Ko Outlaw-Spruell
SID: 914662783

Problem Set 2

I. SHORT ANSWER PROBLEMS

1. **Which grouping algorithm would be appropriate to recover the model parameter hypotheses from the Hough continuous vote space? Briefly describe and explain.**

   Mean-shift would be the best approach to recover the model parameter hypotheses. The problem with K-means is that the user must specify the number of clusters to find. Since we cannot predict this, using K-means will not return desirable outcomes. Since the desired parameters will have many points around it in high density, we want to find small clusters with high density. Graph-cut algorithm does not allow for us to specify a window size, so this is not ideal either. So, we use mean-shift, and if we specify a window that is relatively small(similar size to the discrete squares in the Hough space), the mean-shift will create multiple clusters, many of which will have a high density. Go through each cluster, and if a cluster contains more points than the specified threshold, use the center of gravity point of that cluster to be the model parameter.

2. **What is a likely k-means clustering assignment that would result for a space that has points arranged as a circle within another circle, with k=2? Briefly explain your answer.**

   The desired outcome will be the points representing the inner circle being one cluster, and the points representing the outer circle being another, but this is not what will happen. K-means clustering can only detect spherical arrangements. So, the algorithm will split both of the circles in half. Wherever the two points start, the next points(mean values) will be between the two circles, across from each other. The first cluster will contain half of the inner and outer circle points, and the second cluster will contain the other half of the inner and outer circles.

3. **Write pseudocode showing how to group the blobs according to the similarity of their area (# of pixels) into some specified number of groups. Define clearly any variables you introduce.**

```
function findSimilarBlobs

Points[] = array of black points in the binary image
WinSize = window size for mean-shift

//use mean-shift algorithm to categorize the black pixels into blobs
for each point p in Points
        centerPt = p
        while (centerPt still moving)
                calculate the mean point in window using WinSize and centerPt
                store the mean point in centerPt
        end while
        CenterPts[p] = cnterPt //store the peak(centerPt) for this point in array
end for

Blobs[] //array containing the number of pixels in each blob. Initialize all to 0

//make an array with the contents being the number of black points(area) for each cluster(blob)
for each p in Points
        find out which blob p belongs to
        increment count in that Blob
end for

set number of clusters, k
randomly assign numbers for k centroids

//use K-means to cluster the blobs using area as a feature(which are the contents of Blob[])
while centroids not converged
        find Blobs that belong to each centroid
        recompute centroid
end while

return  an array containing info about each blob(which are an array of pixel coordinates),
        and the cluster # for each blob.

end function
//Done. We have a cluster of blobs, where the blobs with similar area being in the same cluster
```

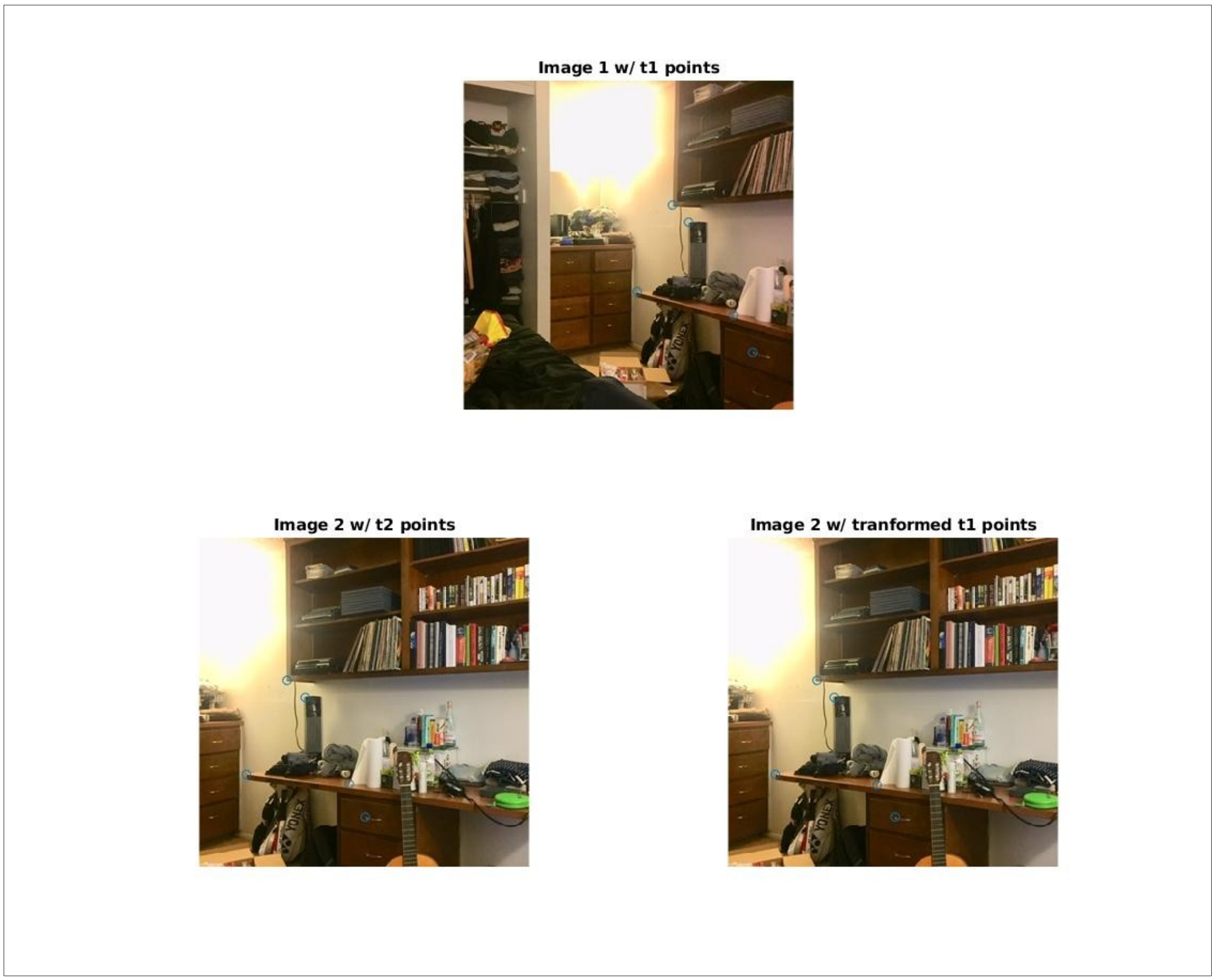
II. PROGRAMMING

a) Getting Correspondences

code is in file named *getPoints.m*

b) Computing the homography parameters:
Write a function H = computeH(t1, t2)
Verify that the homography matrix your function computed is correct by mapping the clicked image points from one view to the other, and displaying them on top of each respective image.

Verification:

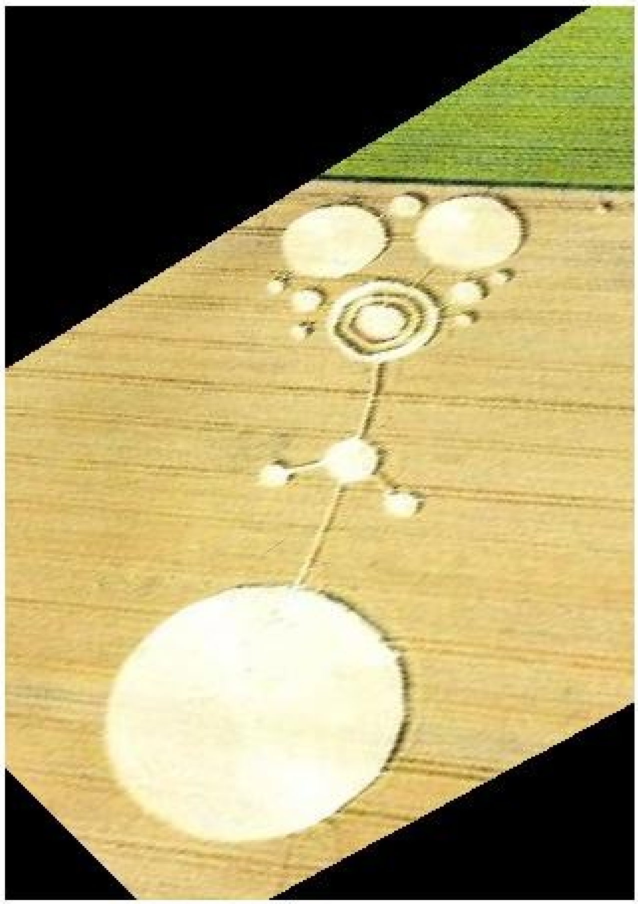


c) **Warping between image planes:**
code written in *warpImage.m*

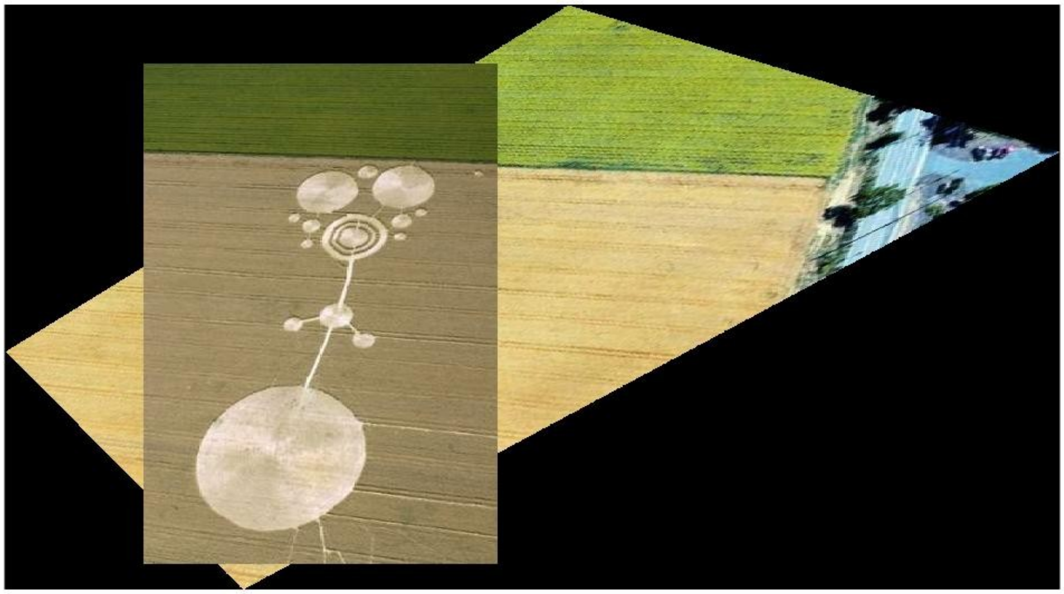
d) **Apply your system to the following pairs of images, and display the output warped image and mosaic in your answer sheet.**

Pair 1:

Warped Image:

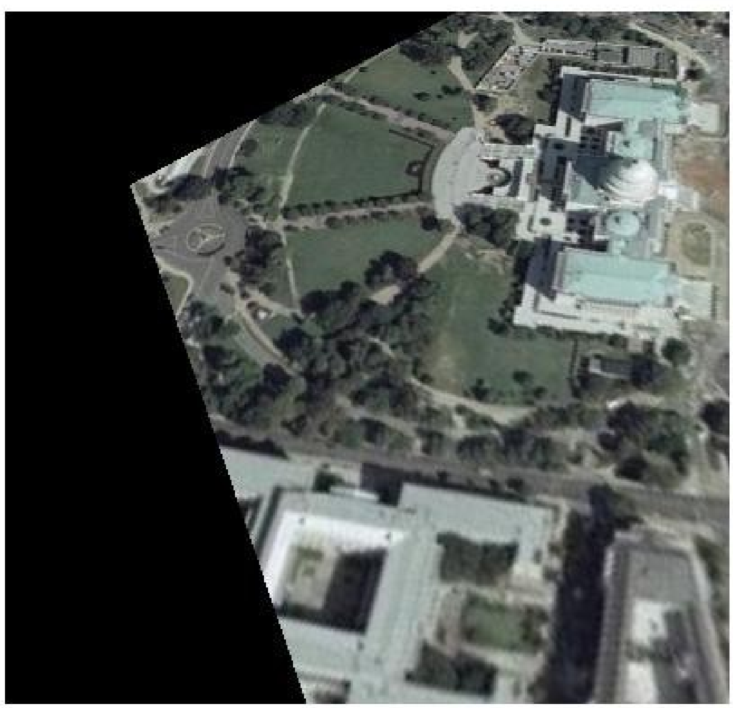


Merged Image:

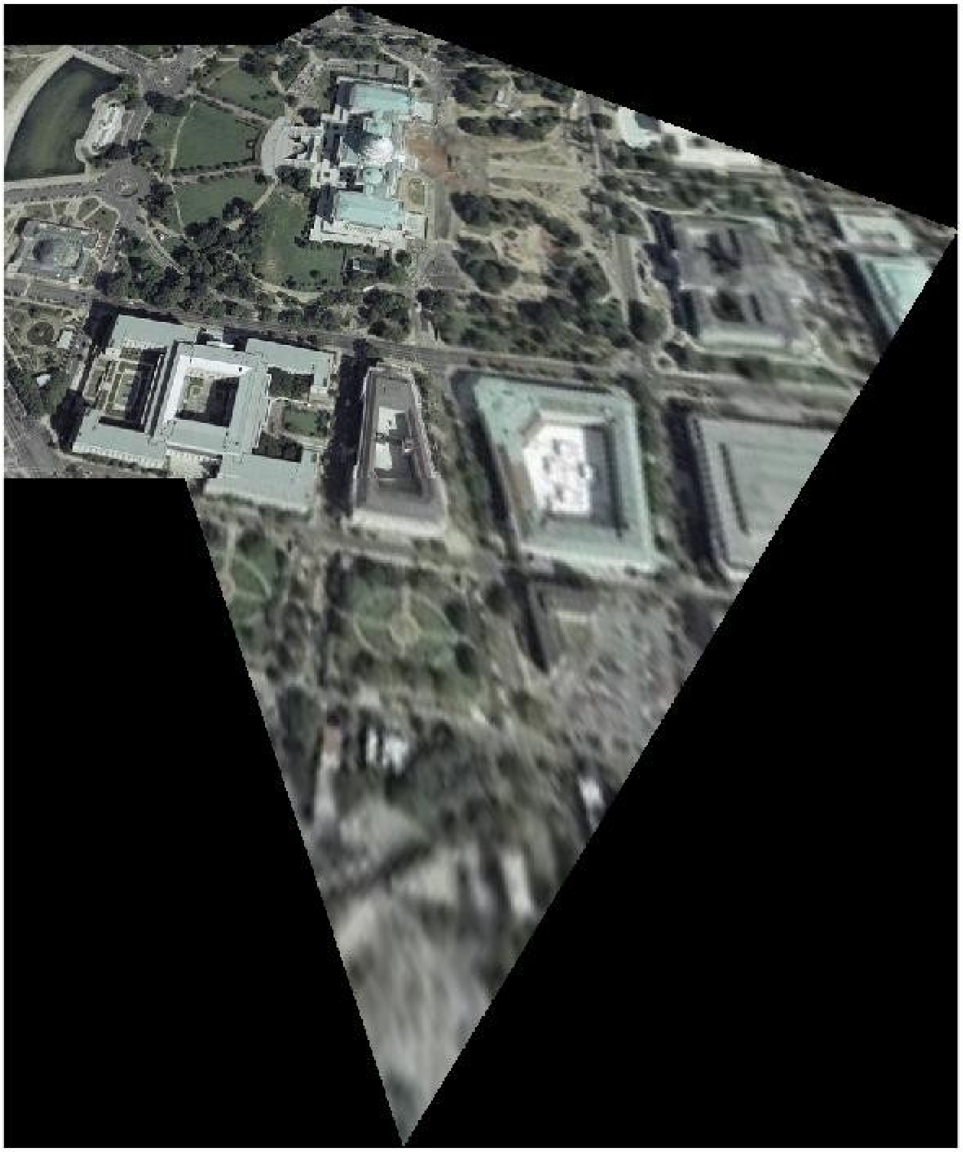


Pair 2:

Warped Image:

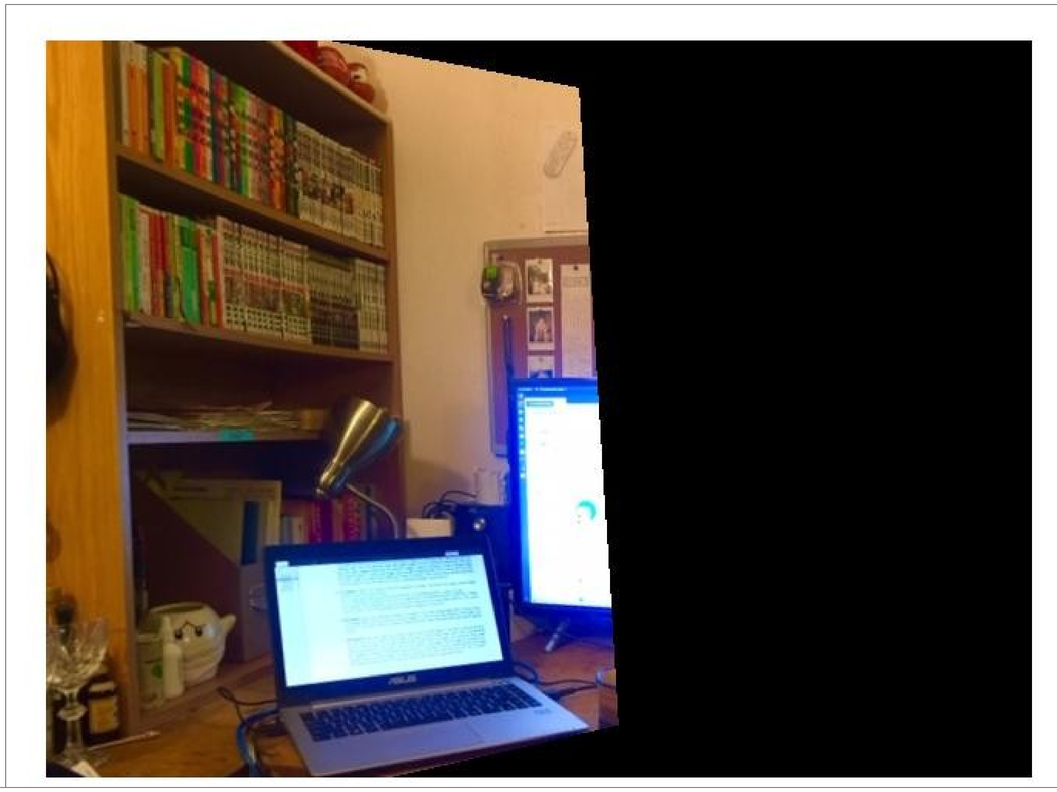


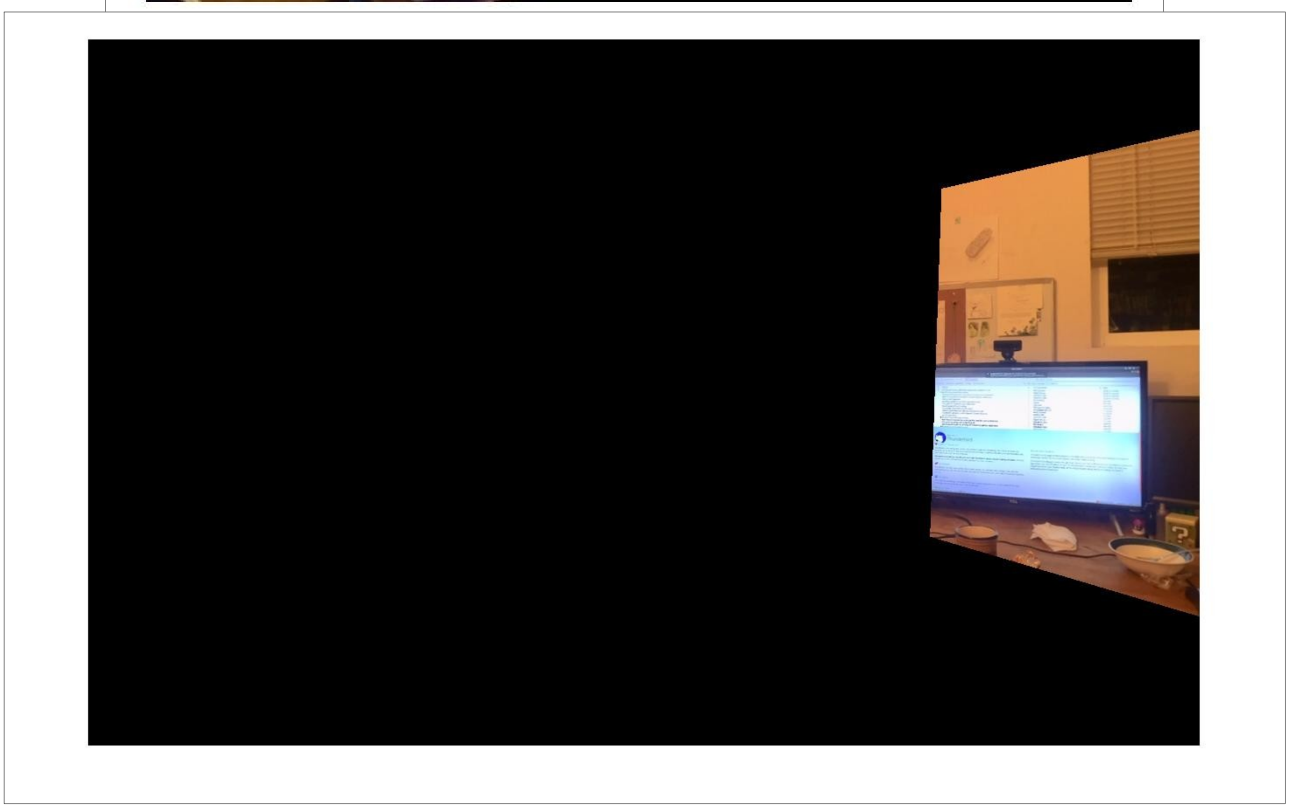
Merged Image:



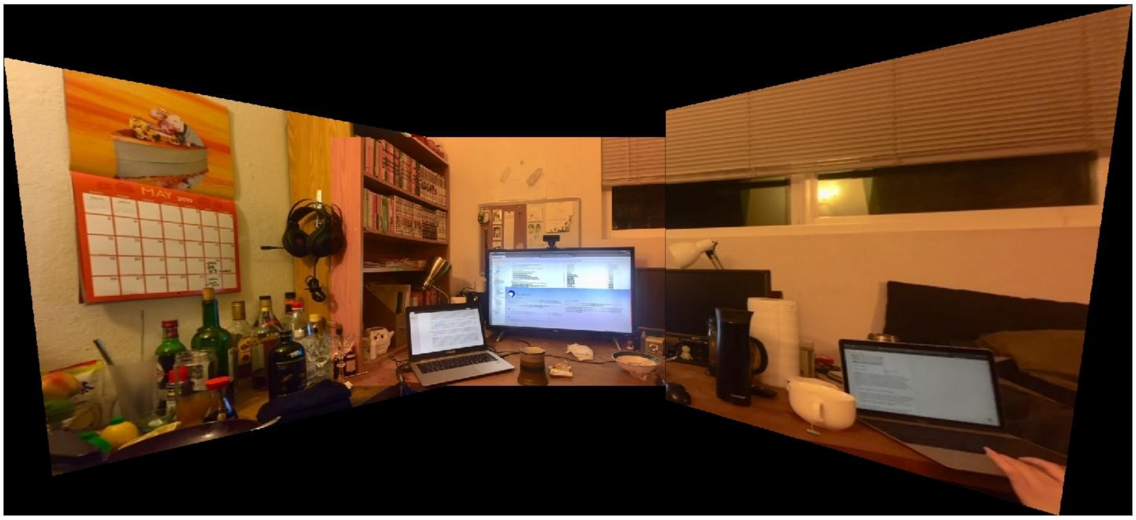
e) Show one additional example of a mosaic you create using images that you have taken.

Warped Image:





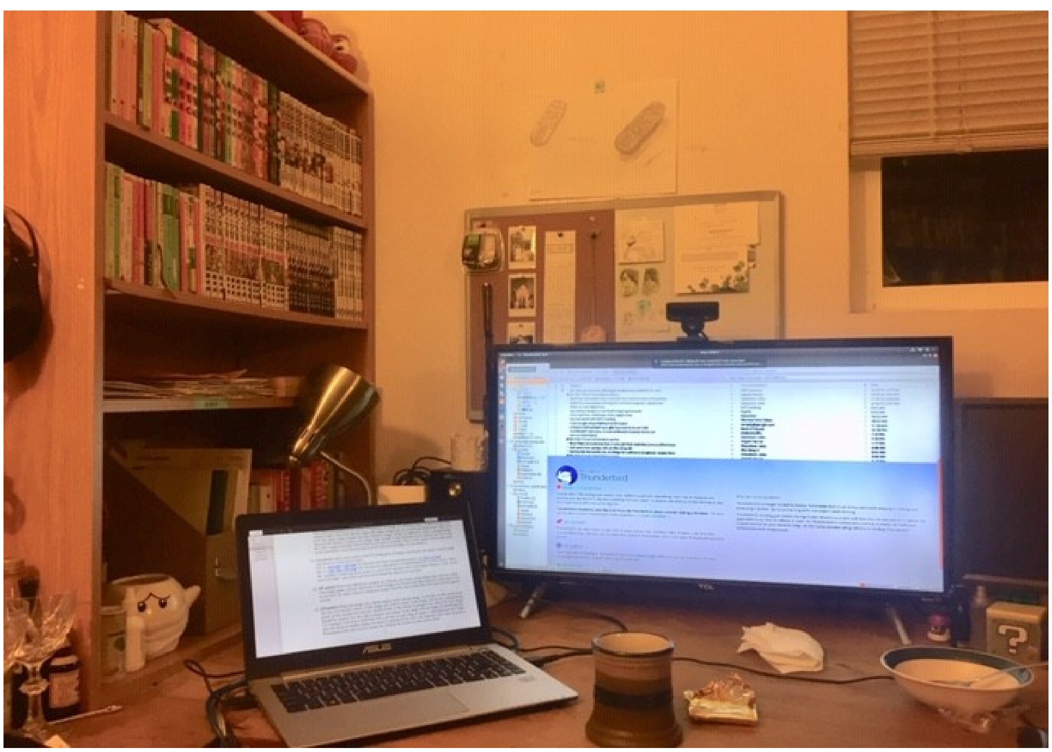
Merged Image:



f) Warp one image into a frame region in the second image.
Image to frame:

Source: Nicholas Cage’s face

Background:

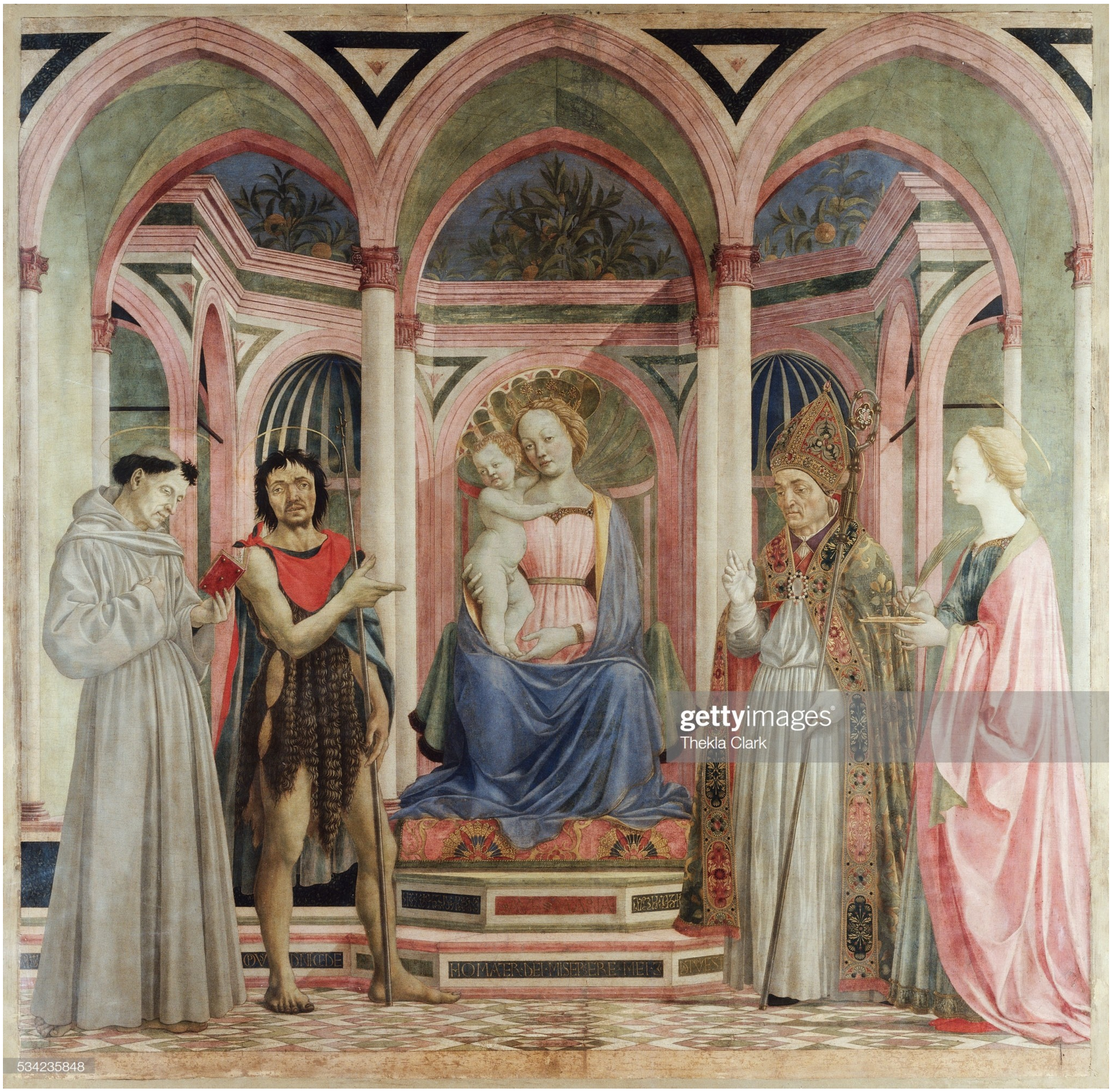


Result:

III. EXTRA CREDIT

b) Rectify an image with some known planar surface and show the virtual fronto-parallel view.

Original Image:



Rectified Image: