

Project 3 Part 1 - Image Warping and Mosaicing

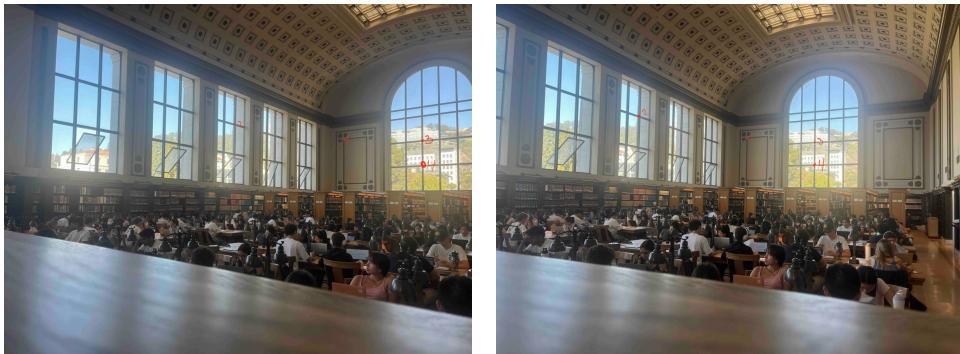
A.1

Here we see two sets photos where the transforms between them are projective. They are shot with a (reasoably) fixed center of proejction and rotation of the camera.





A.2 For each set of images, we can recover a homography between them by manually first selecting matching points between the images. For example, 4 cooresponding points in the first two images in the library set above are marked below in red. A minimum of 4 points is required to calculate a homography matrix.



To solve for our homography matrix, we can establish a system of equations. For a single correspondence mapping points (x,y) to (u,v) , we obtain the following system of equations where \mathbf{h} is the flattened homography matrix. With 4 point correspondences and 8 unknowns in the homography matrix, we're able to solve for \mathbf{h} .

$$\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -ux & -uy \\ 0 & 0 & 0 & x & y & 1 & -vx & -vy \end{bmatrix} \mathbf{h} = \begin{bmatrix} u \\ v \end{bmatrix}$$

Specifically, for the two images above,

$$\mathbf{A} = [[873, 346, 1, 0, 0, 0, -312534, -123868],$$

$$[0, 0, 0, 873, 346, 1, -288090, -114180],$$

$$[873, 458, 1, 0, 0, 0, -309915, -162590],$$

$$[0, 0, 0, 873, 458, 1, -384993, -201978],$$

$$[930, 339, 1, 0, 0, 0, -382230, -139329],$$

$$[0, 0, 0, 930, 339, 1, -307830, -112209],$$

$$[929, 457, 1, 0, 0, 0, -379032, -186456],$$

$$[0, 0, 0, 929, 457, 1, -410618, -201994]]$$

$$\mathbf{h} = [h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8]$$

$$\mathbf{b} = [358, 330, 355, 441, 411, 331, 408, 442]$$

When we solve this system of equations, we solve

$$\mathbf{H} = [[2.51425309e+00 \ -3.55347070e-02 \ -1.42755447e+03]$$

$$[6.20890951e-01 \ 2.12595382e+00 \ -5.78868787e+02]$$

$$[1.24765224e-03 \ 6.88009940e-05 \ 1.00000000e+00]]$$

A.3 With the homography matrix we can warp the images from 1 plane to another. We will do so using inverse warping to avoid holes on the output images. Below are the two examples above, where the left and right image are warped to match the plane of the first. We compare the difference in warping using nearest neighbor interpolation and bilinear interpolation. With the naked eye it is hard to see a difference in the methods.

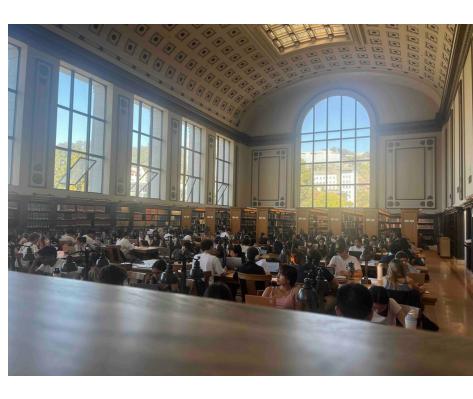
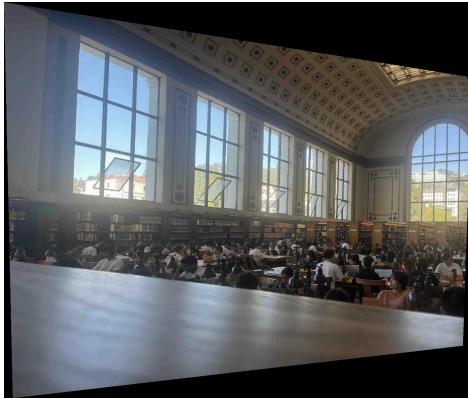
Neighborhood scene with nearest neighbor interpolation:



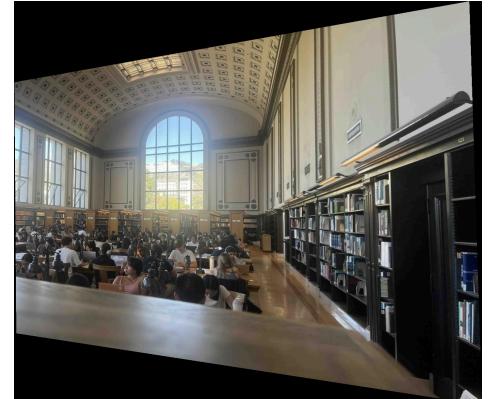
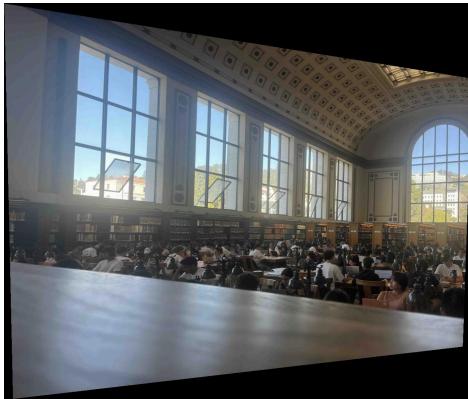
Neighborhood scene with bilinear interpolation:



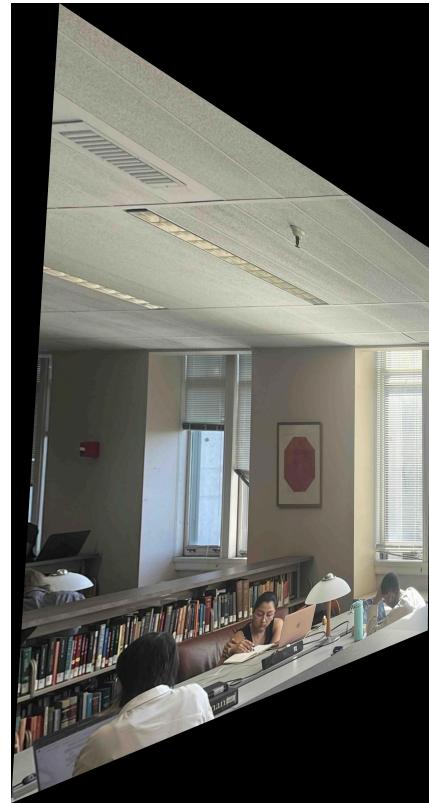
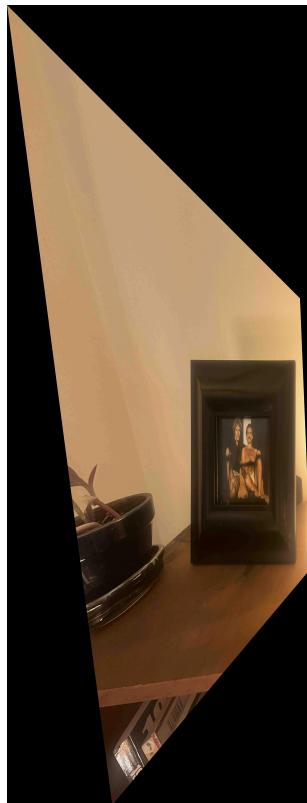
Library scene with nearest neighbor interpolation:



Library scene with bilinear interpolation:



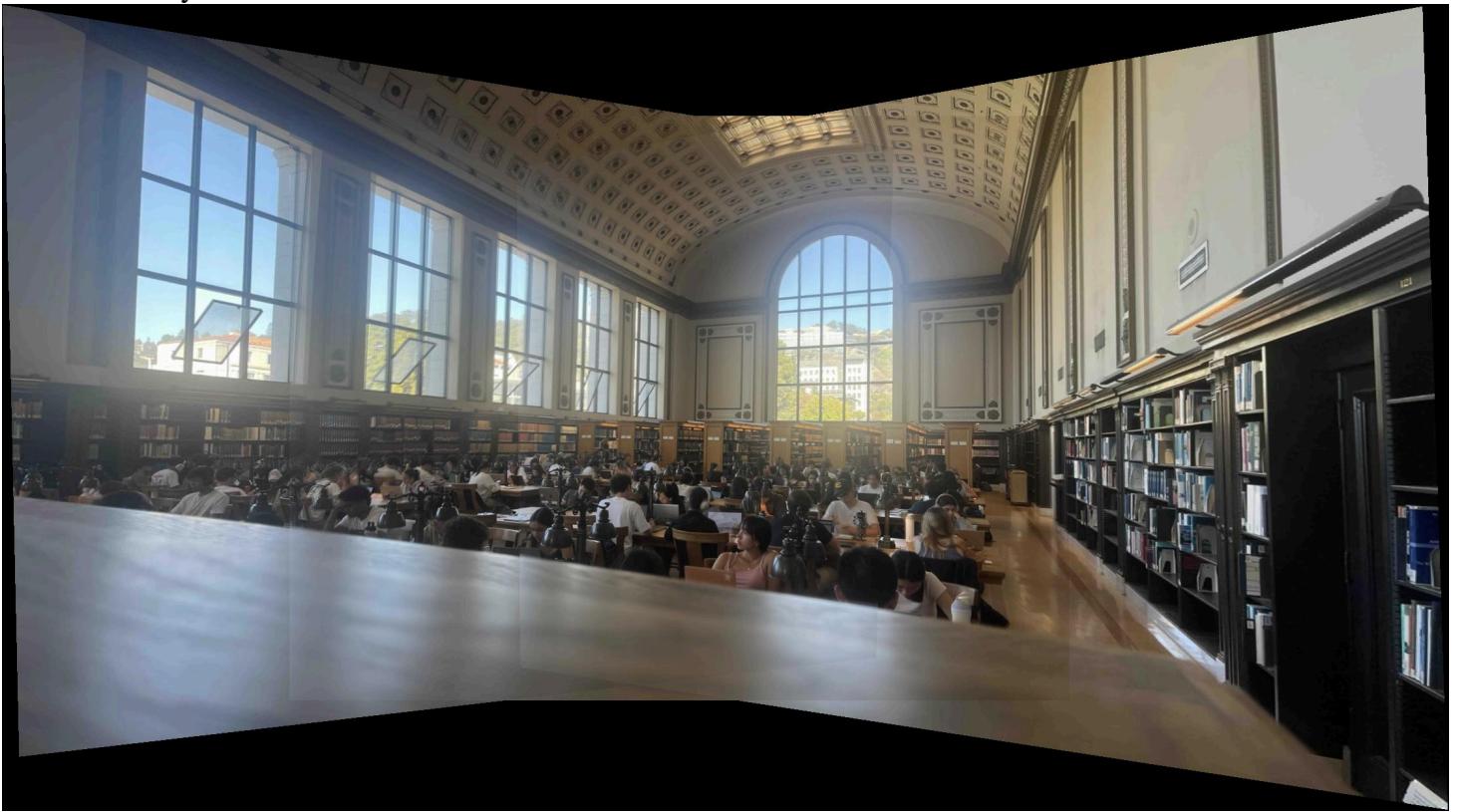
To check this method is working properly, we can rectify images. For example, if we take a skewed photograph of a known rectangular object and calculate a homography for fitting it to a rectangle, we can examine the results. Here are two rectified photos, showing the homography transformation is working as expected:



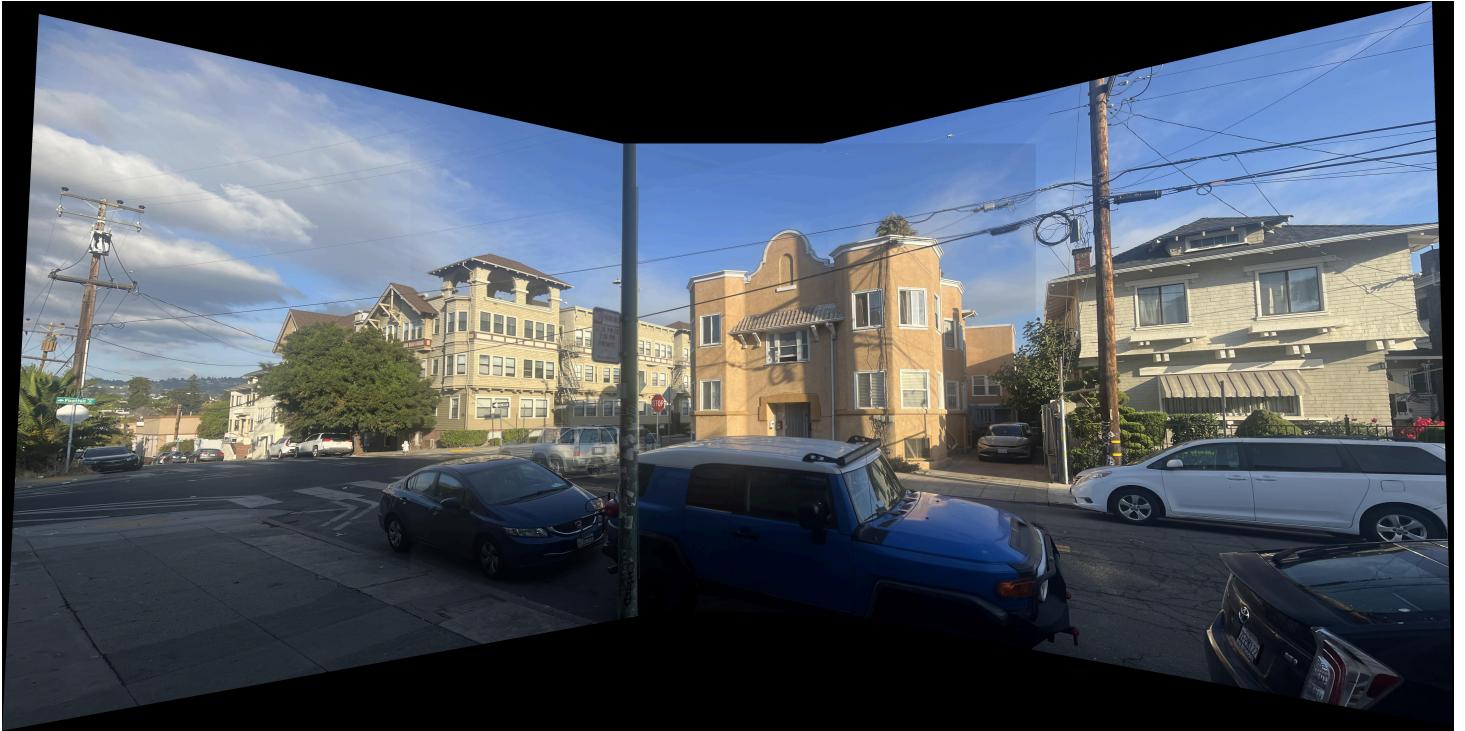
A.4 After the images can be matched to a single plane, they can be blended into a mosaic. I do this by mapping the left image into the plane of the center image and creating a canvas that fits both images. The center image must be mapped to the canvas by some translation, given by T . Thus, if the left image is mapped to the center image's plane by H , it is mapped to the canvas by $T @ H$. A similar procedure is used to map the right image to the canvas.

Furthermore, a weighted average mask is applied to each image, taking only half the information from the overlapping areas to ensure smoother blending.

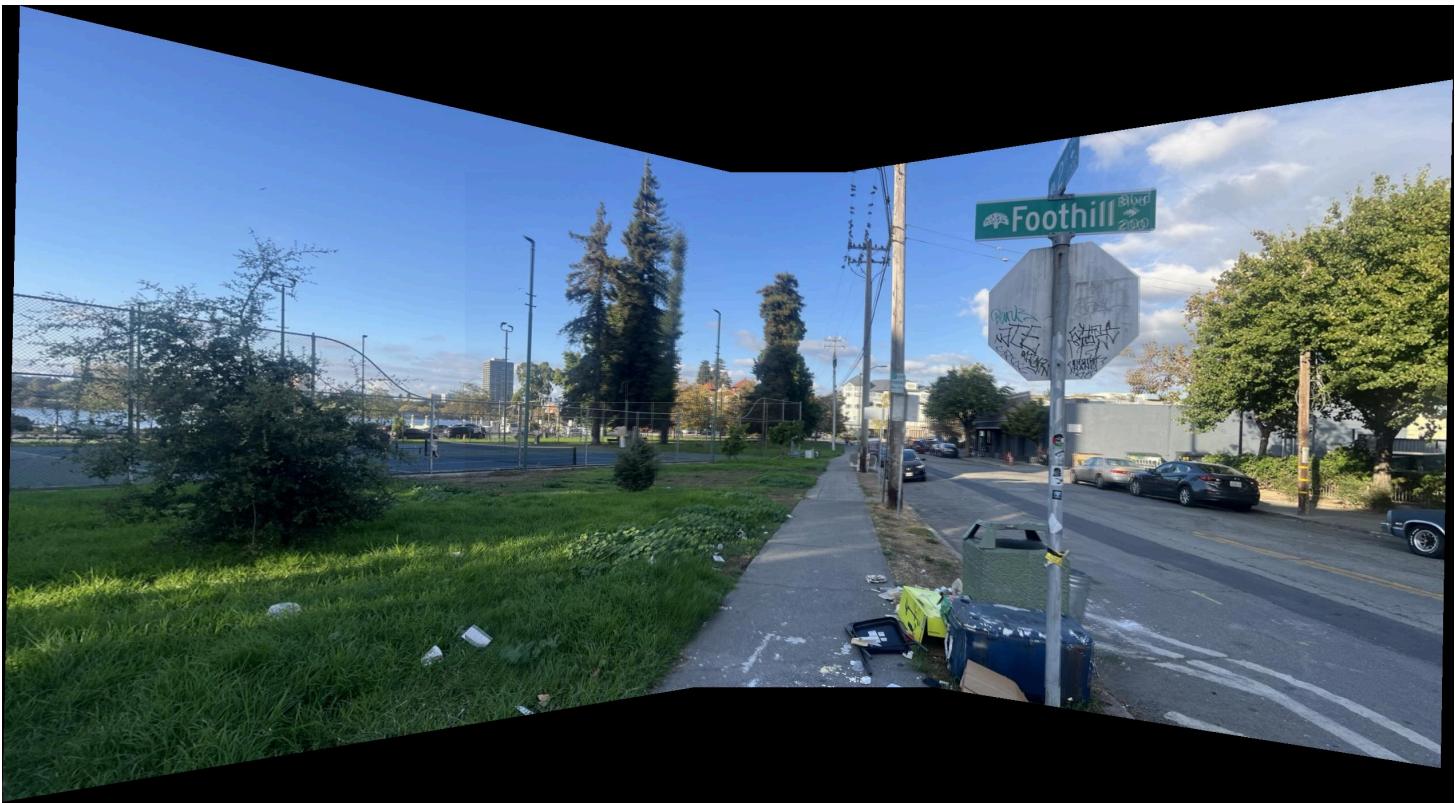
The full library scene mosaic:



The full neighborhood mosaic:

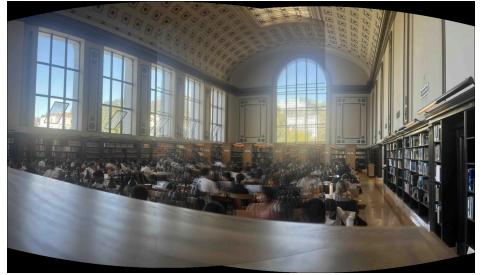


And an additional Oakland scene mosaic:



A.5 Additionally, I tried mapping images onto a cylindrical surface. This meant changing the coordinate system and diverging from linear transformations. For previous mosaics, I manually created point coorespondences which were used to create a homography matrix. This matrix was no longer useful in the cylindrical coordinate system. After translating both images into cylindrical coordinates, I tried to estimate a translation by translating the original 2D coorespondences into cylindrical coordinates, finding a median translation, and translating two images. However, the resulting images are not as convincing as the 2D mosaics. Some example results of two images are below.

Here is image 1 in cylindrical coordinates, image 2 in cylindrical coordinates, and a blend of the two images.



Here is another example.

