Name (netid): Tyler Smithline (tylergs2)
CS 445 - Project 5/Final Project: Image Based Lighting

Complete the claimed points and sections below.

Total Points Claimed	**[140] / 250**

Core

1. Stitch two key frames	[20] / 20
2. Panorama using five key frames	[15] / 15
3. Map the video to the reference plane	[15] / 15
4. Create background panorama	[15] / 15
5. Create background movie	[10] / 10
6. Create foreground movie	[15] / 15
7. Quality of results and report	[10] / 10

B&W

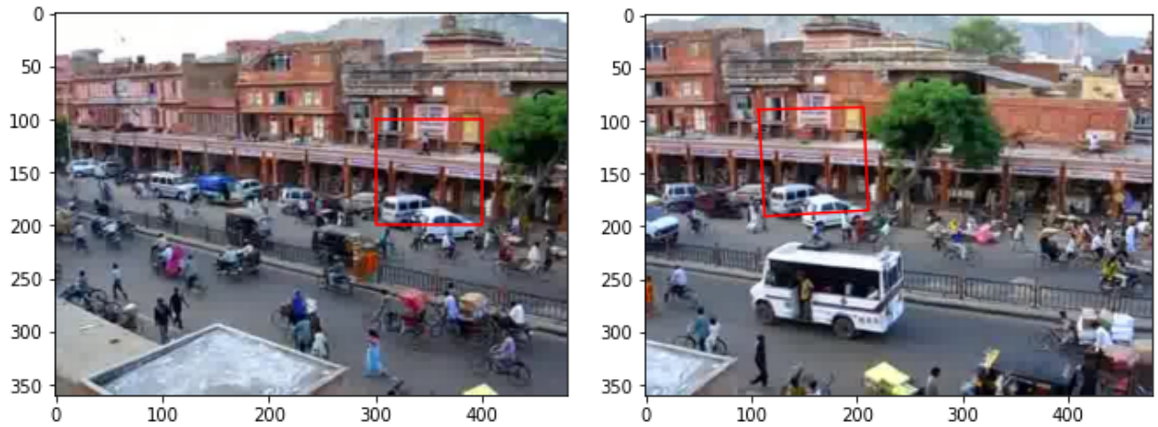
8. Insert unexpected object	[ ] / 15
9. Process your own video	[ ] / 20
10. Smooth blending	[30] / 30
11. Improved fg/bg videos	[ ] / 40
12. Generate a wide video	[10] / 10
13. Remove camera shake	[ ] / 20
14. Make streets more crowded	[ ] / 15

Project Overview and Results

I am submitting this project as my final project. I completed all of the core sections, as well as the wide video and smooth blending sections of the bells and whistles. Generally speaking, I am very pleased with the quality of my results. The initial stitching, mapping, and background movie all came out well, but the foreground movie didn't come out quite as well as I had hoped. I found it challenging to make an effective foreground pixel detection algorithm, and future work could be done to improve the functionality of it to reduce noise. I really enjoyed experimenting with the applications of homographies and found the results to be visually satisfying. I hope to finish the bells and whistles section of this project on my own time after the semester ends to experiment with better blending and creating results using my own video.

1. Stitch two key frames

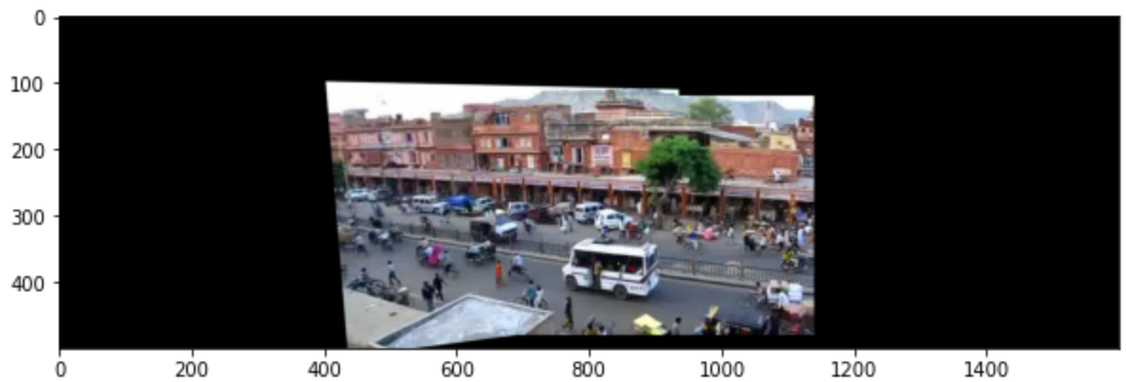
Display of image frames 270 and 450 with the red plot lines showing corresponding regions:



Printout of 3x3 homography matrix normalized so that the largest value is 1:

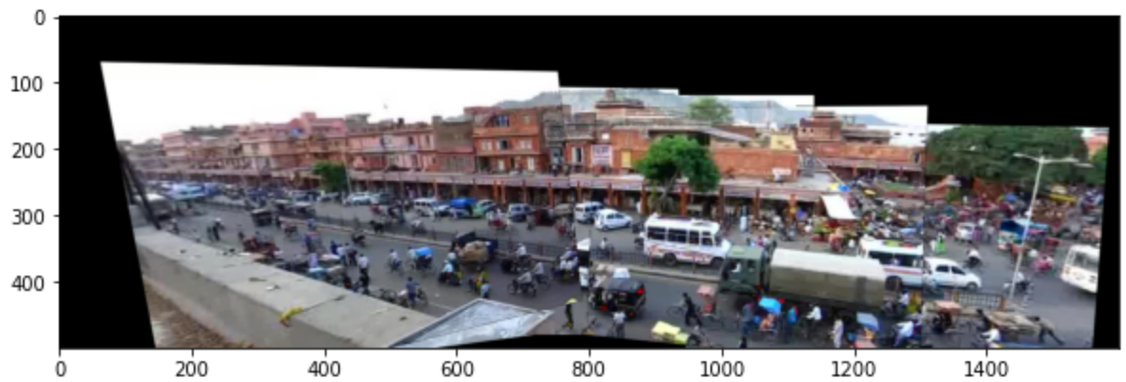
```
H:
[[ 1.00000000e+00  5.12212564e-02 -2.04990719e+02]
 [ 1.75303135e-02  9.58080172e-01 -1.77573023e+01]
 [ 3.93819367e-04  6.76698071e-05  8.07355636e-01]]
```

Stitch of two key frames:



2. Panorama using five key frames

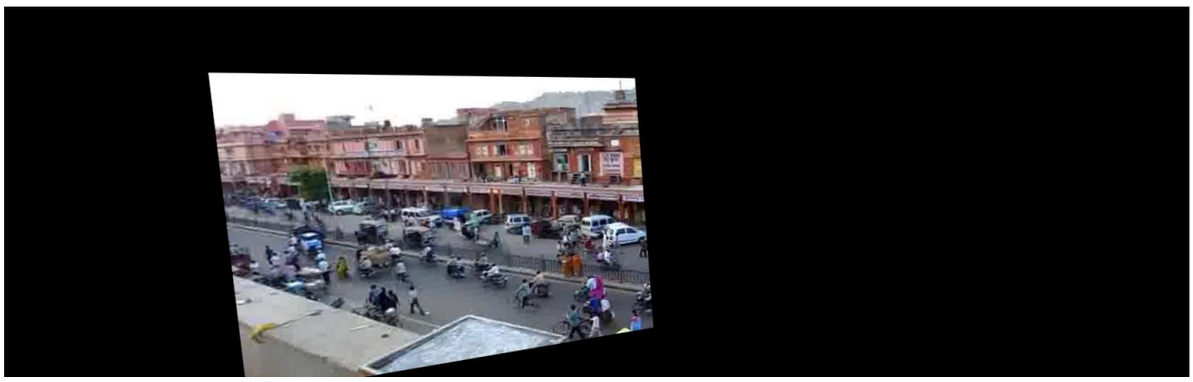
Include your panoramic image:



3. Map the video to the reference plane

Link to my video:
https://youtu.be/FdCvcB41WtU

Display frame 200 of your video:



Briefly explain how you solved for the transformation between each frame and the reference frame:

I first compute the homographies for the five key frames (90, 270, 450, 630, 810). I then find which key frame is the closest for each of the 900 frames by finding the absolute difference between the current frame index and each of the key frame indices, and I use argmin to determine the index of the closest key frame in the key frames index array.

Then for each frame, I compute the homography between that frame and the closest key frame. I use a dot product between that homography and the homography between the closest key frame and the reference frame to find the homography from the current frame to the reference frame. The transformation is done using the cv2.warpPerspective function sending the dot product between the translation matrix and the corresponding homography that we just calculated as a parameter. This results in an image warped to be in the same plane as the reference frame.

4. Create the background panorama

Picture of the background panorama:

Explain your method of computing the background color of a pixel:

I first put each frame from the video into a single numpy array of projected frames, with the first axis being time. I then perform a median calculation along the time axis of this array, since this calculation tends to ignore outliers and will show the background pixel if it appears for 50% of the time. To accomplish this, I use np.apply_along_axis with a lambda function applying np.median to only consider nonzero values of the array. This appears to produce better results near the center of the image, which makes sense as the video got shaker near the edges of the frame.

5. Create the background movie

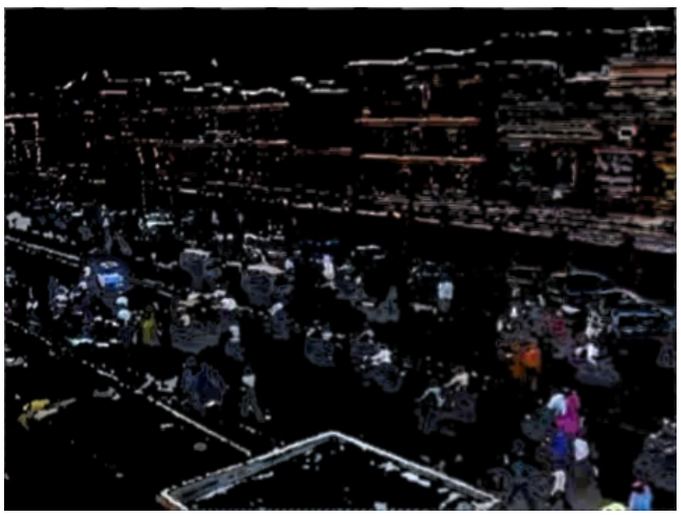
Link to your video:
https://youtu.be/VEdesnAOhXY

Display frame 200 of your video:

6. Create the foreground movie

Link to your video
https://youtu.be/es3vKwgpYi8

Display frame 200 of your video:



My results for the foreground video had more noise than I would have liked. I used euclidean distance from the background to determine foreground pixels, but a different method may have resulted in less noisy results.

7. Quality of results / report

Nothing extra to include (scoring: 0=poor 5=average 10=great).

8. Insert unexpected object

Include link to your video.

9. Process your own video

Include:

- Background image
- Link to background video
- Link to foreground video

10. Smooth blending

Include panoramic image from part 2 with better blending:

11. Improved fg/bg videos

Include panoramic image from part 2 with better blending

12. Generate a wide video

Include link to your video:
https://youtu.be/exMFeJd6yKM

13. Remove camera shake

Include link to your stabilized video

14. Make street more crowded

Include link to your video

Acknowledgments / Attribution

List any sources for code or images from outside sources

Stack Overflow (“CT Zhu”):
https://stackoverflow.com/questions/22049140/how-can-i-ignore-zeros-when-i-take-the-median-on-columns-of-an-array