**Part 1**

I have used the subsampling size of 80 X 80 ,  and converted that to 1600 for flattened vector for each image.

2. Used the svm classifier as given in the assignment and trained the model and tested model. Parsed the output files. Used the system to execute and the results are parsed.

3. 43 correct / 250 mages => color

Grey 27/250

color images performed on our svm train and test.

Color image:

Final epsilon on KKT-Conditions: 0.09809

Upper bound on duality gap: 0.09812

Dual objective value: dval=0.32340

Primal objective value: pval=0.42151

Total number of constraints in final working set: 601 (of 889)

Number of iterations: 890

Number of calls to 'find\_most\_violated\_constraint': 46250

Number of SV: 576

Norm of weight vector: |w|=0.80427

Value of slack variable (on working set): xi=0.02469

Value of slack variable (global): xi=0.09809

Norm of longest difference vector: ||Psi(x,y)-Psi(x,ybar)||=10958.65958

Runtime in cpu-seconds: 1287.93

Final number of constraints in cache: 6250

Compacting linear model...done

Writing learned model...done

[rtawde@silo gmadhwan-rtawde-nsuri-a3]$ ./a3 testing baseline

a3 | main :: info : main function started

Error: Can't find directory testing

[rtawde@silo gmadhwan-rtawde-nsuri-a3]$ ./a3 test baseline

a3 | main :: info : main function started

a3 | main :: info : files loaded

SVMBasedAlgo | train :: info : inside function

SVMBasedAlgo | write\_map\_to\_file :: info : inside function

Reading model...done.

Reading test examples... (250 examples) done.

Classifying test examples...done

Runtime (without IO) in cpu-seconds: 0.50

Average loss on test set: 82.8000

Zero/one-error on test set: 82.80% (43 correct, 207 incorrect, 250 total)

Grayscale image:

Final epsilon on KKT-Conditions: 0.09392

Upper bound on duality gap: 0.09405

Dual objective value: dval=1.82950

Primal objective value: pval=1.92356

Total number of constraints in final working set: 870 (of 1568)

Number of iterations: 1569

Number of calls to 'find\_most\_violated\_constraint': 55000

Number of SV: 838

Norm of weight vector: |w|=1.91293

Value of slack variable (on working set): xi=0.02578

Value of slack variable (global): xi=0.09392

Norm of longest difference vector: ||Psi(x,y)-Psi(x,ybar)||=2941.50637

Runtime in cpu-seconds: 958.77

Final number of constraints in cache: 6250

Compacting linear model...done

Writing learned model...done

[rtawde@silo gmadhwan-rtawde-nsuri-a3]$ vim SVMBasedAlgo.h

[rtawde@silo gmadhwan-rtawde-nsuri-a3]$ ./a3 test baseline

a3 | main :: info : main function started

a3 | main :: info : files loaded

SVMBasedAlgo | train :: deep info : inside function

SVMBasedAlgo | write\_map\_to\_file :: info : inside function

Reading model...done.

Reading test examples... (250 examples) done.

Classifying test examples...done

Runtime (without IO) in cpu-seconds: 0.15

Average loss on test set: 89.2000

Zero/one-error on test set: 89.20% (27 correct, 223 incorrect, 250 total)

**Part 2.1- Eigen- PCA:**

* How to run the program (after executing the make command):
  + Training:   ./a3 train Eigen
  + Test: ./a3 test Eigen
* Magic Parameters: ( Eigen.h file)
  + Subsample size: 40x40
  + Top K = 400 top Eigen vectors as our PCA vectors
  + %25 used in the function Write to file. %25 says how many files are there in the each folder. Once we get the final project image into the Eigen space, we are using the %25 to say that 1st 25 are class 1 and 2nd 25 are class 2 etc…
  + If you have a different number of files in each folder of your training image, then we should change the numbering or coding there accordingly.
  + Since we are given 25 images in each folder, we used that 25. If you have suppose 10 test files in each folder please change 25 to 10 in the Eigen.h file
* Program execution time (tested on burrow.soic.indiana.edu):

I have run on the small set of images the training. And after successful working on the small data sample, we fed in all the images – 1250 – to the code. Its taking longer to calculate the eigenvectors. We don’t have the exact numbers. But our code ran perfect on the small train data. Please test on your side for accuracy of all data, as our model is still running and getting trained on the all images.

* Problems faced:  Since the image are color, we used the channel as 3 in the one of the matrix creations. But while multiplying with the Cimg since our image in the 3 channel mode and the PCA calculated matrix was in singular channel, it has created the problem. Which was considered to be handled. But later realized that we should keep everything in same channel. And CIMG multiplication depends on this.
* Assumption:
  + Existence of folders train/ and test/.
  + Enough space for the training file on disk.
* Working of code:
  + Program starts eigen\_taining()
  + Subsampled the images
  + Stacked all images by flattening each image into a vector f row
  + Normalized the each image with mean
  + Then calculated the covariance of the data
  + Then calculated the eigen vectors on covariance matrix
  + Then selected top k eigen vectors
  + Then project the original image to the new top k eigen space by multiplying them
  + Then used the resultant transformed image vector to the svm classifier
  + Then trained with that data
  + Later tested on that model

**Part 2.2- Haar-like features:**

* How to run the program (after executing the make command):
  + Training: ./a3 train haar
  + Test: ./a3 test haar
* Magic Parameters:
  + Subsample size: 30x30
  + Mask rectangles count: 300
  + Mask rectangle dimensions: Random (min: , max: )
* Program execution time (tested on burrow.soic.indiana.edu): 3hrs approx
* Problems faced: when we increase the subsample size, like 80x80, we get ‘out of memory’ exception due to large number of features being generated and limited amount of disk space on server/machine. So as we increase the subsample size, we need to decrease mask rectangle counts accordingly. To change the subsample size, change the value for variable ‘subsample\_size’ and to change the mask count, change the value for variable ‘maskCount’.
* Assumption:
  + Existence of folders train/ and test/.
  + Enough space for the training file on disk.
* Working of code:
  + Program starts with generate random masks with random rows and columns.
  + We subsample each image and pass each image to ‘performHaar’ function. In this function, we take each mask and move on top of the image to calculate the feature values. -1 denotes black and 1 denotes white. We calculate sum for each component visible and subtract from the other. This value constitute one feature. We repeat these steps for each image.
  + We write the features into haartrain.dat file and this is passed as input to SVM. SVM then trains or tests based on the mode.
* **Note: Program is sometimes giving segmentation fault after running SVM. SVM is giving that exception and that’s why couldn’t resolve it. If getting ‘out of memory’ exception, please reduce the Mask rectangle count. It is because of the large train file.**
* Result with rect count 100: Reading test examples... (250 examples) done.
  + Classifying test examples...done
  + Runtime (without IO) in cpu-seconds: 1.63
  + Average loss on test set: 90.0000
  + Zero/one-error on test set: 90.00% (25 correct, 225 incorrect, 250 total)

**Part 2.3- Bag of features:**

* How to run the program (after executing the make command):
  + Training: ./a3 train bow
  + Test: ./a3 test bow
* Magic Parameters:
  + K-mean: 100
  + Mask rectangles count: 500
  + Mask rectangle dimensions: Random (min: , max: )
* Program execution time (tested on burrow.soic.indiana.edu): Still running with 10,000 iterations
* Problems faced: Handling large values when calculating distance was the biggest issue. It was giving –NaN and was biased towards first cluster always.
* Assumption:
  + Existence of folders train/ and test/.
  + Enough space for the training file on disk.
* Working of code:
  + First, dump all images sift vectors to one vector and from those select k random sift descriptors as initial centroids.
  + Created a vector of structure with each structure object holding each sift vector with its corresponding cluster id.
  + Then we iterate for 10000 times clustering and re-centering clusters over new calculated centroids.
  + Once we get converged centroids which we use to plot histograms over these centroids.
  + We take each image, its sift descriptors and calculate distance of each sift vector to each of the centroids and increment the count for corresponding centroid with least distance.
  + This will give histograms over features.
  + Finally we do this for all image and feed the feature file to svm
  + Similarly, we calculate feature while testing and feed to svm
* Observations:
* Since, program was still running, we could not test on test data.

Q. 3)

Trained the given weight files generated model. On that cnn model we trained and tested.

**Part 3- Overfeat:**

* How to run the program (after executing the make command):
  + Training: ./a3 train bow
  + Test: ./a3 test bow
* Magic Parameters:
  + Layer 21 weights were used for this training.
* Program execution time (tested on burrow.soic.indiana.edu): took 1.5 hours for training
* Problems faced: Size of training file generated
* Assumption:
  + Existence of folders train/ and test/.
  + Enough space for the training file on disk.
* Working of code:
  + Run overfeat to extract all the features
  + Save all the features to feature file
  + Run svm over that file
  + Test on the model file generated
* Observations:
* 68 correctly classified from 250 images.

**Observations**: As overfeat requires images of size 231x231, some of the training/testing images were not subsampled and hence were giving segmentation fault during training, but it does not affect training /testing process with the other images. It shows segmentation fault on the console but still proceeds with the training/testing phase for the other images.

Q. 3)

Final epsilon on KKT-Conditions: 0.09278

Upper bound on duality gap: 0.08575

Dual objective value: dval=69.36502

Primal objective value: pval=69.45078

Total number of constraints in final working set: 402 (of 553)

Number of iterations: 554

Number of calls to 'find\_most\_violated\_constraint': 36250

Number of SV: 400

Norm of weight vector: |w|=6.63060

Value of slack variable (on working set): xi=47.41046

Value of slack variable (global): xi=47.46834

Norm of longest difference vector: ||Psi(x,y)-Psi(x,ybar)||=88.35872

Runtime in cpu-seconds: 2060.59

Final number of constraints in cache: 5669

Compacting linear model...done

Writing learned model...done

[rtawde@silo gmadhwan-rtawde-nsuri-a3]$ ./a3 test deep

a3 | main :: info : main function started

a3 | main :: info : files loaded

SVMBasedAlgo | overfeat-test :: info : inside function

sh: line 1: 105926 Done                    convert test/pizza/1718.jpg ppm:-

     105927 Segmentation fault      | /nfs/nfs7/home/rtawde/gmadhwan-rtawde-nsuri-a3/overfeat/bin/linux\_64/overfeatcmd /nfs/nfs7/home/rtawde/gmadhwan-rtawde-nsuri-a3/overfeat/data/default/net\_weight\_0 -1 0 21

sh: line 1: 113792 Done                    convert test/sushi/603.jpg ppm:-

     113793 Segmentation fault      | /nfs/nfs7/home/rtawde/gmadhwan-rtawde-nsuri-a3/overfeat/bin/linux\_64/overfeatcmd /nfs/nfs7/home/rtawde/gmadhwan-rtawde-nsuri-a3/overfeat/data/default/net\_weight\_0 -1 0 21

SVMBasedAlgo | write\_map\_to\_file :: info : inside function

Reading model...done.

Reading test examples... (250 examples) done.

Classifying test examples...done

Runtime (without IO) in cpu-seconds: 1.01

Average loss on test set: 72.8000

Zero/one-error on test set: 72.80% (68 correct, 182 incorrect, 250 total)

Final epsilon on KKT-Conditions: 0.09278

Upper bound on duality gap: 0.08575

Dual objective value: dval=69.36502

Primal objective value: pval=69.45078

Total number of constraints in final working set: 402 (of 553)

Number of iterations: 554

Number of calls to 'find\_most\_violated\_constraint': 36250

Number of SV: 400

Norm of weight vector: |w|=6.63060

Value of slack variable (on working set): xi=47.41046

Value of slack variable (global): xi=47.46834

Norm of longest difference vector: ||Psi(x,y)-Psi(x,ybar)||=88.35872

Runtime in cpu-seconds: 2247.00

Final number of constraints in cache: 5669

Compacting linear model...done

Writing learned model...done

[rtawde@silo gmadhwan-rtawde-nsuri-a3]$ ./a3 test deep

a3 | main :: info : main function started

a3 | main :: info : files loaded

SVMBasedAlgo | test :: baseline info : inside function

sh: line 1: 40156 Done                    convert test/pizza/1718.jpg ppm:-

     40157 Segmentation fault      | /nfs/nfs7/home/rtawde/gmadhwan-rtawde-nsuri-a3/overfeat/bin/linux\_64/overfeatcmd /nfs/nfs7/home/rtawde/gmadhwan-rtawde-nsuri-a3/overfeat/data/default/net\_weight\_0 -1 0 21

sh: line 1: 47213 Done                    convert test/sushi/603.jpg ppm:-

     47214 Segmentation fault      | /nfs/nfs7/home/rtawde/gmadhwan-rtawde-nsuri-a3/overfeat/bin/linux\_64/overfeatcmd /nfs/nfs7/home/rtawde/gmadhwan-rtawde-nsuri-a3/overfeat/data/default/net\_weight\_0 -1 0 21

SVMBasedAlgo | write\_map\_to\_file :: info : inside function

Reading model...done.

Reading test examples... (250 examples) done.

Classifying test examples...done

Runtime (without IO) in cpu-seconds: 0.97

Average loss on test set: 72.8000

Zero/one-error on test set: 72.80% (68 correct, 182 incorrect, 250 total)