Assignment2

27th February 2019

Prepared by

K PUNEETH

IMT2016018

1)

Part1:-

The first part of the question is to stitch a panorama. The first set of input images are:-

To stitch these images together, we will be using the SIFT feature in opencv. After this, we use the function ComputeAndDetect of SIFT to find the keypoints and the descriptors. We require only Descriptors only so, we ignore the keypoints. Now we have to compare the features present in the two images. We want only comparable features. For that, we use the BF Matcher function with knn matching present in opencv. This gives us a list of matches but, we want only the good ones. So, we filter them such that the ratio of distances is less than 0.7. We use the homography function to avoid the overlapping at different angles. Now, we have both the images and the Homography image. We have to attach them to get the output. For this we use the warp perspective function. The final output will look like this.

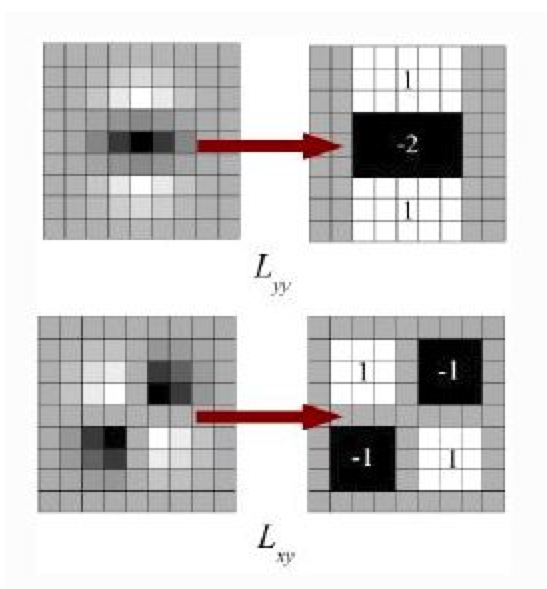
The second set of images looks like this.

The panorama of the image looks like this.

Part2:-

The second part of the question is about how SURF is different from SIFT. The differences are:-

- SURF is 3 times faster than SIFT in terms of performance comparison. In fact, SURF was created to improve the features of SIFT.
- In SIFT, Lowe approximated Laplacian of Gaussian with Difference of Gaussian for finding scale-space. SURF goes a little further and approximates LoG with Box Filter. Below image shows a demonstration of such an approximation.



- The size of the descriptor is 128 bits in SIFT but in SURF the size of the descriptor is 64 bits.
- SIFT uses the local maxima of the neighboring points to calculate a potential key point. SURF sign of Laplacian (trace of Hessian Matrix) for underlying interest point.
- For orientation assignment, SURF uses wavelet responses in a horizontal and vertical direction for a neighborhood of size 6s, while SIFT creates an orientation histogram with 36 bins covering 360 degrees

Part3:-

The third part is about the main principles of RANSAC/FLANN matching.

RANSAC MATCHING:-

- **Random sample consensus** (**RANSAC**) is an iterative method to estimate parameters of a mathematical model from a set of observed data that contains outliers when outliers are to be accorded no influence on the values of the estimates.
- The RANSAC algorithm is essentially composed of two steps that are iteratively repeated:
    1. In the first step, a sample subset containing minimal data items is randomly selected from the input dataset. A fitting model and the corresponding model parameters are computed using only the elements of this sample subset. The cardinality of the sample subset is the smallest sufficient to determine the model parameters.
    2. In the second step, the algorithm checks which elements of the entire dataset are consistent with the model instantiated by the estimated model parameters obtained

from the first step. A data element will be considered as an outlier if it does not fit the fitting model instantiated by the set of estimated model parameters within some error threshold that defines the maximum deviation attributable to the effect of noise.

- The set of inliers obtained for the fitting model is called the consensus set. The RANSAC algorithm will iteratively repeat the above two steps until the obtained consensus set in certain iteration has enough inliers.
- The input to the RANSAC algorithm is a set of observed data values, a way of fitting some kind of model to the observations, and some confidence parameters. RANSAC achieves its goal by repeating the following steps:
    1. Select a random subset of the original data. Call this subset the *hypothetical inliers*.
    2. A model is fitted to the set of hypothetical inliers.
    3. All other data are then tested against the fitted model. Those points that fit the estimated model well, according to some model-specific loss function, are considered as part of the *consensus set*.
    4. The estimated model is reasonably good if sufficiently many points have been classified as part of the consensus set.
    5. Afterward, the model may be improved by reestimating it using all members of the consensus set.

This procedure is repeated a fixed number of times, each time producing either a model which is rejected because too few points are part of the consensus set, or a refined model together with a corresponding consensus set size. In the latter case, we keep the refined model if its consensus set is larger than the previously saved model.

- An advantage of RANSAC is its ability to do a robust estimation of the model parameters, i.e., it can estimate the parameters with a high degree of accuracy even when a significant number of outliers are present in the data set. A disadvantage of RANSAC is that there is no upper bound on the time it takes to compute these parameters (except exhaustion). When the number of iterations computed is limited the solution obtained may not be optimal, and it may not even be one that fits the data in a good way. In this way RANSAC offers a trade-off; by computing a greater number of iterations the probability of a reasonable model being produced is increased.

FLANN MATCHING:-

- FLANN stands for Fast Library for Approximate Nearest Neighbors. It contains a collection of algorithms optimized for fast nearest neighbor search in large datasets and for high dimensional features. It works faster than BFMatcher for large datasets.

2)

Procedure:-

- First, take the images into lists and read them using opencv function imread().

- Using SIFT find the descriptors of the images into a list using the function ComputeAndDetect().
- Take the list of descriptors and apply kmeans to get clusters and centroids of the image.
- For the clusters obtained, create a histogram for each descriptor with the number of clusters as the range and the number of descriptors having that cluster.
- For horses and bikes add a label for each of them. This will be the expected output label. Divide the data into train and test.
- Pass the histogram as input and the expected output label using SVM/Linear Regression/KNN. This will train the model. Similarly, create a histogram with test data and run the model on it to get the output.
- Use accuracy_score() to get the accuracy of model.

My Approach:-

- While reading the lists I have put the labels for bikes as ‘0’ and for horses, I have put them as ‘1’.
- In CIFAR10 dataset the labels are given so I have directly extracted the train data and test data with labels.
- The types of images have to be converted in CIFAR10 dataset to uint8 for images to be read and manipulated.
- I have SIFT to extract the descriptors and used Kmeans with 36 clusters for Bikes and Horses classification. For CIFAR10 I have used 100 clusters to get best results.
- After that I divided the first ⅓ data in Bikes and Horses as Test and remain as Train. CIFAR10 didn’t require this.
- After that, I created Histograms for Test and Train in both datasets and used SVC, Logistic Regression and KNN algorithms to create models.

Observations:-

- For Bikes and Horses dataset I got best results using Logistic Regression with an accuracy score of 0.864406779661017
- For CIFAR10 dataset I got best results using Logistic Regression with an accuracy score of 0.2677.
- If the size of images can be enlarged using interpolation then we will get more descriptors and hence good accuracy score.