AIN433 – Computer Vision Lab.

Programming Assignment 1

Burak Kurt

2200765010

# PART 1: Dimensionality Reduction with PCA:

PCA (Principal Component Analysis) is a dimensionality reduction technique that transforms a set of correlated features into a smaller set of uncorrelated features called principal components. PCA preserves as much of the variance in the original data as possible. Dimension reduction is important especially for images because high-dimensional data can be noisy, redundant and computationally expensive to process. By reducing the dimension of image data, we can remove irrelevant information, improve the efficiency and accuracy of learning algorithms, and visualize the data more easily.

In this assignment, PCA algorithm was implemented to reduce the dimensions of given images. To achieve that, first, images were converted into grayscale (only 1 channel) and flattened. 65536 feature was obtained for each image and since there were 11 image, data matrix were become 65536x11. Then obtained mean vector and subtract it from data matrix to calculate normalized data matrix (D). Next, covariance matrix is calculated with formula $D^TD$. Then, eigenvalues and eigenvectors are calculated from covariance matrix. First N (number of features we want after PCA) eigenvectors were selected corresponding to their sorted eigenvalues (S). Higher eigenvalues and their corresponding eigenvectors can present more of the data than the lower ones, so thats why highest N eigenvalue and eigenvector were chosen. For the next step, projection matrix was obtained from formula $DS^T$ . In the final step, each image was multiplied with projection matrix to obtain 1xN matrix, where 1 corresponds to 1 image, N corresponds to feature number. After concatenation of images, final result was 11xN matrix. For the part 1 of assignment, N was chosen as 3.

| | PC 1 | PC 2 | PC 3 |
|---|---|---|---|
| Aligned_Fighter01.bmp | 4646.820479 | -7124.494547 | 6909.510773 |
| Aligned_Fighter02.bmp | 3188.792547 | -7273.808788 | 5670.775232 |
| Aligned_Fighter03.bmp | 3413.039555 | -7260.533458 | 6082.272681 |
| Aligned_Fighter04.bmp | 4170.074847 | -7293.941298 | 6702.460664 |
| Aligned_Fighter05.bmp | 5744.879745 | -7498.931901 | 7556.356061 |
| Aligned_Fighter06.bmp | 5367.981405 | -8684.689673 | 8058.051254 |
| Aligned_Fighter07.bmp | 5136.980652 | -8301.256711 | 7333.720943 |
| Aligned_Fighter08.bmp | 5951.747332 | -8404.716809 | 7120.684672 |
| Aligned_Fighter09.bmp | 3001.531976 | -6945.471664 | 6974.377085 |
| Aligned_Fighter10.bmp | 4750.030160 | -8030.641108 | 7772.263157 |
| Aligned_Fighter11.bmp | 4717.298746 | -7704.194933 | 7548.539368 |

*Fig 1: New features after PCA algorithm*

Final result is shown in Fig1. Although image dimensions are reduced to 3 from 65536, new features contain very high number which could cause problems when they were given into model. To avoid this problem, scaling methods can be applied to these features. Scaled values are shown in Fig2.

| | PC 1 | PC 2 | PC 3 |
|---|---|---|---|
| Aligned_Fighter01.bmp | 0.557684 | 0.897067 | 0.518891 |
| Aligned_Fighter02.bmp | 0.063474 | 0.811216 | 0.000000 |
| Aligned_Fighter03.bmp | 0.139484 | 0.818849 | 0.172371 |
| Aligned_Fighter04.bmp | 0.396087 | 0.799640 | 0.432160 |
| Aligned_Fighter05.bmp | 0.929881 | 0.681776 | 0.789846 |
| Aligned_Fighter06.bmp | 0.802128 | 0.000000 | 1.000000 |
| Aligned_Fighter07.bmp | 0.723828 | 0.220463 | 0.696587 |
| Aligned_Fighter08.bmp | 1.000000 | 0.160976 | 0.607349 |
| Aligned_Fighter09.bmp | 0.000000 | 1.000000 | 0.546062 |
| Aligned_Fighter10.bmp | 0.592668 | 0.376059 | 0.880287 |
| Aligned_Fighter11.bmp | 0.581573 | 0.563756 | 0.786572 |

*Fig 2 Same features after scaling.*

# PART 2: Image Retrieval

A histogram is a graphical representation of the distribution of data, where each bin shows the frequency or count of observations in a given range. PCA (Principal Component Analysis) is a technique that reduces the dimensionality of data by finding a set of linearly uncorrelated variables, called principal components, that capture most of the variance in the data. For image data, PCA can be used to compress or visualize the images by projecting them onto a lower-dimensional space spanned by the principal components.

Image Retrieval algorithm has been implemented using 3 different features: PCA components, RGB and HSV histograms. First, images were represented with these features. Next, for each query image, distance to the other images were calculated and ten nearest image (lowest distance) to the each query image were shown. Finally, MAP score for each class was calculated. Starting from first rank, images were retrieved. If image was relevant (belongs to same class with query image), then precision value is calculated for that rank. Precision calculation ended when all relevant images were retrieved (30 relevant images per query image). Since there were 2 query image per class, mean of the MAP values were taken as a final MAP score for given class. Results were shown in notebook file.

For the ten nearest image algorithm, results were bad. Mean of the accuracies was nearly 0.2 where accuracy was defined by (Relevant Images) / (Retrieved Images). For each feature setting, results were similar.

MAP score of images where PCA components were used as features was the worst. HSV and RGB color spaces’ scores were similar however three of them had bad scores. Highest MAP score was 0.2834 on “Airplane” class with RGB histogram components.

# PART 3: Classification

Logistic regression is a classification algorithm which assigns weights for each feature and add a extra bias term and uses sigmoid function to obtain output of the model. To calculate output, features and their corresponding weights are multiplied first, then this multiplications are added along with the bias term. This calculation becomes the input of the sigmoid function where output is ranged between 0 and 1. With given threshold value, classifier assign that output into one of the classes. Optimization techniques are used to adjust weights of the model to increase accuracy of the model such as gradient descent.

In the binary classification part, first 2 classes were used for testing (airplane and bear). 30 airplane and 30 bear images with their labels were used for training the model. 2 airplane and 2 bear images were taken from QUERY_IMAGES folder as a test images. Test images were given to the model  after training. Model’s accuracy was 100% which means that it predicted every image’s class correctly. Result seems perfect but it was only tested with 4 images which is very low number to understand if the model’s generalization power is enough or not. With more test samples, more accurate analyses can be made.

To classify all the classes instead of 2, One versus All Logistic Regression is used. This model contains 10 (number of classes) binary logistic regression model in it. Each binary model is responsible to predict its own class. Each model was trained seperately. In the prediction, each model returns a value for given test data. Model with the highest prediction value choses the class of the test data.

Training and test data is same as the binary logistic regression except all the images and classes are used in this model. Testing was done with query images. For the 20 test images, model predicted 4 of them correctly and most of them selected “airplane” as a predicted class.  Each sample result and confusion matrix is in Fig3 and Fig4.

| Truth | Prediction | |
|---|---|---|
| 0 | 0 | Match |
| 0 | 0 | Match |
| 1 | 0 | |
| 1 | 0 | |
| 2 | 2 | Match |
| 2 | 0 | |
| 3 | 6 | |
| 3 | 8 | |
| 4 | 0 | |
| 4 | 0 | |
| 5 | 0 | |
| 5 | 0 | |
| 6 | 9 | |
| 6 | 2 | |
| 7 | 0 | |
| 7 | 9 | |
| 8 | 9 | |
| 8 | 6 | |
| 9 | 9 | Match |
| 9 | 0 | |

*Fig 3 Truth and prediction value for each image*

```
[2, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[2, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[1, 0, 1, 0, 0, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 0, 0, 1, 0, 1, 0],
[2, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[2, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[0, 0, 1, 0, 0, 0, 0, 0, 0, 1],
[1, 0, 0, 0, 0, 0, 0, 0, 0, 1],
[0, 0, 0, 0, 0, 0, 1, 0, 0, 1],
[1, 0, 0, 0, 0, 0, 0, 0, 0, 1]
```

*Fig 4 Confussion matrix for Logistic Regression model*