Keerthi Raj Nagaraja (nagaraj1@purdue.edu)

10-01-2014

1 Harris Corner Detector

Harris corner detector finds interest points in the given image where the gradient levels vary largely in at least two different directions.

1.1 Detecting Corners

Steps involved in Harris Corner Detector are:

- Step-1: Calculate x and y derivatives

  Using a Sobel operator of size 3 or 5, x and y derivatives of the given image are found and stored separately. 3x3 sobel operators used to find dx and dy images are respectively,

  $\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$ and $\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

- Step-2: Find Correlation Matrix

  Using a $5\sigma \times 5\sigma$ window for each pixel, we find the correlation matrix C using the dx and dy images.

  $$C = \begin{bmatrix} \Sigma dx^2 & \Sigma dxdy \\ \Sigma dxdy & \Sigma dy^2 \end{bmatrix}$$

  If the pixel under consideration is a true corner, then C will have full rank.

- Step-3: Find Corner strength Large values of diagonal elements and small values of cross-diagonal elements in C indicates large gradient change in x and y directions and more the chances of pixel being a corner. Hence we quantify the corner strength as,

  $$Cornerstrength = det(C) - k(trace(C)^2)$$

  Corner strength is large for true corners, small for constant regions and negative for edges.

- Step-4: Thresholding and Non-maximum Suppression We retain pixels whose corner strength exceeds chosen threshold value. Later, to eliminate corners which are close to each other, we do Non-maximum suppression where only the corner points which are locally maximum in a given window are retained and others are suppressed.

1.2 Finding correspondences between two images of same scene

Given two images of same scene which may have in plane rotation, we would like to find corresponding corners in both the images. We use harris corner detector to detect corners in both the images. Then, we establish a feature descriptor which captures the information contained in a defined neighborhood of the corner pixel. We can use two different feature descriptors for this:

- Normalized Cross Correlation (NCC)

$$NCC = \frac{\Sigma\Sigma(f_1(i,j) - m_1)(f_2(i,j) - m_2)}{\sqrt{[\Sigma\Sigma(f_1(i,j) - m_1)^2][\Sigma\Sigma(f_1(i,j) - m_1)^2]}}$$

  Where, f_1 and f_2 are the pixels values of the window around the corner pixels in image 1 and 2 respectively. m_1 and m_2 are the mean pixel values of f_1 and f_2.

  We find NCC for each pair of corner points in image 1 and 2. When there is a similarity, then NCC approaches 1. We exhaustively search through the NCC values to maximize the NCC for a given corner pixel in image 1 to find a match in image 2. Harris Corner Detector need not always detect the same interest point in the two images and hence we choose points whose NCC values exceed certain threshold.

  Also, there may be non-unique features around a corner pixel and this leads to false correspondences. In such cases, the value of maximum NCC will be very close the second maximum and hence find the ratio of second maximum NCC to maximum NCC and discard the point if

$$\frac{SecondMaxNCC}{MaxNCC} > r_{ncc}$$

- Sum of Squared differences (SSD)

$$SSD = \Sigma\Sigma|f_1(i,j) - f_2(i,j)|^2$$

Where, f_1 and f_2 are the pixels values of the window around the corner pixels in image 1 and 2 respectively.

We find SSD for each pair of corner points in image 1 and 2. When there is a similarity, then SSD approaches 0. We exhaustively search through the SSD values to minimize the SSD for a given corner pixel in image 1 to find a match in image 2. Again, Harris Corner Detector need not always detect the same interest point in the two images and hence we choose points whose SSD values are less than certain threshold.

Also, false correspondences are reduced in the same way as we did in NCC. In such cases, the ratio of minimum SSD to second minimum SSD will be very close 1 and we discard the point if

$$\frac{MinSSD}{SecondMinSSD} > r_{ssd}$$

1.3 Parameters used

The following parameters are tuned for the best extraction of interest points from the Harris Corner Detector.

Image	W_s	W_h	W_r	W_{ssd}	W_{ncc}	T_h	T_{ssd}	T_{ncc}	r_{ssd}	r_{ncc}	k
pic1-set1	3	5σ	5σ	53	39	1e+9	50	0.95	0.92	0.95	0.04
pic2-set1	3	5σ	5σ	53	39	2e+9	50	0.95	0.92	0.95	0.04
pic1-set2	3	5σ	5σ	45	17	2e+9	85	0.93	0.95	0.95	0.04
pic2-set2	3	5σ	5σ	45	17	7e+9	85	0.93	0.95	0.95	0.04
mypic1-myset	3	5σ	5σ	53	17	1e+9	80	0.92	0.9	0.92	0.04
mypic2-myset	3	5σ	5σ	53	17	1e+9	80	0.92	0.9	0.92	0.04

The table below explains the symbols used

Parameter	Explanation
W_s	Dimension of Sobel operator window
W_h	Dimension of Harris Correlation matrix window
W_r	Dimension of Non-maximum supression window
W_{ssd}	Dimension of feature descriptor window for SSD
W_{ncc}	Dimension of feature descriptor window for NCC
T_h	Threshold for harris corner strength
T_{ssd}	Threshold for SSD score
T_{ncc}	Threshold for NCC score
r_{ssd}	Threshold for ratio of SSD minimum to second minimum scores
r_{ncc}	Threshold for ratio of NCC second maximum to maximum scores
k	Corner strength parameter

1.4 Comparison of SSD and NCC

- NCC takes more computations. However, because we find the cross correlation by subtracting the pixel values in a window by its mean, it is more robust to illumination changes. It worked well for both set 2 and my own set where the images had significant illumination differences.

- SSD takes less computations and just relies on the absolute intensity differences and will not perform better when there is change in illumination between two images. It failed to establish correct correspondences and had significant errors in both set 2 and my own set because of illumination variation between the images. However, it worked well in set 1 where there was no illumination difference.

2 Speeded Up Robust Features (SURF)

SURF is a method of extracting features from the image's interest points. We apply OpenCV's inbuilt SURF feature Extractor to get the interest points and their corresponding feature descriptors.

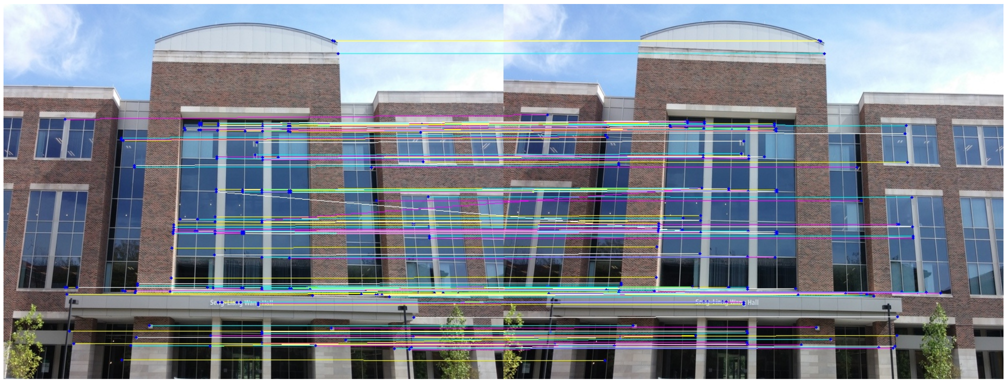
In SURF, an interest point is defined as the one where the determinant of the hessian given below is maximized at some scale σ. SURF works with a set of gaussian smoothed images $ff(x,y,\sigma)$ at different σ values. By maximizing the hessian, we are maximizing the double derivatives while minimizing the cross derivatives which discriminates the edges from true corners.

$$H(x,y,\sigma) = \begin{bmatrix} \frac{\partial^2}{\partial x^2} ff(x,y,\sigma) & \frac{\partial^2}{\partial x \partial y} ff(x,y,\sigma) \\ \frac{\partial^2}{\partial x \partial y} ff(x,y,\sigma) & \frac{\partial^2}{\partial y^2} ff(x,y,\sigma) \end{bmatrix}$$

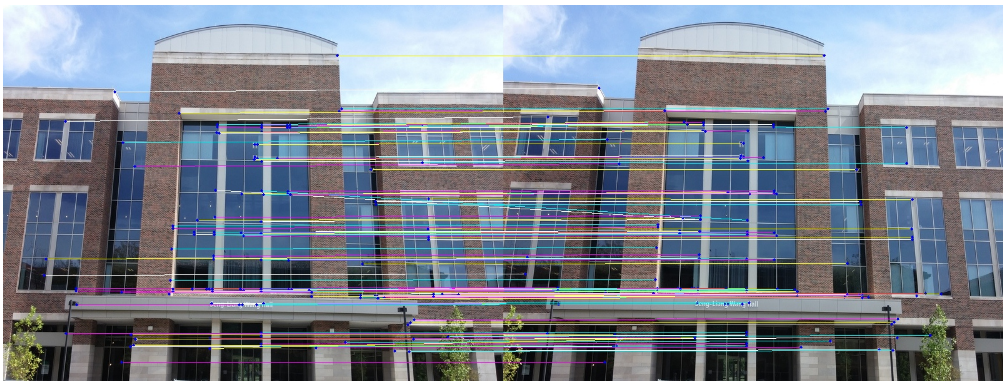
SURF is computationally fast because of the use of Integral images. The feature descriptor at each corner point is with respect to the dominant direction and hence SURF is robust to inplane rotations. The descriptor at a corner point is calculated using a $20\sigma \times 20\sigma$ window around the point. A 64 element descriptor vector is then created for each corner point. These feature descriptors can directly be used to get the euclidean distance which measures the correspondence quality. Similar points will have small euclidean distance between them. The measured distance is thresholded and the ratio of the minimum distance to second minimum distance is also thresholded similar to SSD to find the correspondences.

3 Observations

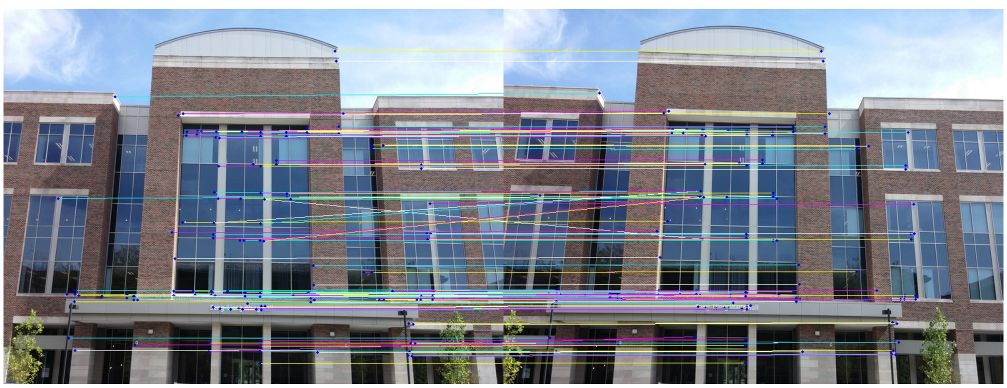
- SURF was much faster compared to Harris Corner detector
- SURF produced more false positive corners than Harris but the correspondence accuracy of SURF was better than the Harris detector with SSD and was comparable to Harris detector with NCC
- Harris detector with NCC was very efficient overall and illumination invariant
- Changing the scale in harris detector either increased or decreased the number of corners detected depending on the image. For images in set 1, as the scale increased from 1 to 2.6, the correspondences decreased monotonically. For images in set 2, same behavior was observed in NCC version. Since SSD version was not completely accurate, effect of scale was not clear. For my own set, correspondences were maximum at scale 1.4 and decreased for other scales. This behavior illustrates that different images have different amount of information at various scales.



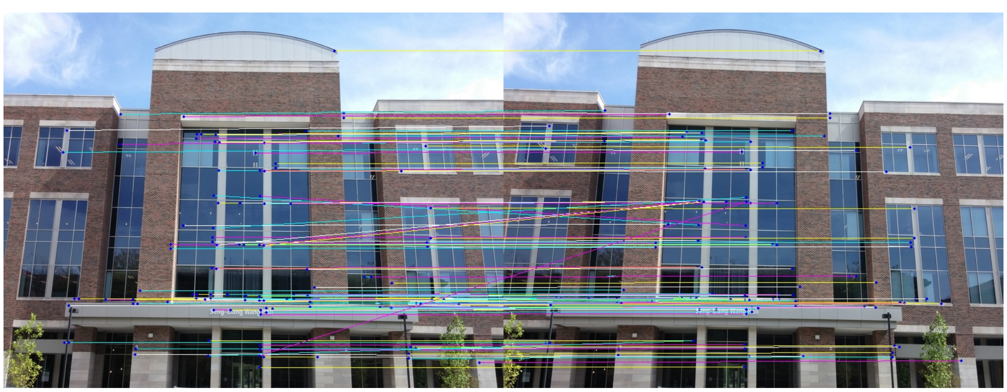
(a) Set 1, NCC, Scale=1.0, 139/141 Correct matches



(b) Set 1, NCC, Scale=1.4, 119/120 Correct matches

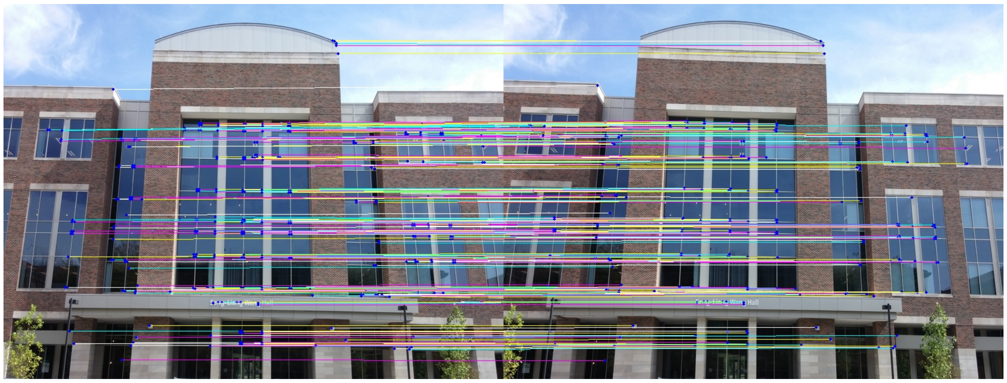


(c) Set 1, NCC, Scale=2.2, 120/125 Correct matches

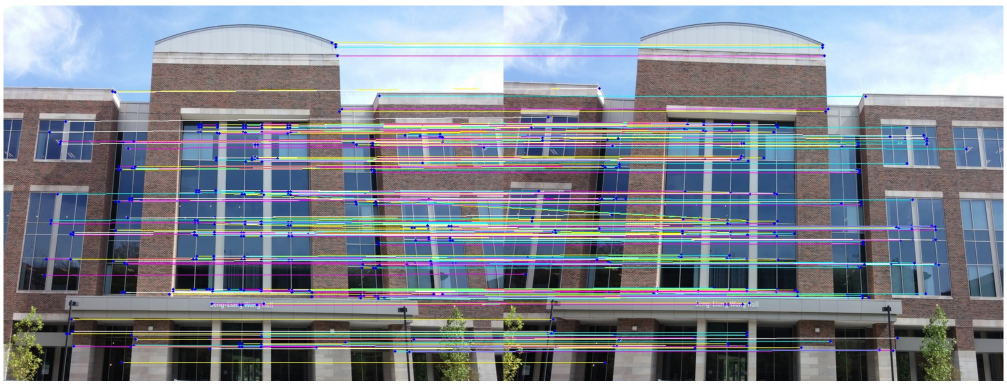


(d) Set 1, NCC, Scale=2.6, 108/113 Correct matches

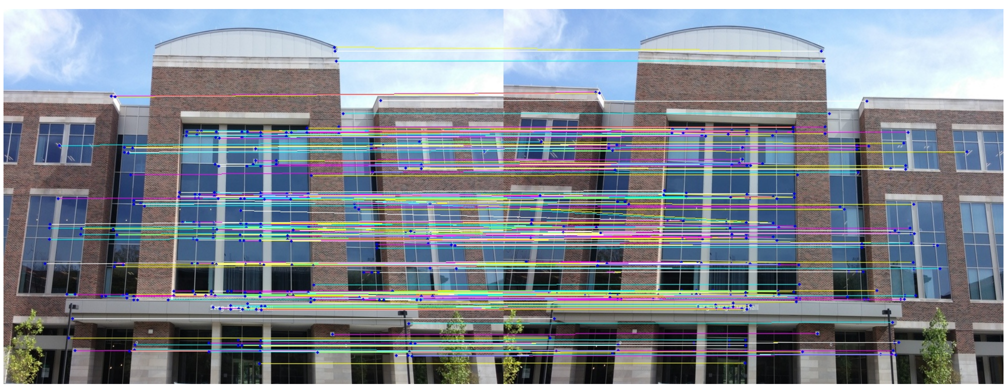
Figure 1: Set 1, NCC Method



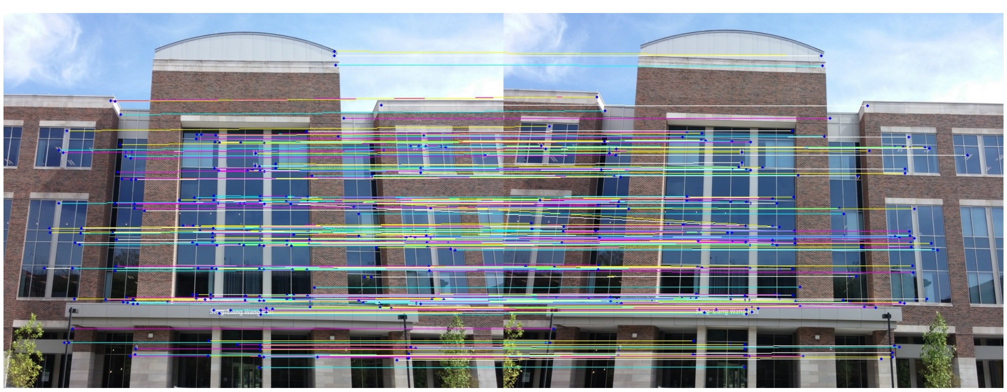
(a) Set 1, SSD, Scale=1.0, 284/284 Correct matches



(b) Set 1, SSD, Scale=1.4, 243/245 Correct matches



(c) Set 1, SSD, Scale=2.2, 223/225 Correct matches



(d) Set 1, SSD, Scale=2.6, 205/207 Correct matches

Figure 2: Set 1, SSD Method

(a) Set 2, NCC, Scale=1.0, 17/17 Correct matches

(b) Set 2, NCC, Scale=1.4, 15/15 Correct matches

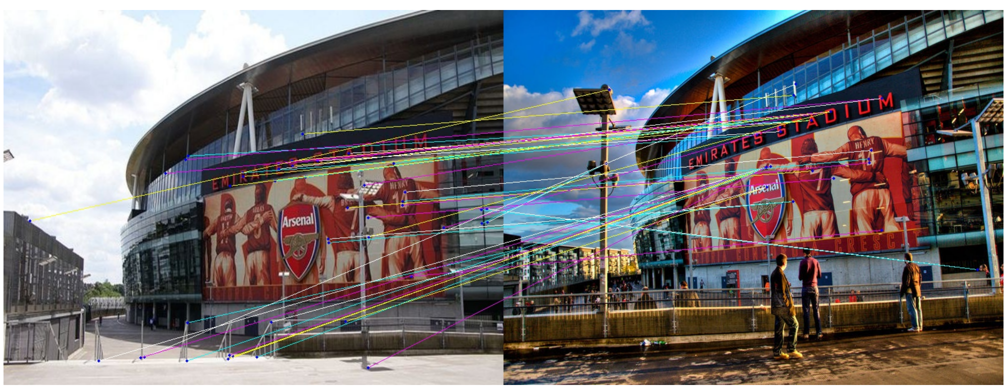
(c) Set 2, NCC, Scale=2.2, 8/8 Correct matches

(d) Set 2, NCC, Scale=2.6, 7/7 Correct matches

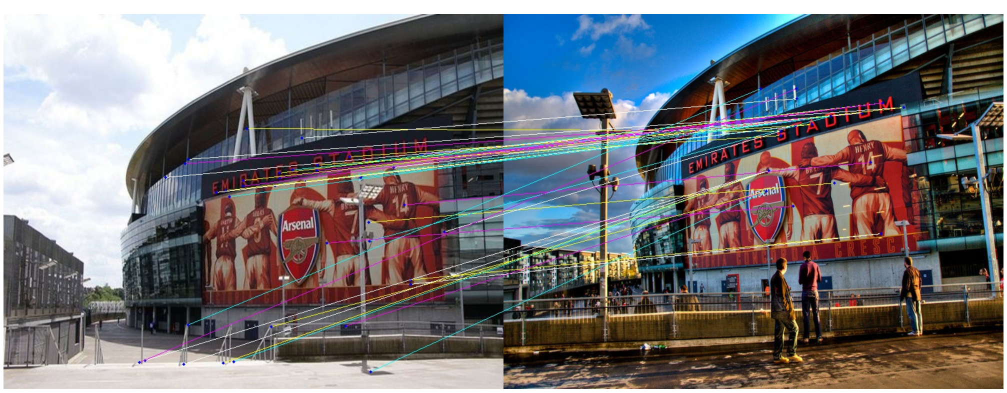
Figure 3: Set 2, NCC Method

(a) Set 2, SSD, Scale=1.0, 6/17 Correct matches

(b) Set 2, SSD, Scale=1.4, 6/23 Correct matches

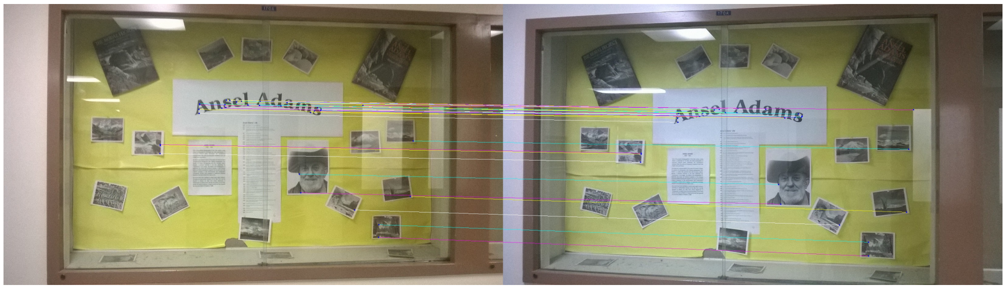


(c) Set 2, SSD, Scale=2.2, 25/45 Correct matches

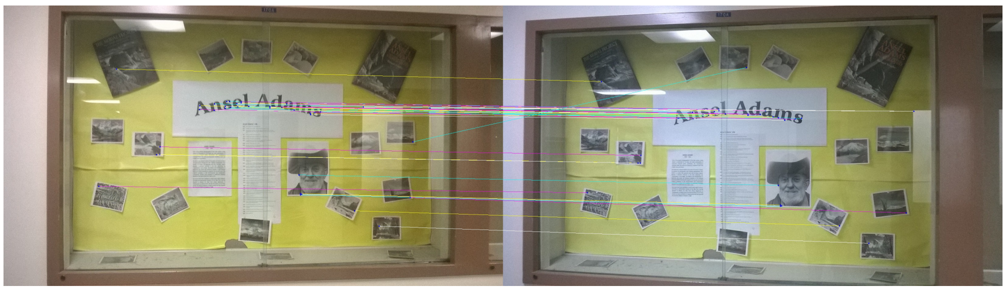


(d) Set 2, SSD, Scale=2.6, 22/39 Correct matches

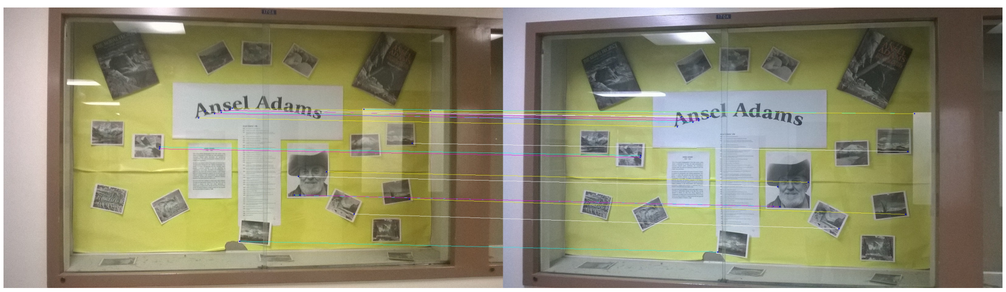
Figure 4: Set 2, SSD Method



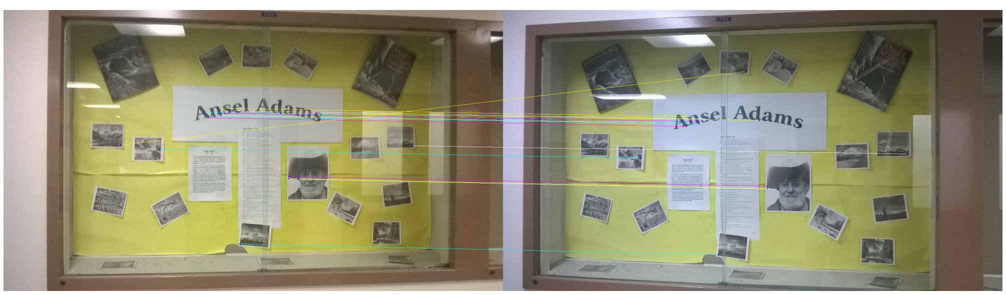
(a) Myset, NCC, Scale=1.0, 32/32 Correct matches



(b) Myset, NCC, Scale=1.4, 37/38 Correct matches

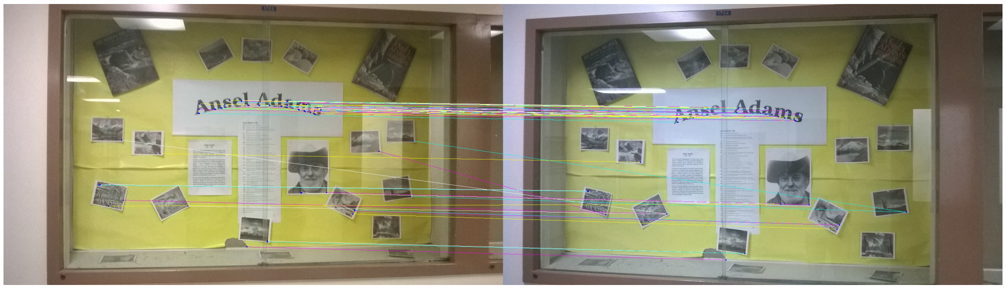


(c) Myset, NCC, Scale=2.2, 19/19 Correct matches

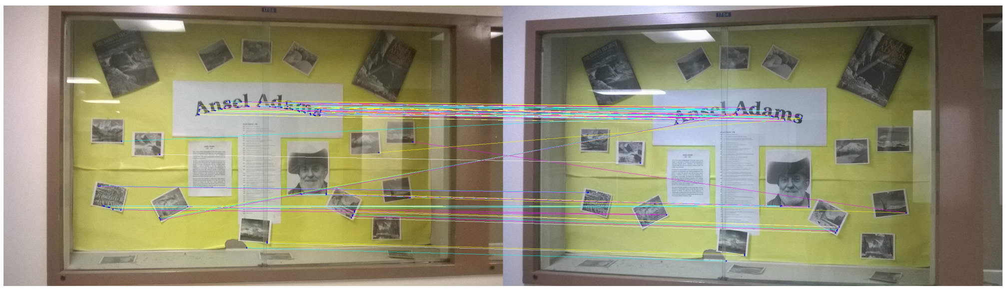


(d) Myset, NCC, Scale=2.6, 14/15 Correct matches

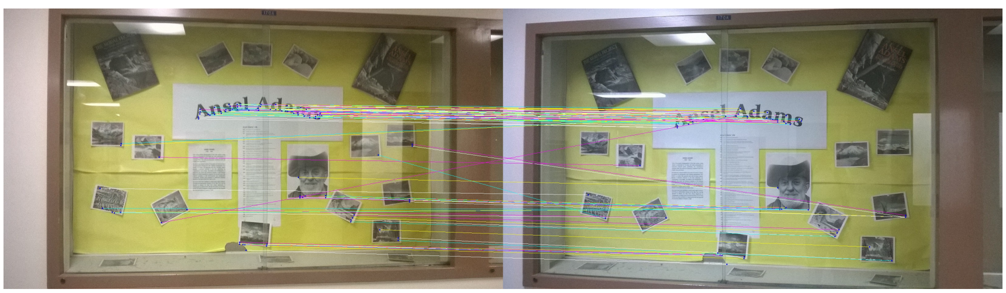
Figure 5: Myset, NCC Method



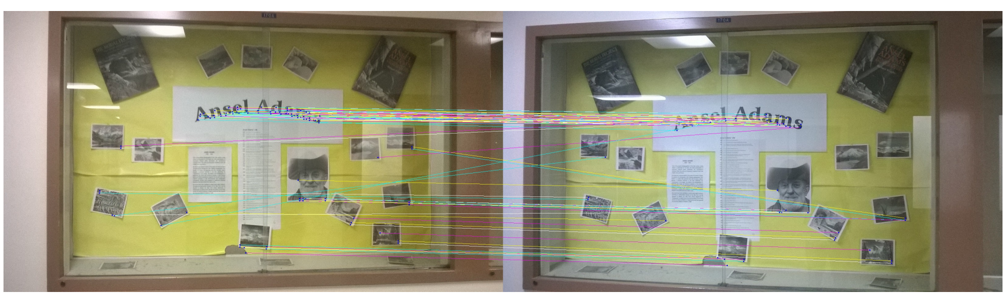
(a) Myset, SSD, Scale=1.0, 57/60 Correct matches



(b) Myset, SSD, Scale=1.4, 75/79 Correct matches

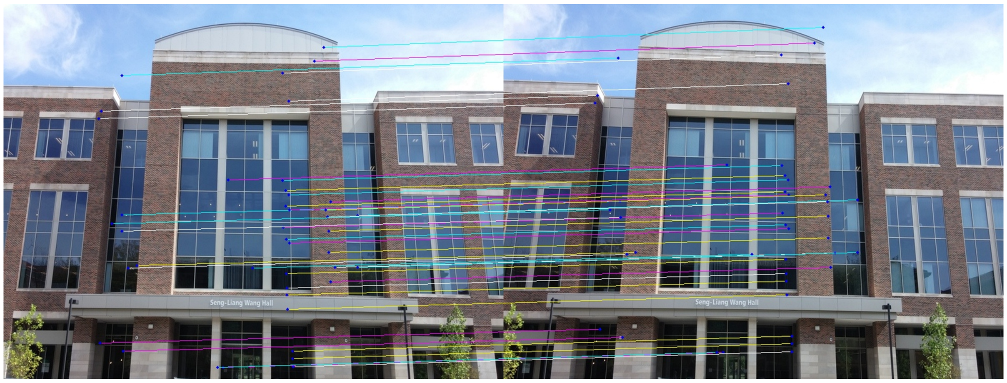


(c) Myset, SSD, Scale=2.2, 84/90 Correct matches



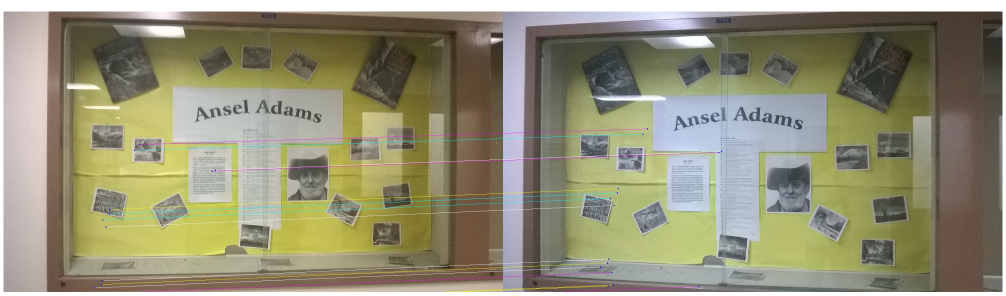
(d) Myset, SSD, Scale=2.6, 80/88 Correct matches

Figure 6: Myset, SSD Method



(a) Set 1, $H_{th} = 2400$, Distance Threshold= 0.05, Ratio Threshold=0.9

(b) Set 2, $H_{th} = 1400$, Distance Threshold= 0.13, Ratio Threshold=0.9



(c) Myset, $H_{th} = 3000$, Distance Threshold= 0.1, Ratio Threshold=0.9

Figure 7: SURF Results