

CPSC 425: Assignment 4
Name: Terence Chen
Student number: 42602136

Q1) Nothing to hand in

Q2) Nothing to hand in

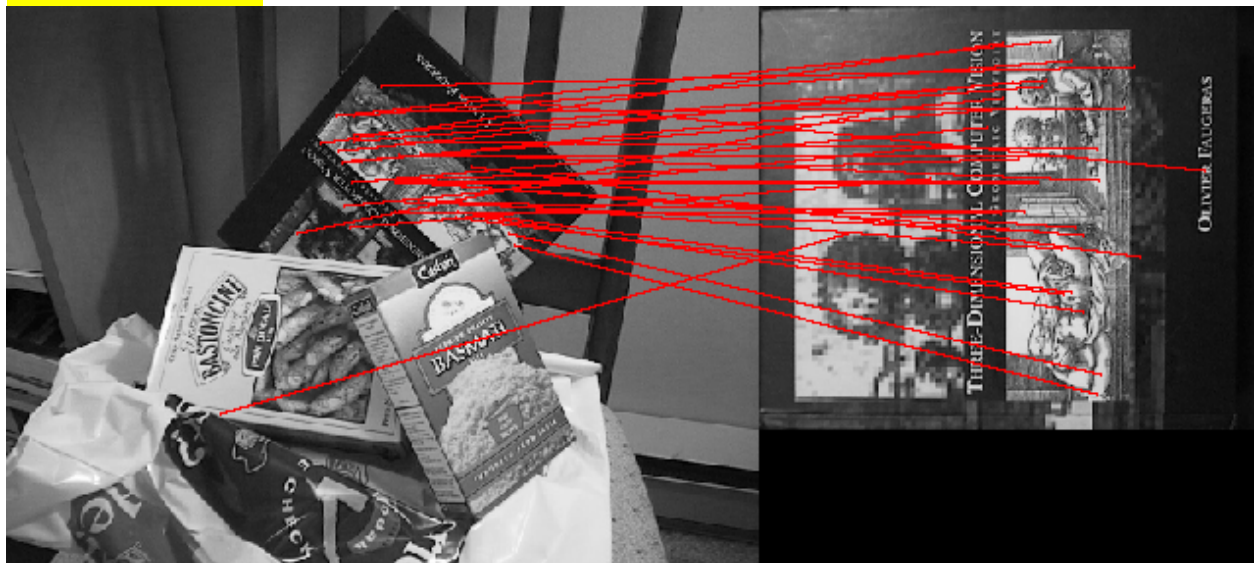
Q3)

Code changes in match function:

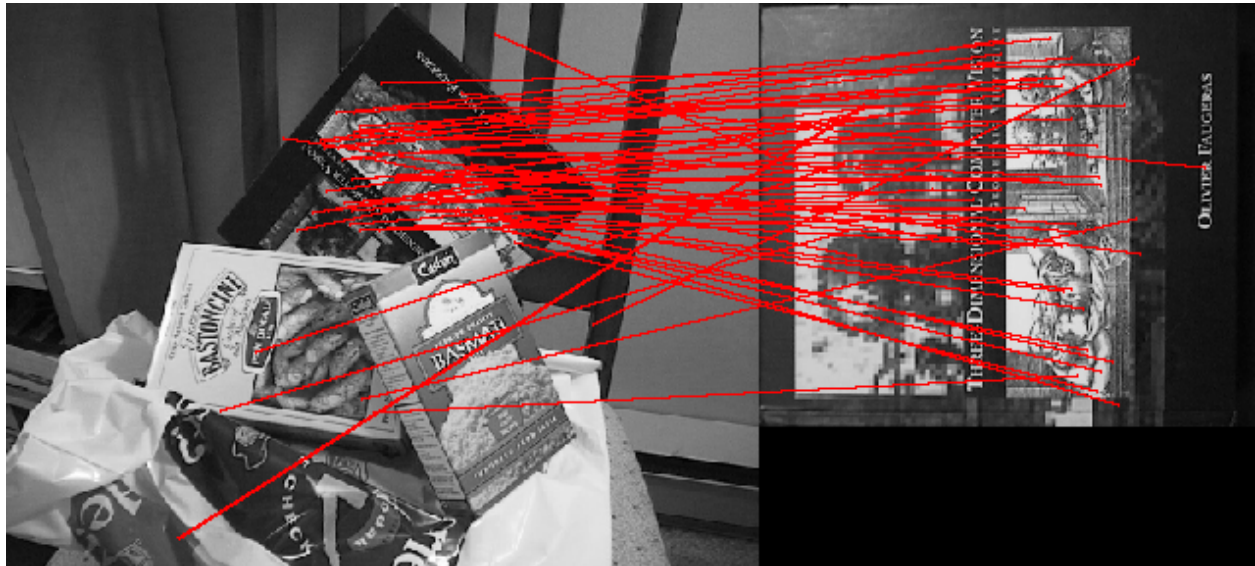
```
95 im1, keypoints1, descriptors1 = ReadKeys(image1)
96 im2, keypoints2, descriptors2 = ReadKeys(image2)
97 #
98 # REPLACE THIS CODE WITH YOUR SOLUTION (ASSIGNMENT 5, QUESTION 3)
99 #
100 #Generate five random matches (for testing purposes)
101 matched_pairs = []
102 threshold_value = 0.4
103
104 for i in range(len(descriptors1)):
105     # initialize an array to store our computed angles
106     angle_array = []
107
108     for x in range(len(descriptors2)):
109         # Calculate the angle between each corresponding pair of descriptors from descriptor1 and descriptor2
110         angle_array.append(math.acos(np.dot(descriptors1[i], descriptors2[x])))
111
112     # Sort the stored angles and get the smallest one
113     min_match_angle = sorted(angle_array)[0]
114
115     # The second smallest angle
116     second_min_angle = sorted(angle_array)[1]
117
118     # Calculate the threshold ratio
119     threshold_ratio = min_match_angle/second_min_angle
120
121     # Checks that the threshold ratio is satisfied
122     if (threshold_ratio < threshold_value):
123
124         # Get the key points
125         kp1 = keypoints1[i]
126         kp2 = keypoints2[angle_array.index(min_match_angle)]
127
128         # Add the pair of keypoints into our match pair list
129         matched_pairs.append([kp1, kp2])
130
131
```

Result:

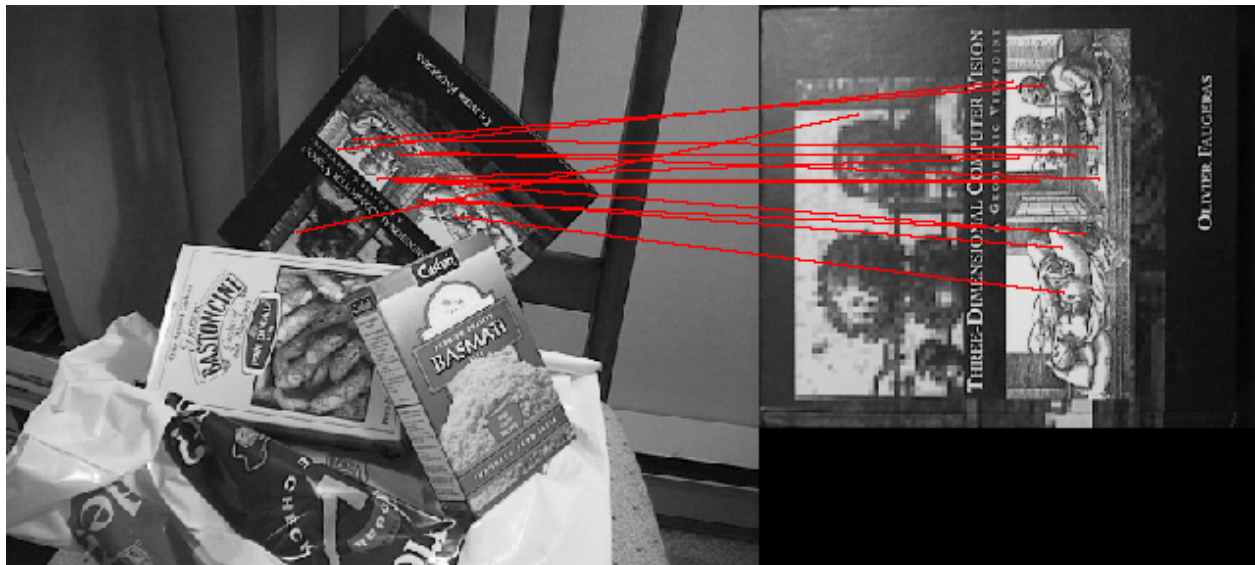
Chosen Threshold = 0.60



Test threshold = 0.80



Test threshold = 0.40



The threshold I chose for the book image was 0.60, it seems like around 0.60 we have a good amount of correct matches compared to the mismatches. When the threshold is 0.40, it seems like we have a small amount but correct matches. Whereas for threshold = 0.80, we get more matches overall but more incorrect matches.

The threshold value was important as a low threshold yields good matches but in a small amount and a high threshold yielded more of matches with a lot of the matches being incorrect.

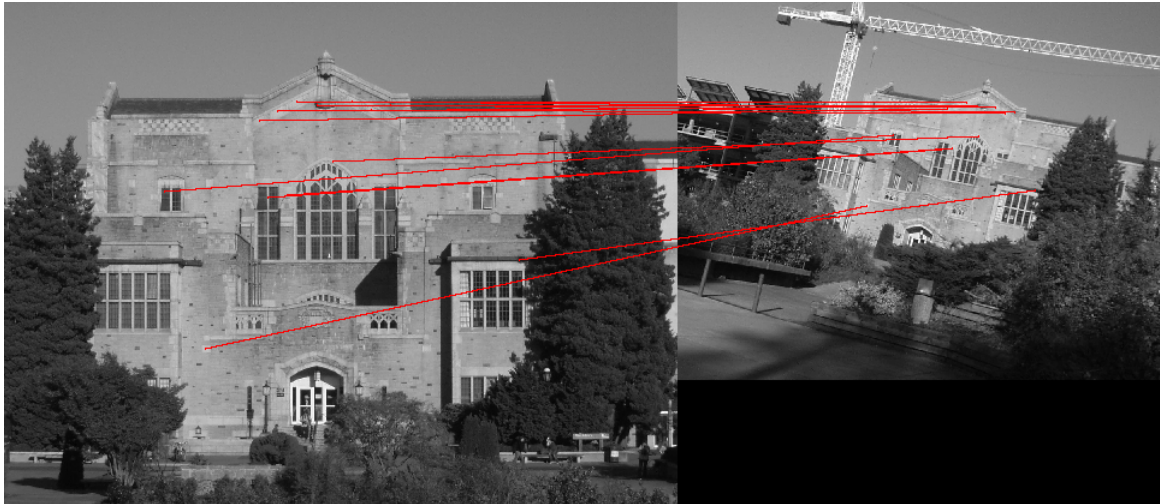
4)

Note: code is in the match function, more specifically below the code from question 3

Code:

```
134 # Initializes ransac set, orientation, scale and random selection of 10 times
135 ransacSet = []
136 orientation_val = 55
137 scale_val = 0.15
138 numRandSelection = 10
139
140 for i in range(numRandSelection):
141     sastifiedSet = []
142
143     # Apply randomization
144     rand_matched_pairs = matched_pairs[np.random.randint(len(matched_pairs))]
145
146     # Compute the change of scale and change of orientation for the first matched pair
147     # Ensures orientation is not greater than 2pi by mod pi*2 to the computed orientation change
148     scale_change1 = rand_matched_pairs[0][2] / rand_matched_pairs[1][2]
149     orientation1_change = (rand_matched_pairs[0][3] - rand_matched_pairs[1][3]) % (math.pi * 2)
150
151     for pair in matched_pairs:
152
153         # Compute the change of scale and change of orientation for the first matched pair
154         scale_change2 = pair[0][2] / pair[1][2]
155         orientation2_change = (pair[0][3] - pair[1][3]) % (math.pi * 2)
156
157         # Get the difference in scale and orientation between the two matched pairs
158         scale_ratio_difference = abs(scale_change1 - scale_change2)
159         orientation_difference = (orientation1_change - orientation2_change) % (math.pi * 2)
160
161         # Deal with the case when the difference in angle between the two pairs is greater than pi
162         if (orientation_difference > math.pi):
163
164             # Subtract pi if the difference in orientation is greater than pi
165             orientation_difference = orientation_difference - math.pi
166
167         # Ensures the thresholds are sastified
168         # Convert orientation from degree to radians
169         if ( scale_ratio_difference <= scale_val and orientation_difference <= orientation_val * (math.pi/180)):
170
171             # Add it to our set if threshold sastified
172             sastifiedSet.append(pair)
173
174         # We want the largest set, thus we set the ransac set as the largest set
175         if(len(ransacSet) < len(sastifiedSet)):
176             ransacSet = sastifiedSet
177
178     # Displays our new Ransac set
179     im3 = DisplayMatches(im1, im2, ransacSet)
180
181
```

Result: orientation = 55 degrees and scale = 15%



I found that changing the orientation threshold to around 55 degrees and scale threshold to 15% keeps the consistency of correct matches while keeping the outliers low. A higher threshold values yielded more overall matches but it also increases the number of outliers. Whereas a smaller threshold, yields less matches with more consistency incorrect matches.