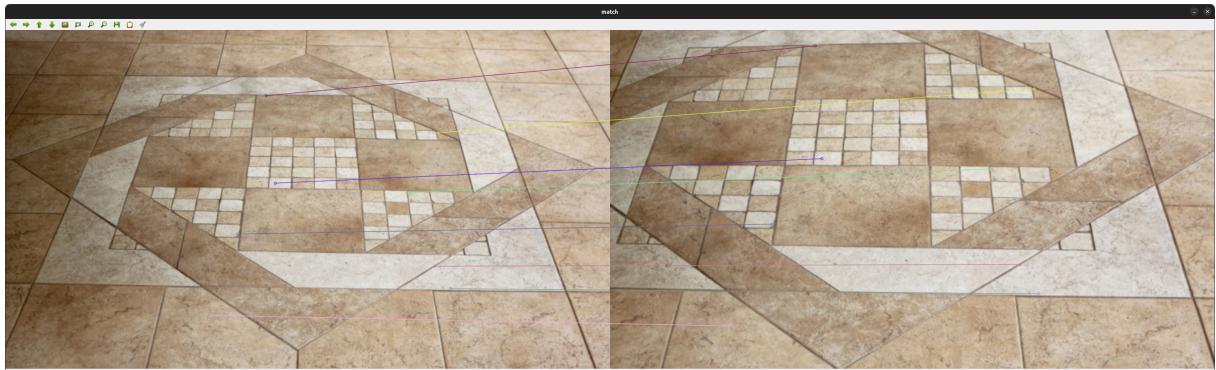
DLT from images/1-0.png to images/1-2.png with 8 correspondences.



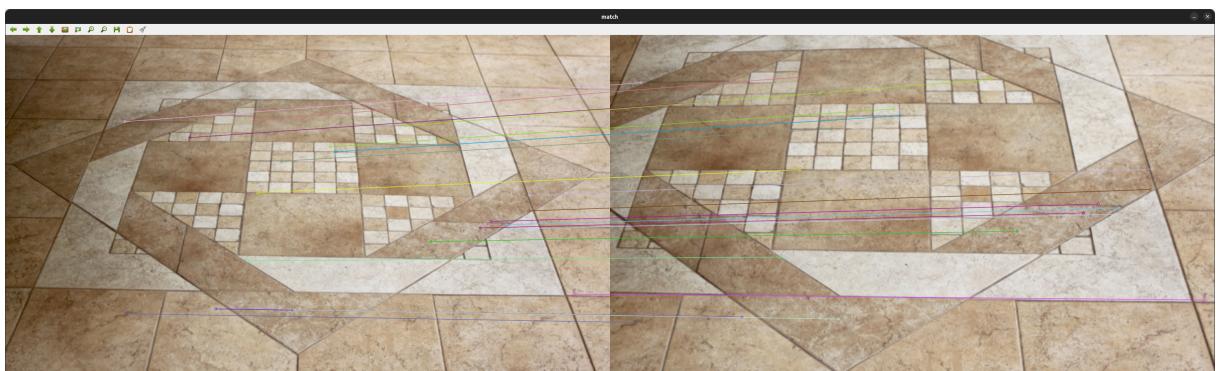
DLT from images/1-0.png to images/1-2.png with 20 correspondences.



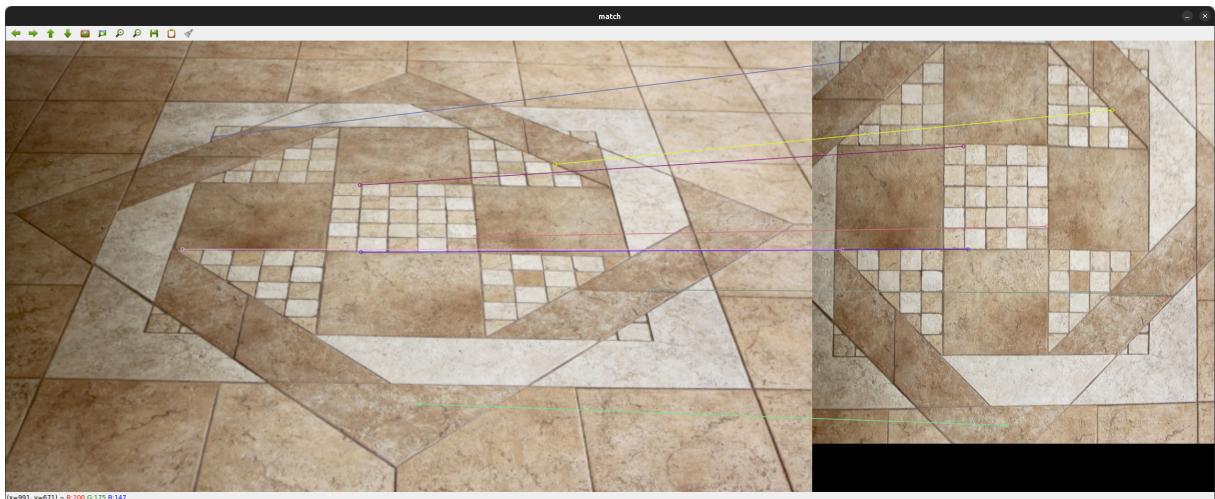
Normalized DLT from images/1-0.png to images/1-1.png with 4 correspondences.



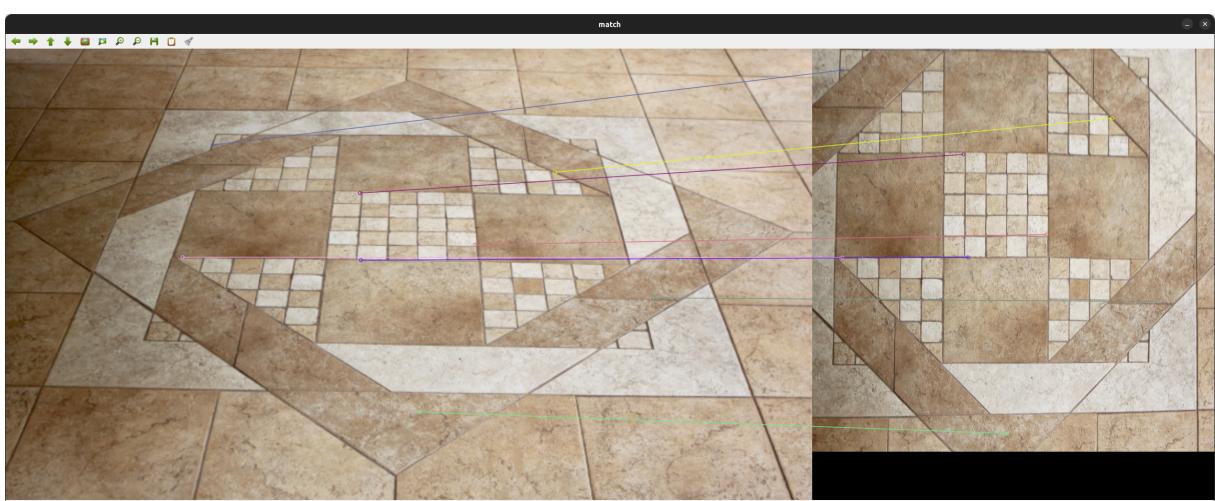
Normalized DLT from images/1-0.png to images/1-1.png with 8 correspondences.



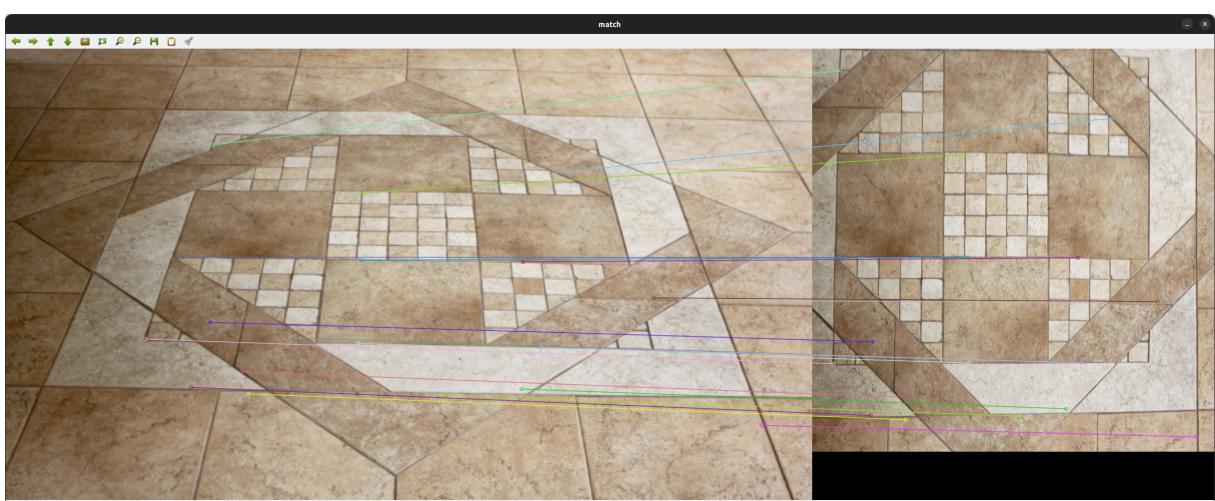
Normalized DLT from images/1-0.png to images/1-1.png with 20 correspondences.



Normalized DLT from images/1-0.png to images/1-2.png with 4 correspondences.



Normalized DLT from images/1-0.png to images/1-2.png with 8 correspondences.



Normalized DLT from images/1-0.png to images/1-2.png with 20 correspondences.

Problem 1 - Compares the errors

The following tables are errors under different ks and DLT / normalized DLT.

		DLT	Normalized DLT
k	4	73.4e-3	204e-6, 67.1e-3
	8	69.3e-3	237e-6, 62.6e-3
	20	41.0e-3	168e-6, 55.2e-3

These homography estimation errors in this table are from images/1-0.png to images/1-1.png.
The column of Normalized DLT show reprojected errors and errors, separated by comma.

		DLT	Normalized DLT
k	4	431e-3	2.20e-3, 401e-3
	8	408e-3	1.72e-3, 317e-3
	20	404e-3	1.45e-3, 279e-3

These homography estimation errors in this table are from images/1-0.png to images/1-2.png.
The column of Normalized DLT show reprojected errors and errors, separated by comma.

Problem 1 - Discussion

SIFT method is astonishing. As a traditional algorithm without deep learning, it also has high accuracy of find correspondences. Although the SIFT method is good, some matches are wrong sometimes. A points corresponds to another point with similar texture around it. This will increase the errors of homography estimation especially under small k.

Problem 1 - Execution

”1-1.py” is aim to obtain the idxes of correspondences which generate less err.

```
python 1-1.py <img1_path> <img2_path> <gt_path> <normalized> <k>
```

img1_path: img1 path.

img2_path: img2 path.

gt_path: groundtruth correspondences path, should be a .npy file.

normalized: use normalized DLT or DLT, should be True or False.

k: the number of correspondences used to homography estimation.

”1.py” is aim to show correspondences and use them to estimate homography.

```
python 1.py <img1_path> <img2_path> <gt_path> <normalized> <idxes>
```

img1_path: img1 path.

img2_path: img2 path.

gt_path: groundtruth correspondences path, should be a .npy file.

normalized: use normalized DLT or DLT, should be True or False.

idxes: the idxes of correspondences used to homography estimation.

Problem 2

The main executor is "2.py" which is designed to receive some arguments.

```
python 2.py <img_src_path> <img_dst_path> <img_dst_height> <img_dst_width>
```

img_src_path: The path of input image read from.

img_dst_path: The path of output image save to.

img_dst_height: The height of output image.

img_dst_width: The width of output image.

After executing the command, 2.py will open a window showing the input image from <img_src_path>. Then user can dot 4 points: left-top, right-top, right-bottom and left-bottom sequentially (clockwise starting from left-top).

After pressing "ESC" to enter the 4 points, 2.py will rectify the range closed by these 4 points to <img_dst_height> × <img_dst_width> then save the result to <img_dst_path> and show it by opening another window.

Youtube Link: https://youtu.be/0TmYZzV_Dv4