Name: Ashwin Ashok
Person Number: 50372708
Project 2 Report

Project 2 Report

Task 1

Code Description

The code for this section contains 4 sub-functions within `stitch_background()`. `feature_matcher()` uses ORB with a 2000 feature limit to generate keypoints and descriptors for the two images to be stitched. The code for matching features is also provided in the same function. The Euclidean distance between all descriptors for Images 1 and 2 are found, the list is sorted in ascending order of distance, and the ratio test is applied based on a threshold of 0.7 using the lowest and second lowest distances. If the test is passed, the `good` list is appended with the Descriptor 1 index (`queryIdx`) and the corresponding index in Descriptor 2 (`trainIdx`).

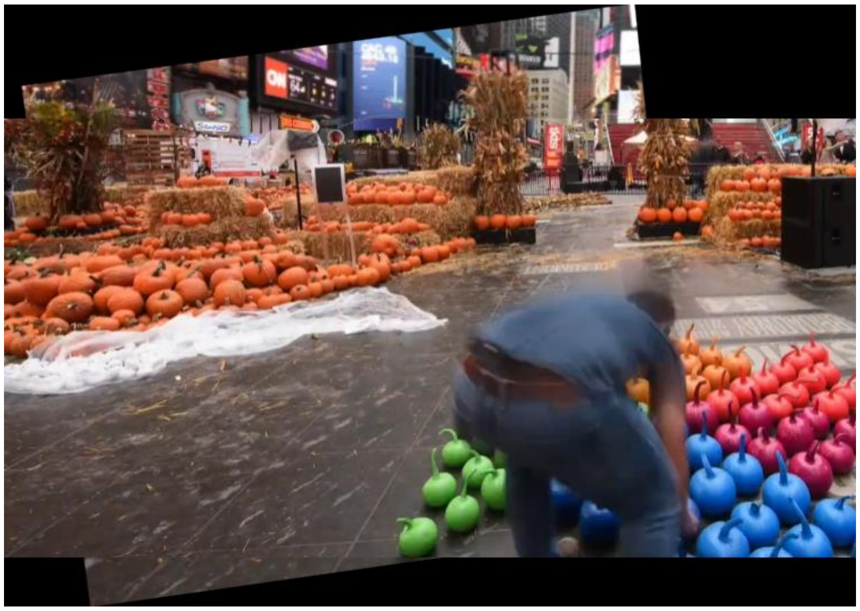
The `perspective_transform()` function is used to find the homography matrix. The code for this is provided in OpenCV documentation (see references). The indices of good matches obtained in `Feature_matcher()` are used to find the relevant keypoints. Then, the `findHomography()` function is used to find the homography matrix.

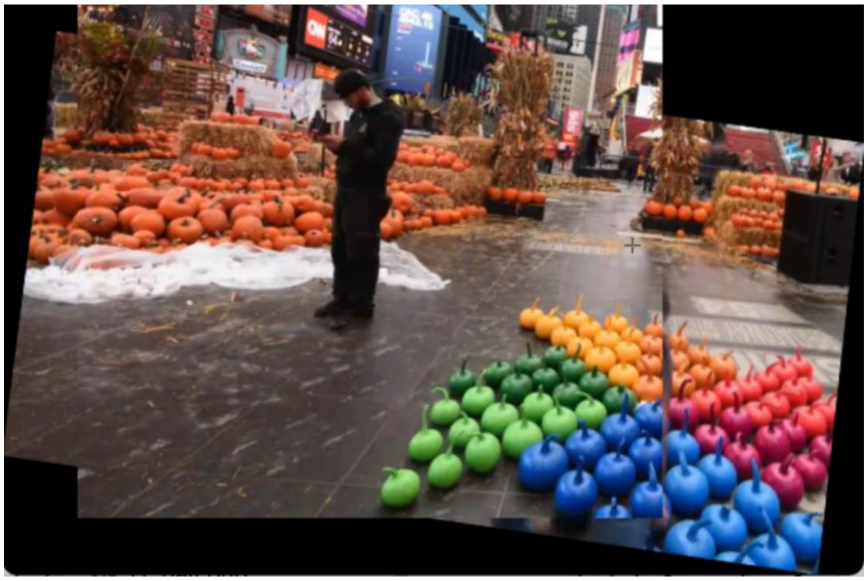
The `warpImages()` function is used to transform the images based on the homography. Parts of this code are available in OpenCV documentation (see references). First, points from each image are selected. The `perspectiveTransform()` function is used to find the coordinates of one image in the frame of the other. Then, the maximum and minimum coordinates are found, and these are used to create a translation vector for the homography matrix H. This prevents the picture from cropping out at the corners of the window. The translation matrix `H_trans` is multipled with the homography matrix, and finally the `warpPerspective()` function is used to output the transformed Image 1. This image then saved using the savepath provided.

Name: Ashwin Ashok
Person Number: 50372708
Project 2 Report

Functionality

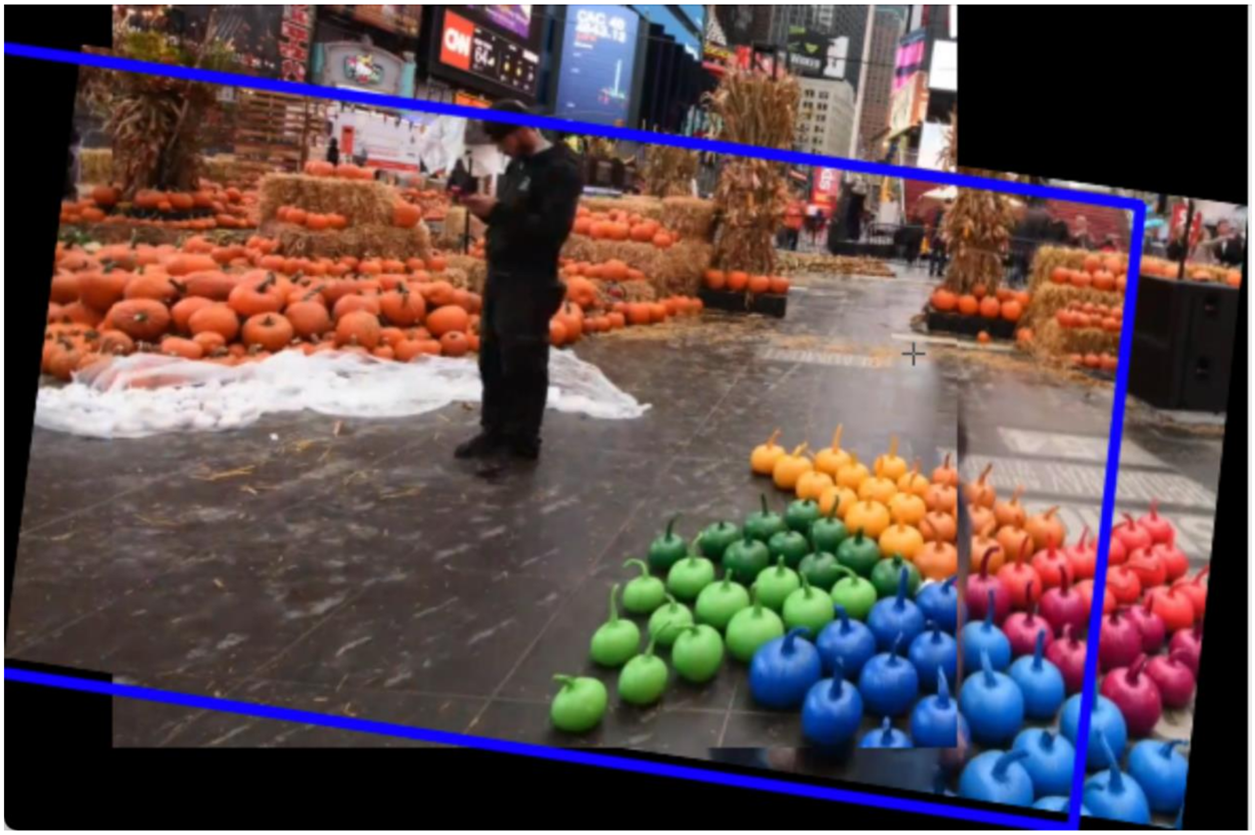
The code obtained did not remove the foreground. The following images show the results when warping is done on image 1 and image 2 respectively.





The following image shows the overlapping region identified using bounding boxes.

Name: Ashwin Ashok
Person Number: 50372708
Project 2 Report



Given more time, I would have done the following to remove the foreground. Crop out the section of pixels in the bounding box. Apply the same homography to the same section of the other image to find the overlapping region without the foreground. Compare the sections pixel by pixel and identify the foreground and replace the foreground pixels.

Task 2

Code Description

This task uses mostly the same code from the previous task. However, there are a few major additions. Firstly, ORB is used to compute and save the keypoints and descriptors of all input images.

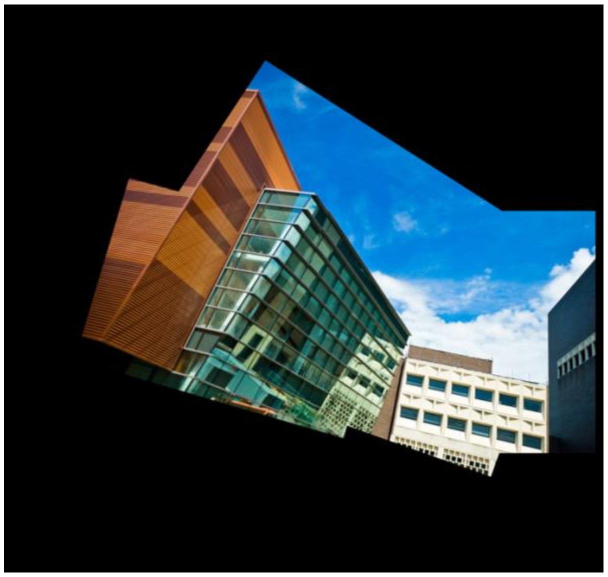
Secondly, the feature matching code from the previous task is applied to all pairings of images. The aim of this is to find the pairs of images that fit together best, and create a threshold based on the number of good matches. These pairings are noted and listed in descending order of number of matches. Image stitching is done with the first pair first to ensure a good match. This would create a base for other images to stitch onto for the panaroma. The pairings and their number of good matches are saved in `match_list`.

Thirdly, there is a loop code at the end to repeat the stitching through all the pairings. When two images are stitched together, the result is stitched with the next image, and so on. This loop looks for pairs where one of the images has already been stitched. This prevents images that are not yet connected to be stitched later when one of their pairings is already stitched.

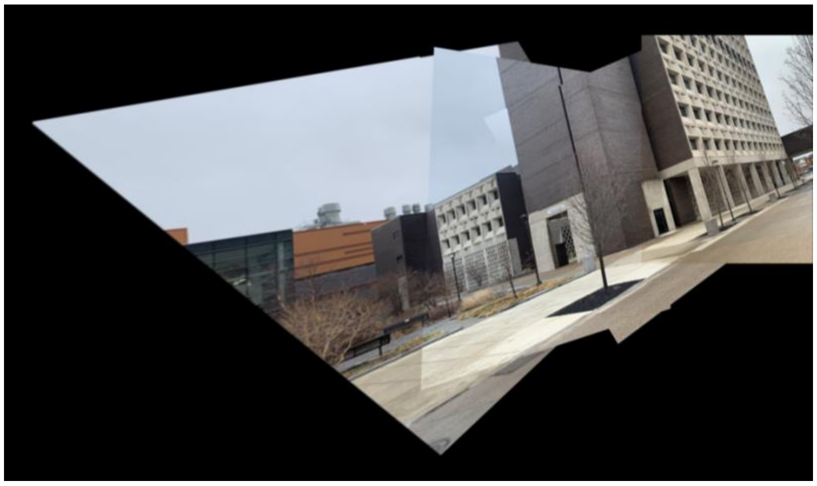
Finally, a section is added to calculate the one-hot array of matches. This goes through `match_list` to represent the matches in `overlap_arr`.

Functionality

The code is successful in outputting the panorama and overlap array. The final panorama for the given images is shown.



For the bonus question, the final panorama for a set of 5 images rather than 4 is shown below. Note that this image has been cropped to make the panorama more visible.



Name: Ashwin Ashok
Person Number: 50372708
Project 2 Report

References

https://docs.opencv.org/master/d1/de0/tutorial_py_feature_homography.html

https://docs.opencv.org/master/da/d6e/tutorial_py_geometric_transformations.html

https://docs.opencv.org/3.4/d1/d89/tutorial_py_orb.html