Jonathan Pearce

260672004

Comp 558

Assignment 4

1.

C1.jpg

C2.jpg

2A.

K (white squares) and R (red squares) image shift.

The shift transformations created by modifying the K and R matrices are quite similar, although not exactly the same. It appears that the points from the modified R matrix (red squares) over and under estimate the K modified points depending on where they are in the image frame, this is experienced on both the x and y axis. R modified points on the far right side of the image have greater x axis pixel values, where as the points near the middle of the image (the left part of the located points) have lower x axis pixels values. The same can be said with the y axis pixel values and whether the points are at the bottom or at the top of the group of mapped points. Therefore we conclude that the modified rotation matrix (R) seems to have stretched the points outwards in both the x and y direction more than modified camera intrinsic matrix (K), it is even possible that the modified camera intrinsic matrix completed a pure shift (i.e. no stretching or warping) and only the R matrix changed/stretched the points. This will be investigated quantitatively in question 3.

K (white squares) and C (red squares) image shift.

The same pattern as with the previous comparison emerged here. The stretching of the points by the modified C matrix was more extreme in comparison to the modified R matrix in the previous image. It is difficult to conclude anything concrete though since this analysis is only qualitative. Regardless we see that the shift transformations created by modifying the K and C matrices are similar and a quantitative analysis will be completed in question 3.

2B.

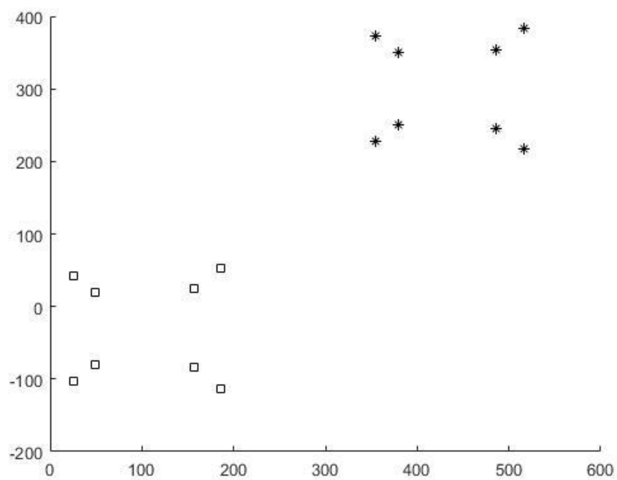
K (white squares) and C (red squares) image Scaling.

The scale transformations created by modifying the K and C matrices are similar, although not exactly the same. The most significant observation is that the distance between the two different transformations seems to increase as the points get farther away from the ‘center’ point. In the lower area of the photo the two transformations have points quite close to one another, in fact one point (highlighted with yellow box) has experienced nearly the exact same coordinate transformation. However looking near the top edge of the image the 2 transformations’ points are not nearly as close. The modified camera intrinsic matrix has transformed images much closer to the physical image edge then the modified C matrix. We will investigate the reason for these difference quantitatively in question 3.

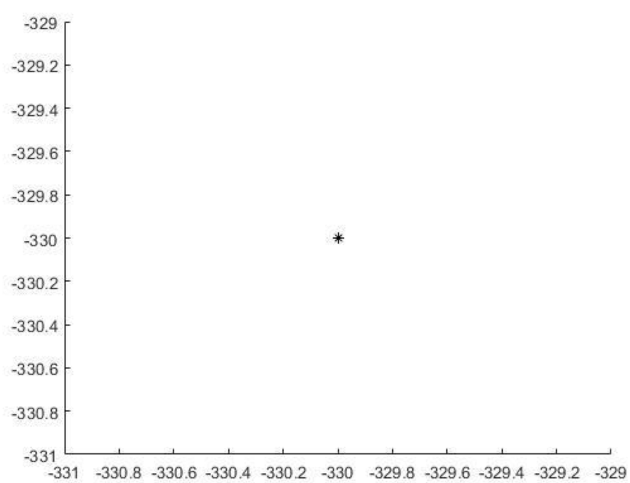
3. Modifying K Matrix for image **shift**

$$\text{Matrix multiply (used to modify K): } \begin{pmatrix} 1 & 0 & 1.1 \\ 0 & 1 & 1.1 \\ 0 & 0 & 1 \end{pmatrix}$$

Modifying K via matrix multiplication with the matrix above produces the following shift in points (square is original, star is new point location)



We can then analyze the change in pixel coordinates in both the x and y direction for each point. (Change in x on x-axis, change in y on the y-axis)

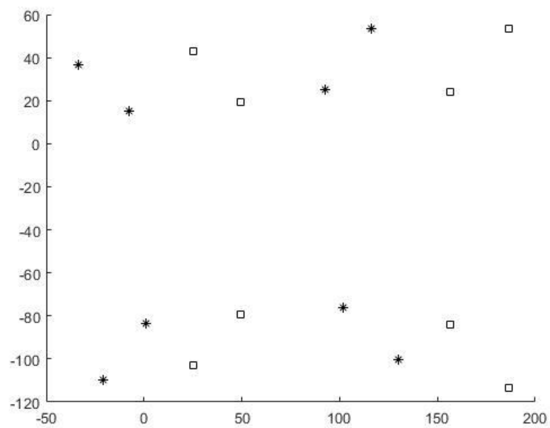


This modification of the K Matrix only changed p_x and p_y (i.e. the location of the principal point). From the second figure we observe that this creates an exact shift (all points move in the exact same way), there is no stretching or warping of the point locations relative to one another. This type of shift makes sense since we our simply shifting the optical center of the camera (intrinsic property) which implies every pixel shifts by a fixed amount in the x and y direction, we confirm this using the second figure.

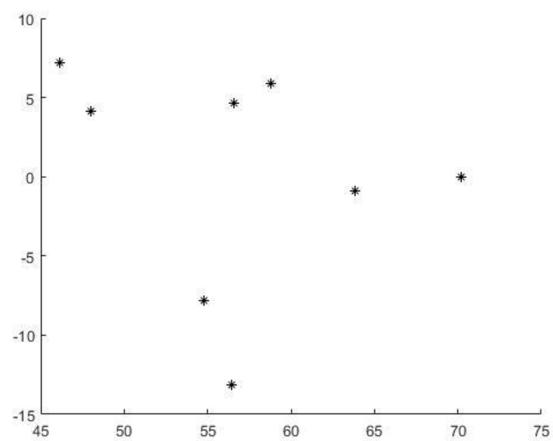
Modifying R Matrix for image **shift**

To create this matrix multiply (to modify R) I multiplied a -10 degree rotation matrix about the y axis with a 5 degree rotation matrix about the z axis. This ensured that R remained a proper rotation matrix

Modifying R via matrix multiplication with the matrix described above produces the following shift in points (square is original, star is new point location)



We can then analyze the change in pixel coordinates in both the x and y direction for each point. (Change in x on x-axis, change in y on the y-axis)

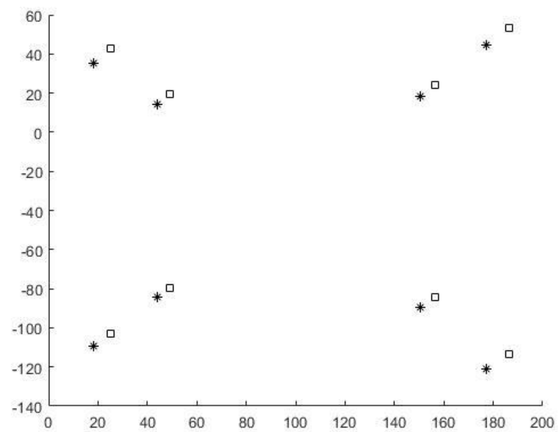


Comparing this method with the K matrix modification above, it is somewhat clear from the first figure that the transformations are different, however the second figure is what tells us why. As we can see in the second figure after modifying the rotation matrix, no 2 points are shifted in the same way in the image. This is because changing the rotation matrix changes the camera's extrinsic orientation in world coordinates and warps the image points depending on the type of rotation. An example of this is in question 4, where the second and third transformations produce trapezoidal image shapes, due to warping. From this it is clear why shifts produced from modifications with the K and R matrices are different.

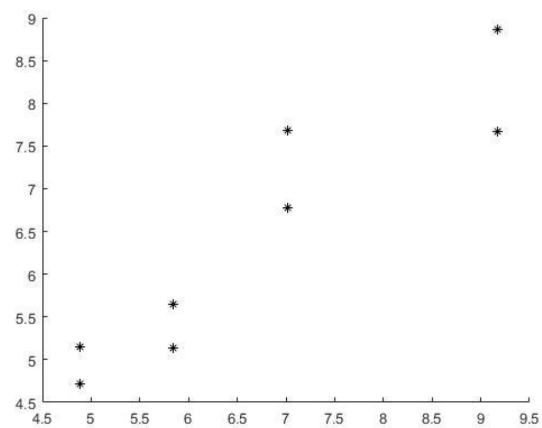
Modifying C Matrix for image **shift**

To modify the C matrix I multiply C_X and C_y by a positive constant (1.1 in this example).

Modifying C via matrix multiplication with the matrix described above produces the following shift in points (square is original, star is new point location)



We can then analyze the change in pixel coordinates in both the x and y direction for each point. (Change in x on x-axis, change in y on the y-axis)

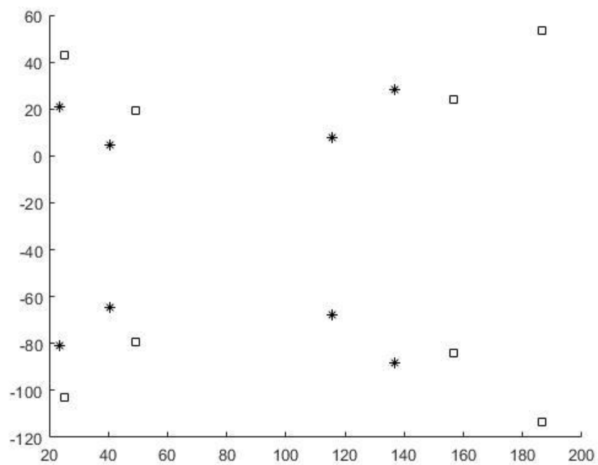


Similar to the R matrix analysis above, it is the second diagram that immediately shows us why these shifts produced by modified K and C matrices are different, in particular for this case all points shift different amounts in the new image, whereas the shift was uniform for K. Modifying C_X and C_y changed the camera's extrinsic position in world coordinates, therefore the change in point positions differs for each point depending on where the point is relative to the camera in the scene, this is why we see each point move different amounts in the image. In the modified K matrix we only modify the intrinsic optical center of the camera, it is the difference between modifying extrinsic and intrinsic coordinates that explains the difference in shifts.

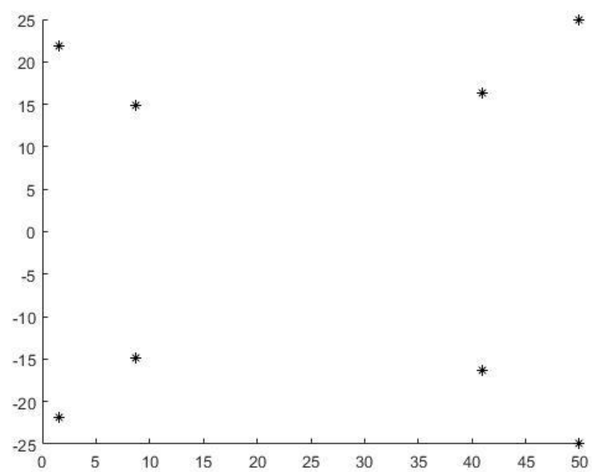
Modifying K Matrix for image **scaling**

$$\text{Matrix multiply (used to modify K):} \begin{pmatrix} 0.9 & 0 & 0 \\ 0 & 0.9 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Modifying K via matrix multiplication with the matrix above produces the following shift in points (square is original, star is new point location)



We can then analyze the change in pixel coordinates in both the x and y direction for each point. (Change in x on x-axis, change in y on the y-axis)

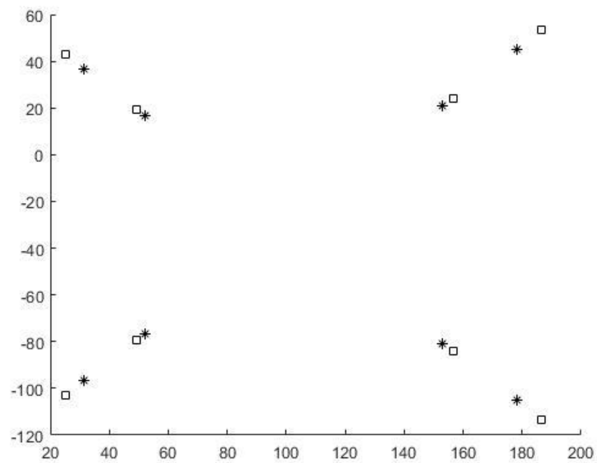


This modification of the K Matrix only changed fm_x and fm_y (i.e. the product of the focal point and the number of pixels per mm in the x and y direction). The most significant observation from the figures is that the scaled point locations changed by different amounts in terms of shift in x axis and y axis, i.e. the scaling was not uniform.

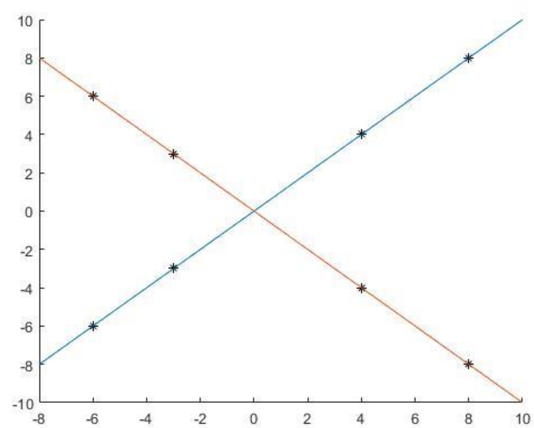
Modifying C Matrix for image **scaling**

To modify the C matrix I multiply C_z by a positive constant (1.1 in this example).

Produces the following expansion in points



We can then analyze the change in pixel coordinates in both the x and y direction for each point. (Change in x on x-axis, change in y on the y-axis). Lines drawn on image are the lines y = x and y = -x.



An initial observation is that this modified C matrix has created a uniform scaling with respect to the change in x axis and y axis pixel values (i.e. absolute change in x is equal to absolute change in y). This is shown visually using the 2 line segments (y = x, y = -x) drawn on the second figure. Modifying C_z changed the camera's extrinsic position in world coordinates, this is equivalent to simply moving the camera toward or away from the scene along the z axis. Similar to our analysis for image shifts it is the difference between modifying intrinsic camera properties (K matrix) and extrinsic camera coordinates in world coordinates (C matrix) that creates the difference between the scaling of these image points.

4.

For this question I assumed positive theta was the equivalent to rotating the camera clockwise around the y axis.

$\Theta = 0$:

Θ = 10:

Θ = 20: