Shashank Reddy
IMT2018069
shashank.reddy@iiitb.org

Final Report
AI 825 - Visual Recognition

2021-02-24

1 Introduction

This assignment consists of 2 main problems, initially we try to stitch similar images by generating interset points, and then matching them. Later, we try to build an image classification model for the CIFAR-10 dataset. To do this, we first generate interest points, we then cluster them to form our visual dicitonary, and then based on the method, that is, BOW or VLAD, we generate descriptors for each image. We then train classical models such as SVM, Logistic Regression and KNN Classifier to classify the image.

2 Panorama

The original images are-

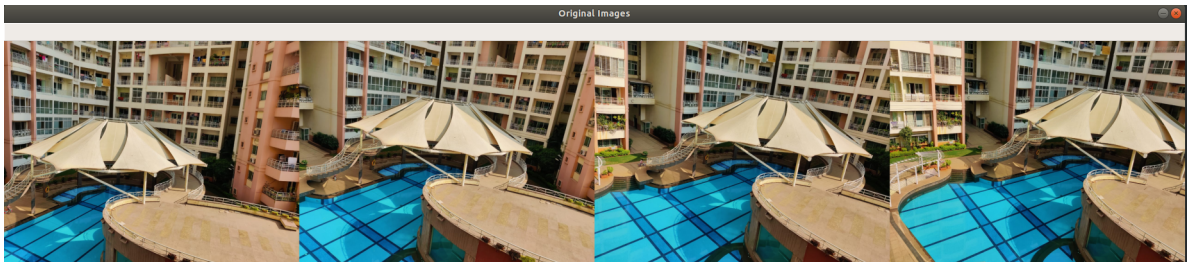

Figure 1: Original Set of images

Although I stitched multiple images together, I basically defined a method to stitch 2 images together and called this function repeatedly. Thus, I will show some of the key steps in stitching 2 images together.

2.1 Interest points

To generate the interest points in the image, we can use multiple detectors such as Harris, SIFT, SURF, etc. In this case we use SIFT (Scale Invariant Feature Transform) and SURF(Speeded-Up Robust Features).
Some key differences between the 2 detectors are-

1. One subtle difference is that in SIFT we use the Difference of Gaussian (DoG) over different sizes of images with the same sized filter to approximate the Laplacian of Gaussian (LoG). However, in SURF we use the Laplacian of Gaussian over an integral image.
2. SIFT returns a descriptor of 128 dimensions and SURF returns a descriptor of 64 dimensions.
3. As the name suggests, SURF is in theory faster than SIFT.

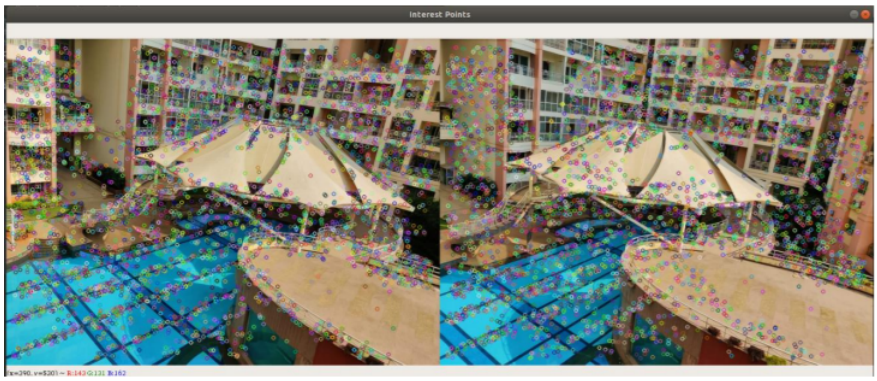

Figure 2: Interest Points generated by SURF

2.2 Matching the Interest Points

After generating the interest points, we need to look for interest points that point to the same points in both the images, for this we need to use a matching algorithm. Here, we use the FLANN (Fast Library for Approximate Nearest Neighbors) Matching Algorithm.
FLANN (Fast Library for Approximate Nearest Neighbors) is a library for performing fast approximate nearest neighbor searches in high dimensional spaces. In this case we are talking about the descriptors in the 128-dim space. It contains a collection of algorithms we found to work best for nearest neighbor search and a system for automatically choosing the best algorithm and optimum parameters depending on the dataset.
The algorithms available are-

1. FLANN_INDEX_LINEAR
2. FLANN_INDEX_KDTREE
3. FLANN_INDEX_KMEANS
4. FLANN_INDEX_COMPOSITE
5. FLANN_INDEX_KDTREE_SINGLE
6. FLANN_INDEX_HIERARCHICAL
7. FLANN_INDEX_LSH

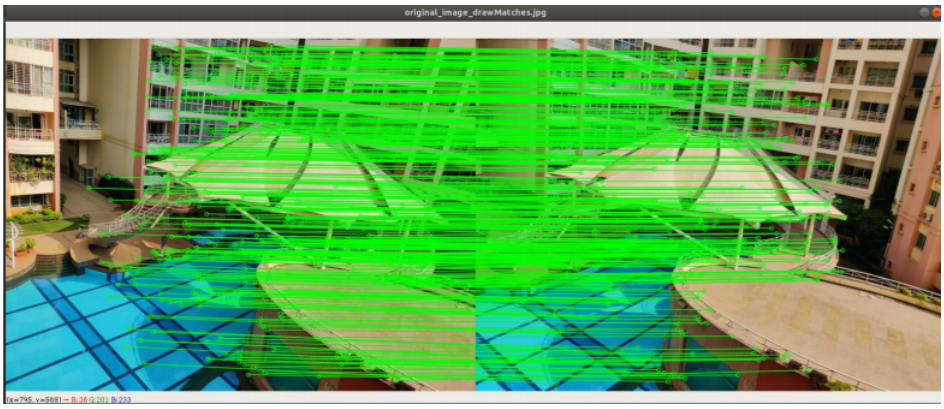

Figure 3: Matching the interest points using FLANN Matching

2.3 Finding the homography-

A Homography is a transformation (a 3×3 matrix) that maps the points in one image to the corresponding points in the other image. Thus, this will help us map the points in the first image to the points in the second image. A homography can be computed when we have 4 or more corresponding points in two images.
Two images of a scene are related by a homography under two conditions.

1. The two images are that of a plane.
2. The two images were acquired by rotating the camera about its optical axis. We take such images while generating panoramas.

It is not uncommon for 20-30% of the matches to be incorrect. Thus, the findHomography method utilizes a robust estimation technique called Random Sample Consensus (RANSAC) which produces the right result even in the presence of large number of bad matches.
RANSAC is a parameter approximation technique designed to work when we have a lot of outliers. Also, unlike conventional sampling techniques that use as much of the data as possible to obtain an initial solution and then proceed to prune outliers, RANSAC uses the smallest set possible and proceeds to enlarge this set with consistent data points.

Algorithm 1 RANSAC

1: Select randomly the minimum number of points required to determine the model parameters.
2: Solve for the parameters of the model.
3: Determine how many points from the set of all points fit with a predefined tolerance ϵ.
4: If the fraction of the number of inliers over the total number points in the set exceeds a predefined threshold τ, re-estimate the model parameters using all the identified inliers and terminate.
5: Otherwise, repeat steps 1 through 4 (maximum of N times).

Figure 4: Ransac Algorithm

2.4 Warp Perspective

Once an accurate homography has been calculated, the transformation can be applied to all pixels in one image to map it to the other image.

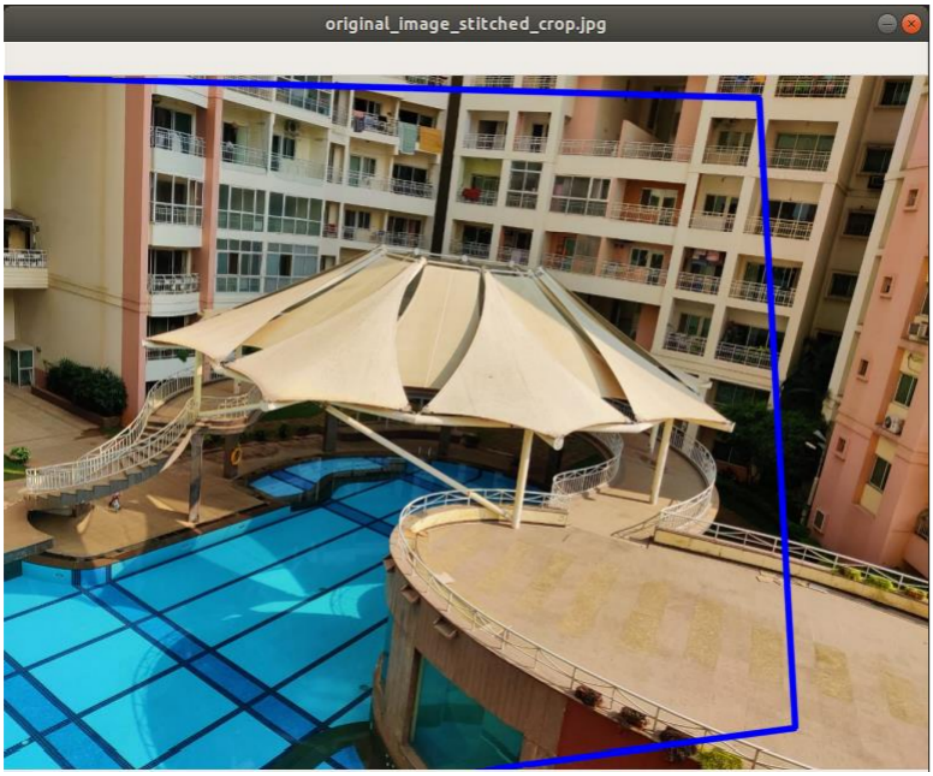

Figure 5: Warp Perspective

2.5 Final Panorama

The final image generated by stitching all the images together is given below.

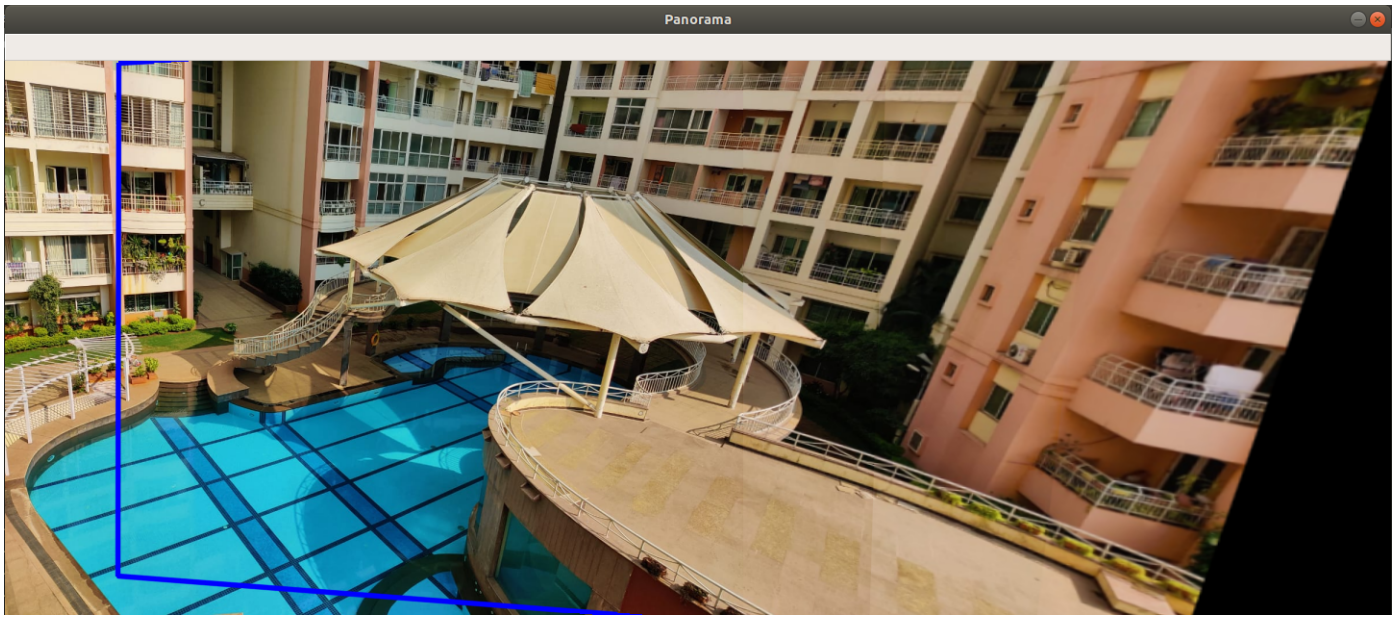

Figure 6: Final Panorama

3 Image Classification - CIPHAR10 data set

In this problem we try to build a model for image classification. We use 2 approaches in this problem, the Bag-of-Words and VLAD. (Vector of Locally Aggregated Descriptors).
First, we will look at some of the processes that are common for both and then we will talk about these 2 approaches.

3.1 Interest Points-

First we generate interest points for each of the images. For this we can use either SIFT or SURF. I found that the models perform better when using SIFT, maybe due to the fact that SIFT generates more interest points. This also explains why generating the model in the case of SIFT takes significantly more time. We then aggregate all the generated descriptors and cluster them.

3.2 Clustering

I used the k-means algorithm to cluster the descriptors. I initially started with 32 cluster centres. I then iteratively doubled the number of cluster centres upto 512 and recorded the performance of the model.

3.3 Bag-of-Words (BoW)

In this approach, we define the cluster centres we obtain as a Visual Vocabulary (Words). Next, represent each image as a histogram of these words. That is we assume that an image is a Bag of Words. This can easily be done by mapping each descriptor to a cluster center and then generating a frequency histogram for the number of maps to each cluster. This set of histograms becomes our new dataset. In this case, each datapoint is of the size k, where k is the number of cluster centres.

3.4 Vector of Locally Aggregated Descriptors (VLAD)

This approach is an extension to the Bag of Words approach. In case of BoW, we only store the number of matches to each cluster centre, however in this approach, we try to add more information by storing the sum of residuals for each cluster. This can be calculated by summing up the difference between the cluster centre and the descriptor for all descriptors mapped to a cluster centre. This set of residuals becomes our new dataset. Thus, we can see that we are capturing much more information as the size of each datapoint is now the order of k * (dim(descriptors)).

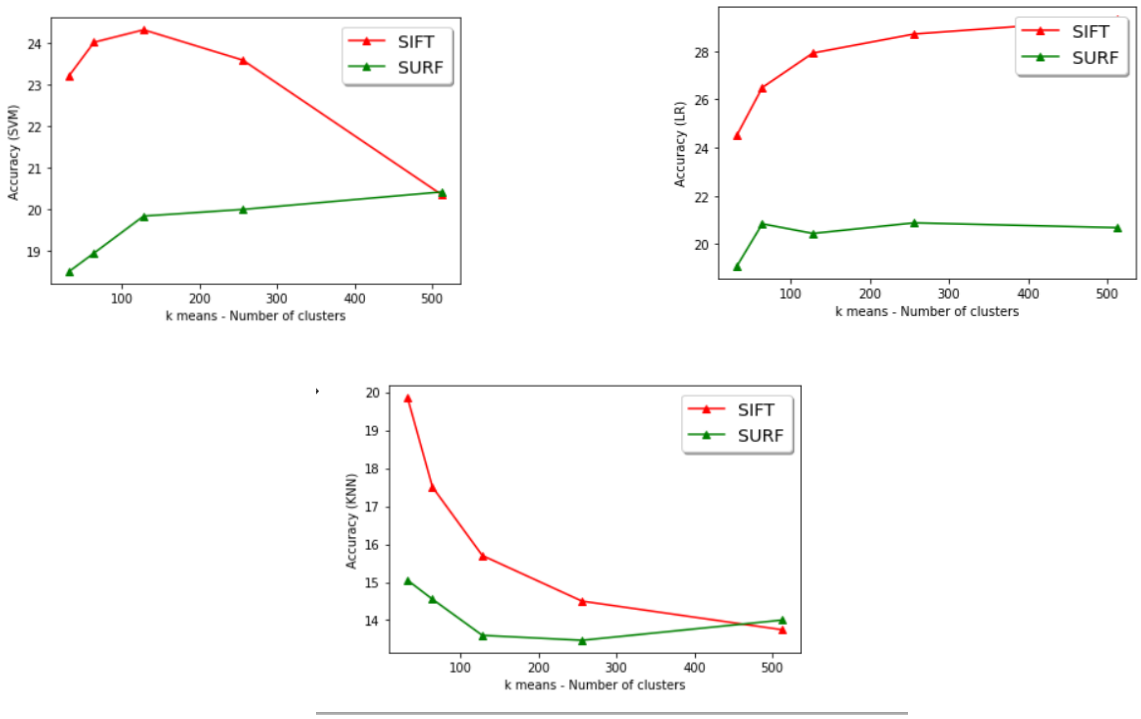

Figure 7: Performance of Models vs Size of the Vocabulary (BoW)

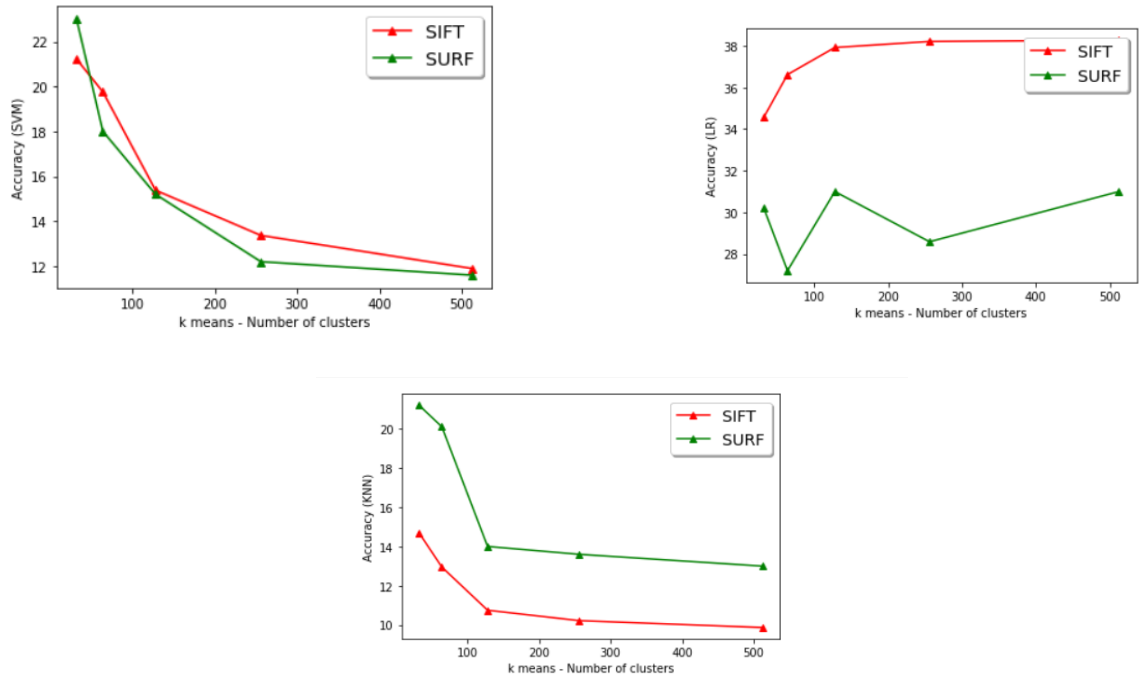

Figure 8: Performance of Models vs Number of Clusters (VLAD)

3.5 Models

I used 3 models to perform the classification, Support Vector Machines (SVMs), Logistic Regression (LR) and K-Nearest Neighbours Classifier (KNN).
I found that the Logistic Regression Model performed the best in these models, in both the cases, that is BOW and VLAD.
For the BOW approach, In the case of SVMs and LR, I found that in most cases the accuracy was increasing with the number of cluster centres, which is expected as more number of cluster centres form a better Visual Dictionary as the words can be more specific to certain features. However, surprisingly in the case of KNN I found that the accuracy of the model was reducing when I increased the number of clusters. I could not come

up with a logical explanation for this.
In the VLAD case, I observed that this was the case for most of the models, that is, the accuracy decreased with the increase in the number of clusters. However, you can see an improvement in the score when using VLAD when compared to BOW as expected, since BOW contains more information about each image.