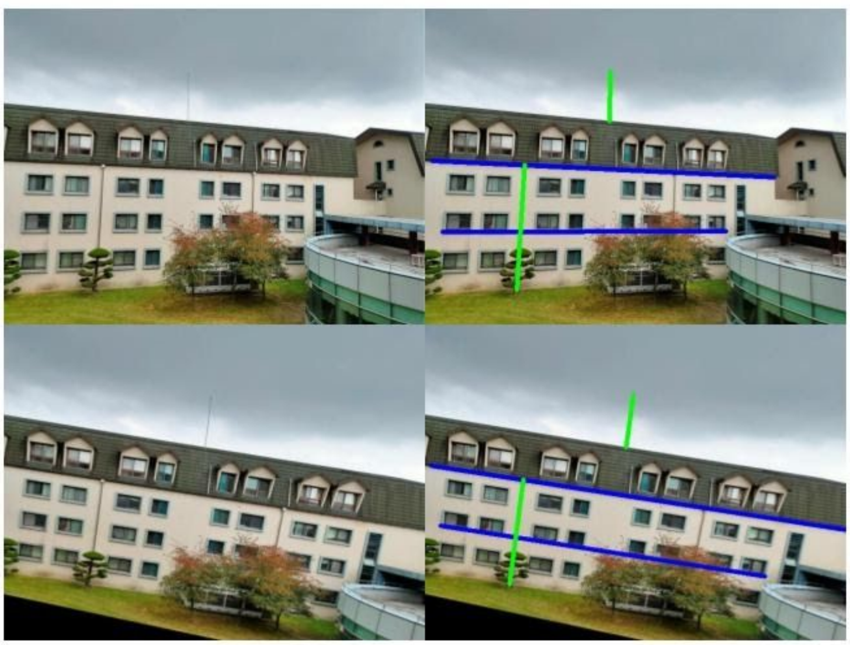
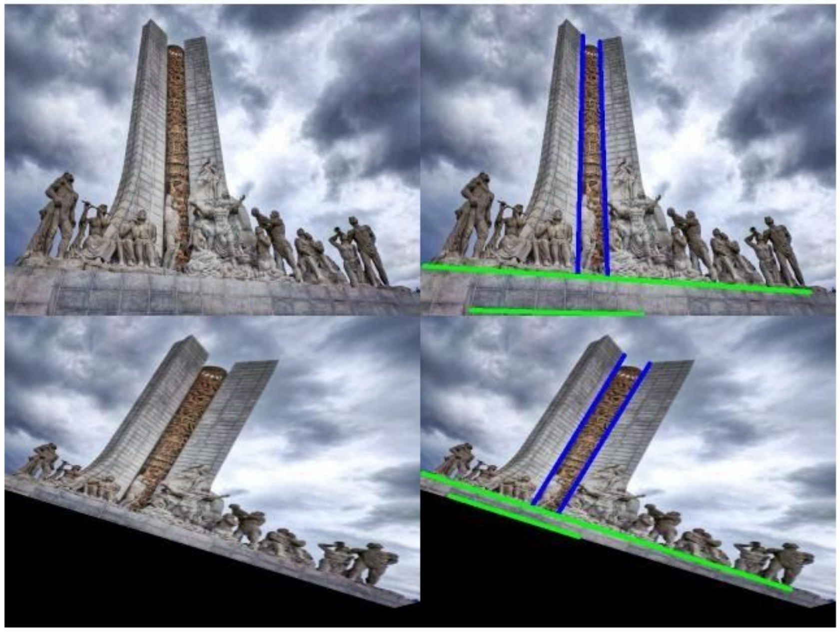
Jedsadakorn Yonchorhor
20194695
Assignment 1

Assignment 1-1: Image Rectification

- Two-Step Method
    - First Step: Perspective Rectification
        - Recover the parallel part of the image by projective transformation
        - The physically parallel objects can appear to be not parallel due to the perspective distortion as shown in the figure below. The green lines should be parallel to each other, as well as the blue lines. However, because of perspective distortion, they intersect at some points, called vanishing points.
        - From the two vanishing points, we can form a vanishing line, l, passing through those two points.
        - We want to make the blue lines and green lines parallel, therefore, the vanishing line needs to be transformed to the ideal line, using the H1 matrix which can be constructed using the vanishing line l.
        - **The result** of the transformation is the second row of each figure. It can be seen that both pairs of the lines become **parallel** lines.

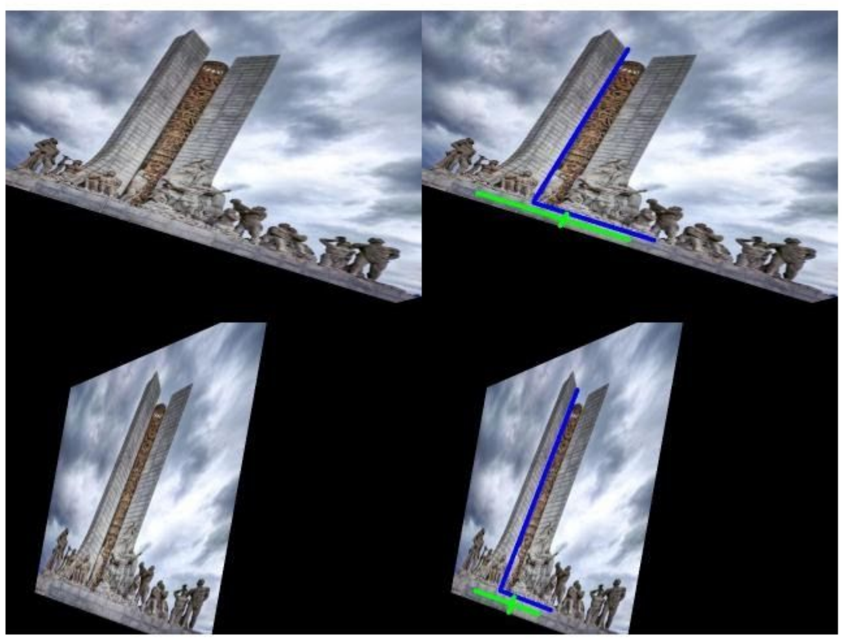


Jedsadakorn Yonchorhor
20194695
Assignment 1



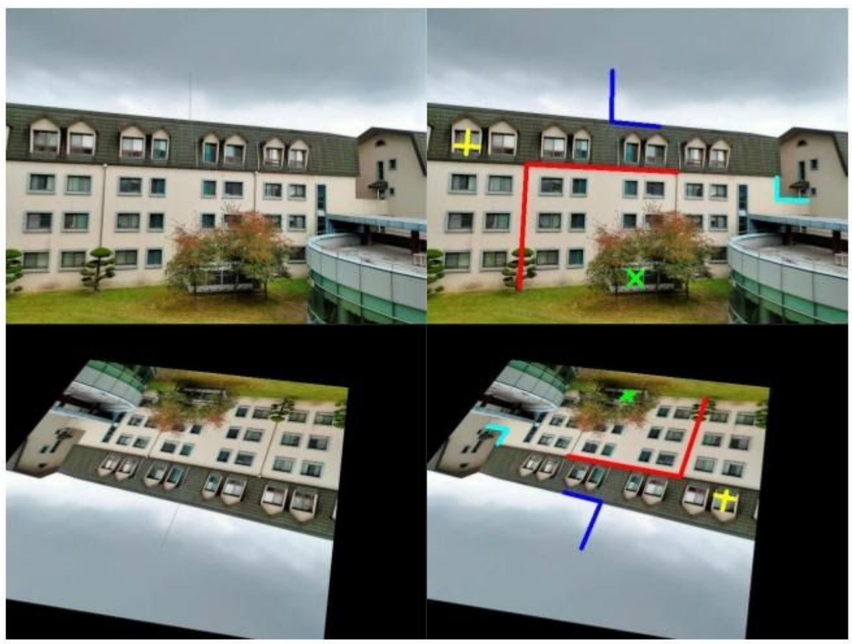
- Second Step: Affine Rectification
    - After the first step, now the images are perspective rectified.
    - However, some parts of the image that should be physically perpendicular to each other are still not. This is because it is not affinely rectified.
    - Therefore, two pairs of perpendicular lines are chosen to form the equation to solve for the affine transformation, the H2 matrix.
    - After the procedure of solving the equations and SVD decomposition, the inverse of the H2 matrix is apply to the result images from the step one to get the rectified images.
    - **The result** of the transformation is the second row of each figure. The figure shows that both pairs of lines become perpendicular lines.

Jedsadakorn Yonchorhor
20194695
Assignment 1

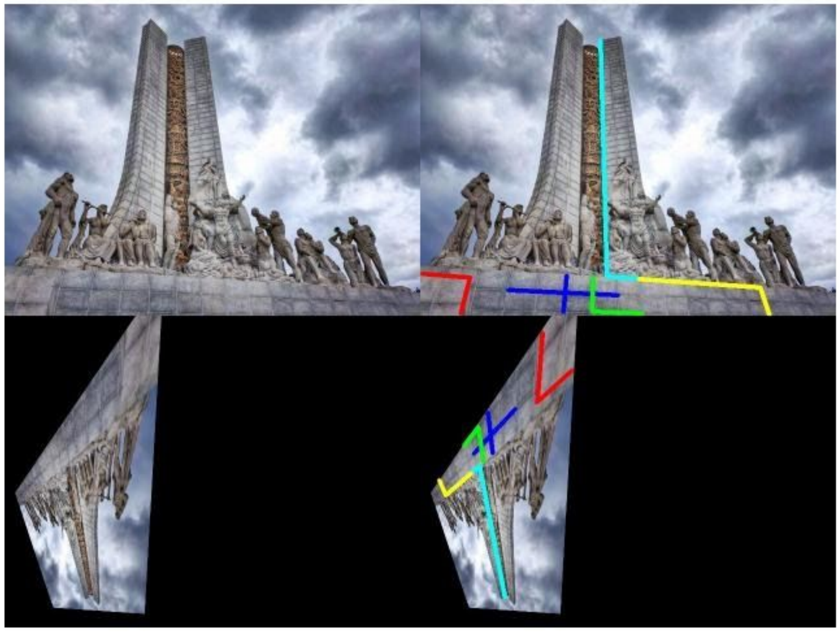


- One-Step Method
    - One step method does not separate the rectification process into two phases like the above method. It will directly rectify the image after we finish choosing five pairs of (physically) perpendicular lines (shown in different colors in the figures below).
    - The five pairs of perpendicular lines form the system equation (Ac = 0), in which the answer of that homogeneous equation is used to form the conics C*.

- The result conics C* is then decomposed using SVD decomposition to get U, S and V components.
- Then, the transformation matrix H is constructed as the following
  H = u @ np.diag([s[0] ** 0.5, s[1] ** 0.5, scale]).
- In the implementation, almost all of the time, the image will be transformed out of the frame. Therefore, I add translation matrix M to bring it back to the frame for visibility. The scale is adjusted so that the image can be seen in the frame.
- The results of the transformation are shown as the second row of each picture. However, we can see that the result is NOT as good as the two step method. The light-blue color pair is not perpendicular in the first image and the red color pair is not perpendicular in the second image.



Jedsadakorn Yonchorhor
20194695
Assignment 1



Question: Why is one-step method unstable?
Answer: The homography matrix is constructed from the following formula,

H = u @ np.diag([s[0] ** 0.5, s[1] ** 0.5, scale]),

which is the product of SVD decomposition of the C*. Ideally, the singular values of C* should have only two non-zero values, because it is the degenerate conics. However, in practice, singular value of C* usually have three non-zero singular values, with the third one significantly smaller than the other two. The scale needs to be adjusted to address this problem. But it is not trivial to do so. Therefore, it is not stable.

Jedsadakorn Yonchorhor
20194695
Assignment 1

Assignment 1-2: Automatic Homography using RANSAC

- Automatic Homography using RANSAC is used to stitch the three images (from pure rotation of the camera) together. Therefore, I will use the second image as the core image and transform the first and the third image so that it can be seamlessly connected with the second image.
- The features of each image are extracted. The ORB feature is used in my implementation because it is fast to be extracted and it is also invariant to rotation and scale. The orb keypoints are shown in the following figure.

- The ORB features at each position in the image will have the descriptor of length 32. We try to match ORB features in the second image to the first and the third image using Sum of the Square Distance (SSD) metric, in which all of the descriptors in the second images are compared with all of the descriptors in the first and the third image. Each of the ORB features in the second images are then paired with the ORB feature, which has the minimum SSD, in the first and third image.
- After we get all the corresponding pairs of ORB features from the second images to the rest, the Homography matrix, H, is computed using RANSAC and normalized DLT.
- After the RANSAC iterations terminate, all of the inliers are then used to compute the homography matrix for more accurate results.
- Since it is likely that the image will be transformed out of the frame, I extend the frame and also apply the translation matrix similar to the problem of assignment 1-1.
- The result of the stitched images is shown in the following figure.

Jedsadakorn Yonchorhor
20194695
Assignment 1