ASSIGNMENT-I

Assumption: The results of the canny edge detector and harris corner detector have been compared with the outputs of the OpenCV standard functions cv2.Canny() and cv2.cornerHarris() for calculation of the following quantitative metrics.

TP = True Positives
FP = False Positives
TN = True Negatives
FN = False Negatives

Accuracy = (TP+TN)/(TP+FN+TN+FP)
F Score = (2*TP)/((2*TP)+FP+FN)

src: building.jpg

Experiment results on Canny Edge Detector:

output: canny_<gaussian_ksize>_<sobel_ksize>_<t>_<T>.png

Sl.No	gaussian_ksize	sobel_ksize	t	T	Metrics	Observations
1	5	5	50	150	Accuracy = 0.86517 F Score = 0.45943	The difference in metrics is due to the the large number of true negatives.
2	5	5	50	200	Accuracy = 0.87661 F Score = 0.46939	As *T* is increased, both the true positives and true negatives increase slightly.
3	5	5	200	250	Accuracy = 0.90117 F Score = 0.41224	The high increase in accuracy and drop in F Score is owed to the large increase in true negatives. This is because the number of edge pixels decrease with increase in *t* and *T*. Low value of F Score is due to the less number of edge pixels, where a small difference in edge indices affects the metric to a large extent.
4	3	5	50	150	Accuracy = 0.870110 F Score = 0.50959	(1) and (4) differ only by the Gaussian Kernel size but show a great difference in F Score. This shows the effect of denoising on the image. Also, there is **no control** of Gaussian kernel size in OpenCV implementation of Canny Edge, which implies that the same canny image is used as ideal image

						for comparison in both (1) and (4).
5	5	3	50	200	Accuracy = 0.89776 F Score = 0.28700	(2) and (4) differ by only the Sobel kernel size but show a great fall in F score. Also, the output image shows very few edge pixels due to high value of T and low value of Sobel kernel. This shows that the sobel kernel size is important for edge pixels to be detected correctly and is an important parameter to be chosen correctly.

The above experiments show that (4) is the best set of parameters for the current input image. It is due to the optimal Gaussian kernel size and a not very low sobel kernel size. The number of edge pixels in this case is the highest in comparison to other cases.

Experiment results on Harris Corner Detector:

output: harris_<window_size>_<sobel_ksize>_<k>_<t>.png

Sl. No	window_size	sobel_ksize	k	t	Metrics	Observations
1	5	5	0.04	0.01	Count = 11018 Accuracy = 0.96257 F Score = 0.74747	The *t* value detects most of the corner points as can be seen in the corresponding output image.
2	5	5	0.04	0.04	Count = 4262 Accuracy = 0.98093 F Score = 0.686417	The increase in *t* leads to a large decrease in the number of corner points due to which a small difference in the corner point indices also affects the F Score largely.
3	3	5	0.04	0.01	Count = 6719 Accuracy = 0.98003 F Score = 0.74627	(1) and (3) differ by *window_size*. This shows that the *window_size* parameter is one of the most important parameters after *t* which decides the number of corner points detected.
4	5	3	0.04	0.01	Count = 11018 Accuracy = 0.96513 F Score = 0.78667	(1) and (4) differ by Sobel kernel size. Although the number of corner point count did not change, the

						F Score shows an increase implying an increase in the number of correctly classified points. Thus, sobel kernel size plays an important role after window size.
5	5	3	0.06	0.01	Count = 10010 Accuracy = 0.96525 F Score = 0.76901	The increase in k from (4) to (5) leads to a slight decrease in the number of corner points and F Score.

The above experiments show that (4) is the best set of parameters for the current input image. However, if a lower number of corner points need to be detected, either the *window_size* or the threshold fraction, *t* could be lowered accordingly as can be seen above.