Image Recognition and Classification using Advanced Algorithms for Classification and Clustering

Bilalpur Maneesh(201532589)
Rhishi Pratap Singh (201532546)
Kranthi Kumar Rachavarapu (201532563)

CSE 471: Statistical Methods in Artificial Intelligence

April 21, 2016

Abstract

The objective of this project is to develop an offline - image recognition and classification system using the techniques learned in the course and study the performance of the algorithms. In the first part, The Bag of Words (BoW's) features are extracted from training images and a histogram is generated for the test images using these BoW's. This histogram is used as a feature representation of image for classification. Methods like K-Nearest Neighbours (KNN), Perceptron and Support Vector Machine (SVM) are used for image classification. We observed that SVM outperforms the other 2 algorithms interms of classification accuracy. The second part This is followed by face recognition using eigen faces.

These techniques are applied on CalTech-101 and Extended Yale Face Database B database. A detailed analysis of the results is presented at the end.

1 Introduction

Humans are able to analyse and classify objects of there surroundings accurately - mystery. Implementing this feature in computer can be useful in many tasks such as robot navigation, security, surveillance etc. But this tasks is very difficult because of the following reasons.

- There are about 10,000 to 30,000 different object categories.
- Viewpoint variation where many objects can look different from different angles
- llumination in which lighting makes the same objects look like different objects
- Background clutter in which the classifier cannot distinguish the object from its background
- Scale, deformation, occlusion, and intra-class variation

Bag of Visual Words (BoVW) is a popular technique in Computer Vision for image classification inspired from natural language processing. BoVW downplays word arrangement (spatial information in the image) and classifies based on a histogram of the frequency of visual words. The set of visual words forms a visual vocabulary, which is constructed by clustering a large corpus of features. To simply put, it uses histograms of images for classification.

We have compared SVM, Perceptron, KNN classification methods on CALTECH-4 database. Section 2 gives a brief outline of BoVW, Classification methods and eigen faces. In Section 4, we present the results and give an analysis of the results. Finally, Conclusions are presented in section 5.

2 Method

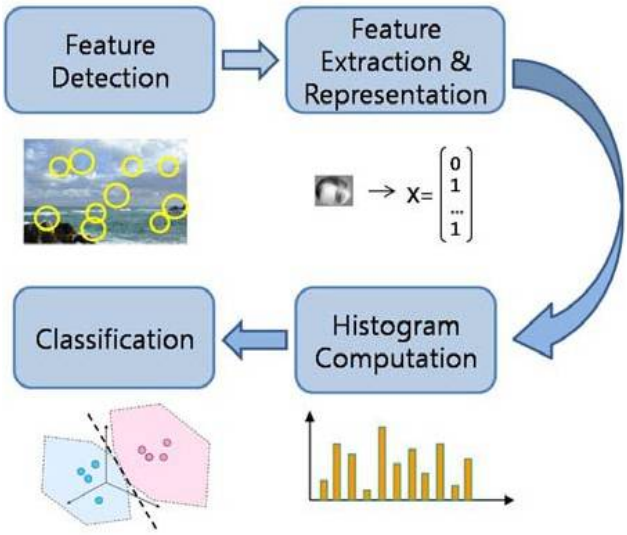

Figure 1: Illustration of Image Classification using BoVW

2.1 Visual BoW Features

One of the most frequently used algorithms for category recognition is the bag of Visual words (abbreviation BoVW) [1]. This algorithm generates a histogram, which is the distribution of visual words found in the test image, and then image is classified based on classifier's characteristics. The purpose of the BoW model is representation. Here, representation deals with feature detection and image representation. Features must be extracted from images in order to represent the images as histograms. In the following sub-sections, we describe the individual steps of Classification process.

2.1.1 Scale Invariant Feature Transform[2]

The first step for our two classification methods is to extract interest points and describe them in an image. For any object in an image, there are certain characteristics that can be extracted and define what the image is. For any object in an image, there are some characteristics that define it and these objects inturn define the image. Features are then detected and each image is represented in different patches. In order to represent these patches as numerical vectors we used SIFT descriptors to convert each patch into a 128-dimensional vector. SIFT features are invariant to view point , illumination, scale, rotation. Hence, they give a robust estimation of similar patches in 2 images.

2.1.2 Building Visual Words Dictionary

After extracting features from both testing and training images, we need to convert the vector represented patches into codewords. And this is performed by K-means clustering. Once we obtain k clusters from training image features, each new feature belongs to the cluster with nearest mean. The number of clusters will have an impact on the classification accuracy. A smaller k will lead to a poor classification accuracy as the visual dictionary is smaller.

2.1.3 Image Representation by Visual words

Each patch in an image is mapped to a certain codeword (a cluster center) through the k-means clustering process and thus, each image can be represented by a histogram of the codewords. This procedure converts the SIFT features in to histogram representation and is used for classification in the later stages.

2.2 Classification

2.2.1 K-Nearest Neighbour

The K-Nearest Neighbour follows a lazy learning approach to classification and uses the mode of label of K-nearest neighbours to label the data. The nearness is measured from euclidean distance while there are other metrics like Manhattan,Minkowski and Hamming (an Information theory based measurement) distance. The effect of K value sometimes causes the sample to be misclassified because of large number of outliers and hence it is not a robust classification technique and it does not involve model construction.

2.2.2 Support Vector Machines

The traditional perceptron model though gives a solution for linearly separable data, the solution does not ensure the maximum possible margin and it is just one of the several possible solutions from the solution space. For the cause of increasing generality, a decision boundary that maximizes the margin and improves accuracy is preferred. Support Vector Machine is an approach which achieves the same.

The decision boundary in SVM is dependant on the Support Vectors, the data points that "support" the margin hyperplanes. The test error in SVM is bounded as the sum of training

error and function of VC-dimension, minimum of both absolute margin and dimensionality of the problem. A minimum test error is possible if we can achieve a zero training error and maximum absolute margin.

$$P\left(TestError <= TrainingError + \sqrt{\frac{h(log(2N/h)+1)-log(\eta/4)}{N}}\right) = 1-\eta$$

The margin depends on the norm of the decision surface which leads to an optimization problem of maximising the margin and minimising the error. The problem is solved using QP-solver for legrange multipliers and support vectors. SVM is robust to non linearities in the data due to noise using Soft margin SVM (adopting a slack variable to control the amount of error the model can tolerate). Unlike perceptron models which need complete Kernel knowledge to separate non linear data, the SVM adopts a kernel trick which can solve the problem using Kernel trick and does not require complete knowledge. These characteristics of SVM make it a robust and a good classification model.

2.2.3 Multi Layer Feed Forward Neural Networks

MLFFNN is a perceptron model with hidden nodes and error back propagation. Unlike the perceptron model this needs a differentiable activation function to facilitate the back propagation of error. The learning law used is differentiable Delta learning law. The hidden layer presents both constructive and destructive features of the network, it is responsible for both non linear data classification and also over-fitting problem which can be overcome by limiting the number of hidden nodes based on the dimensionality of problem.

The input to hidden layer update rule is given by

$$\triangle w_{ji} = \eta\delta_j x_i = \eta\left[\sum_{k=1}^{c} w_{kj}\delta_k\right] f'(net_j)x_i$$

while the hidden to output rule is

$$\triangle w_{kj} = \eta\delta_k y_j = \eta\left[t_k - z_k\right] f'(net_k)y_j$$

Techniques like momentum and regularization provide support to the network so that it does not settle at a local minima.

2.3 Recognition using Eigen face

Eigenfaces is a set of eigen vectors used in the computer vision problem of human face recognition. These eigen vectors are derived from the covariance matrix of the image data [5]. The training and test images are projected over each eigen vector to calculate the dependency of the image with respect to the particular eigen vector. This reduces the image as a linear combination of the eigen vectors. And the euclidean distance between the coefficients of the eigen vectors is used to recognise the face. Lesser the euclidean distance more similar are the faces. Hence an image with least euclidean distance is the recognized face.

3 Results

We have used MATLAB as the programming language to implement Image Classification and Python to implement Face Recognition. The method is evaluated on CALTECH-4 Database [4]. 4 categories - Bikes, Airplanes, Faces and Background are used for classification. Each category has 400 training images and 35 testing images. Fig 2. shows the samples images of these categories.

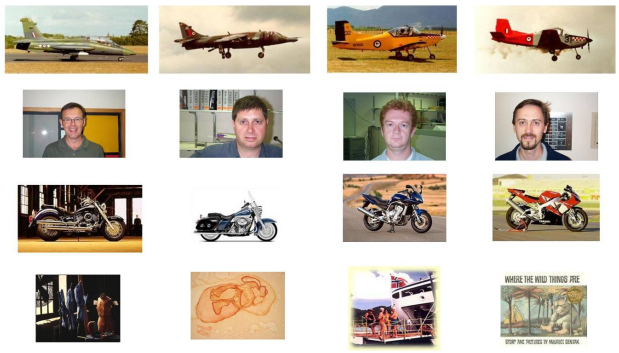

Figure 2: Sample Images of 4 Classes

The implementations of the SIFT features, k-means clustering and SVM are available in VLFEAT library [3], which is an open source library for popular computer vision algorithms. Neural Net Pattern Recognition is a matlab toolbox to implement a 2-hidden layer MLFFNN. We have used that in our implementation.

- Trainig Data : 1600 Images
- Testing Data : 140 Images
- Validation type: 5-fold Cross Validation
- Cluster size: 50, 150, 250 500, 1000, 5000

The results are presented as confusion matrix (percentage of correctly classified vs misclassified).

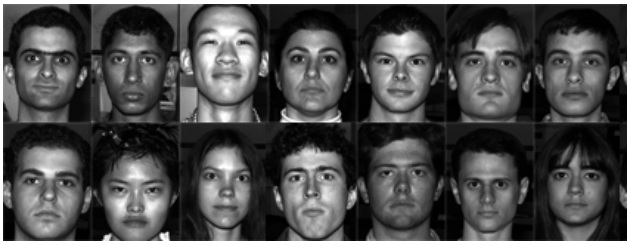

Figure 3: Sample images from Extended Yale face dataset B

The Extended Yale face dataset B [6] has faces of 38 individuals under different illumination conditions and poses. The head regions are cropped from the Original Yale dataset to form Extended Yale dataset B.

- Trainig Data : 1940 Images
- Testing Data : 484 Images
- Kinds of faces: 38
- Validation type: 5-fold Cross Validation

3.1 KNN Classification results

K-NN algorithm performs better if the cluster size is smaller. If the cluster size is increases, the performance of the algorithm decreases drastically as it fails to distinguish between classes. Table 1 shows the classification accuracy using KNN for different cluster sizes. Table 2 shows the confusion matrix corresponding to the highest accuracy i.e., for k = 50.

Cluster Size	**K-NN with K = 11**
50	**91.43%**
150	91.43%
250	82.14%
500	68.57%
1000	52.14%
5000	36.43%
10000	38.57%

	Bikes	Airplanes	Face	Background
Bikes	91%	0%	0%	9%
Airplane	3%	91%	0%	6%
Face	0%	0	100%	0%
Background	14%	3%	0%	83%

3.2 SVM Classication Results

SVM algorithm performs better if the cluster size is larger. As the cluster size increases, the performance of the algorithm improves. The best value of the cluster size is 10000 which is giving 95.3% accuracy. Table 3 shows the classification accuracy using SVM for different cluster sizes. Table 4 shows the confusion matrix corresponding to the highest accuracy. SVM algorithm performs better when the cluster size is larger. It is evident from the results that SVM outperforms K-NN in image classification.

Cluster Size	**SVM - One vs All**
50	80.71%
150	87.14%
250	88.57%
500	91.43%
1000	92.86%
5000	92.86%
10000	**95.71%**

	Bikes	Airplanes	Face	Background
Bikes	97%	0%	0%	3%
Airplane	0%	100%	0%	0%
Face	0%	0	97%	3%
Background	3%	9%	0%	89%

3.3 MLFFNN Classification Results

The architecutre of MLFFNN is (NoofClusters)-20-10-4. The network performs better for the cluster size 250. But the accuracy drops on increasing or decreasing the cluster size.

Table 5 shows the classification accuracy using SVM for different cluster sizes. Table 6 shows the confusion matrix corresponding to the highest accuracy.

The classification accuracy for MLFFNN is poor when compared with K-NN and SVM. And SVM performs better than the remaining algorithms.

Cluster Size	**MLFFNN**
50	86.43%
150	89.3%
250	**90%**
500	88.6%
1000	88.6%
5000	79.3%
10000	40.7%

	Bikes	Airplanes	Face	Background
Bikes	91.42%	0%	2.85%	5.71%
Airplane	0%	85.71%	0%	14.28%
Face	0%	2.85	94.28%	2.85%
Background	5.71%	5.71%	0%	88.57%

3.4 Eigen Face Recognition Result

The Extended Yale Face recognition B dataset has faces with four different poses and illumination conditions of each individuals' face in the test set. Using the eigen face approach to face recognition, we achieved an accuracy of 93.18%.

No. of Eigen Faces	Accuracy
10	47.1%
38	85.12%
190	91.52%
200	**93.18%**
380	91.11%

Ideally, as the dataset contains 38 unique faces, 38 eigen faces should give a maximum accuracy but it contains images with varying intensity and poses and hence heuristically, as we increase the eigen faces the accuracy increases.

4 Conclusions

We have implemented 3 methods for image classification using BoVW approach. Although the classification results on the features, it is observed that SVM outperforms K-NN and MLFFNN in terms of classification accuracy by 6%. This may be attributed to the fact that SVM builds model to classify the features where as KNN doesn't builds any model specific to classes and MLFFNN might be over-fitting the data.

The face recognition algorithm highlighting its simplicity has achieved a good accuracy. No knowledge of geometry and feature is required and raw data can be used directly in the process. It is very simple and efficient approach towards face recognition. However it cannot handle varying pose and illumination robustly. Recognition is efficient only when the number of face classes is larger than the dimensions of the face space.

References

[1] KIM JI, Kim BS, Savarese S. *Comparing image classification methods: K-nearest-neighbor and support-vector-machines*. Ann Arbor. 2012 Jan 25;1001:48109-2122.

[2] Lowe DG. *Distinctive image features from scale-invariant keypoints*. International journal of computer vision. 2004 Nov 1;60(2):91-110.

[3] Vedaldi A, Fulkerson B. *VLFeat: An open and portable library of computer vision algorithms.* InProceedings of the 18th ACM international conference on Multimedia 2010 Oct 25 (pp. 1469-1472). ACM.

[4] Griffin G, Holub A, Perona P. *Caltech-256 object category dataset.*

[5] Matthew A. Turk and Alex P. Pentland *Recognition Using Eigenfaces*. MIT Vision and Modeling Lab, CVPR 91.

[6] Lee KC, Ho J, Kriegman DJ.*Acquiring linear subspaces for face recognition under variable lighting*. Pattern Analysis and Machine Intelligence, IEEE Transactions on. 2005 May;27(5):684-98.