# The PLSA Bag-of-Words Classifier for Image Classification

Michael A. Chrisco
*California State University of Fresno Computer Science Undergraduate*
*michaelachrisco@gmail.com*

## ABSTRACT

*Creating a precise information retrieval method is a widely sought after algorithm. The pLSA Bag-of-Words Classifier for Image Classification is one such algorithm that has been implemented by Rob Fergus[4]. In this paper, I discuss the Latent Semantic Analysis as well as the probabilistic Latent Semantic Analysis[1,2]. I will discuss the concepts of the bag-or-words model and how this relates to the points of interest operators[3]. I will describe in detail the overall code base used for analysis[4]. Investigation on the ability of the overall capacity of recognition with faces and shoes will be discussed. There is also studies on the relation between the training set volume and the pLSA performance. Two case studies show that the implementation works best if the training set is larger and more diverse. Inversely it is also shown with two other case studies, the smaller the training set, the lower the precision and recall.*

## 1. INTRODUCTION

Image recognition and categorization is a field that is emerging as an important technology. Computers have a huge variety in ways to produce images but lack knowledge about the image itself. With this knowledge, however, a wide variety of applications can be applied. One such application is Augmentative Reality. With knowledge of what the computer is looking at, it can tell the person if there is anything that requires the users attention. A smart car could tell, for example, if it is going to hit another object just by images produced[4]. Another example of use is in medical imaging. Informational imaging could also be a boon to medical professionals[7]. X-rays and other medical imaging could give more accurate results and information that could be used to save lives. A more general example is using a image retrieval model[3]. An online service such as google images can categorize images given the knowledge of what they contain and therefore create a better model for retrieving images the user is interested in.

This paper is a case study of one such solution to the informational images retrieval problem. By grouping images by categories or topics, information pertaining to the images can be produced for later retrieval. Work by Rob Fergus shows that it is possible to create such a model[4]. His implementation on the bag-of-words classifiers using the pLSA algorithm shows an improvement to previous models such as the Latent Semantic analysis model[1]. Using this method, this paper will try to prove under what conditions is this implementation accurate and how such an algorithm can do the classification. Possible improvements will also be discussed to create a faster, more accurate image retrieval system.

## 2. METHODS AND IMPLAMENTATION

### 2.1 Bag-of-words

A study on Rob Fergus's Matlab implementation of the Bag-of-Words classifier was used in this paper[4]. This code is based in Matlab and binaries consisting of helper functions for prepossessing the images. The basis of this code is the Probabilistic Latent Semantic Analysis(pLSA).

Latent Semantic Analysis is a process used in document retrieval. Given a document and given the context of the words associative in the document, the LSA will attempt to create an underline meaning to those words[2]. These meanings are categorized into topics. Latent Semantic Analysis works by creating a Bag-of-Words from the documents it is analyzing. The Bag-of-Words algorithm is to take all the words in the document and create an unordered set of these words with the occurrences of these words. Representation of meanings or Concepts is observed as patterns of words usually appear in the documents associative to a topic. Words in the LSA is assumed to have only one meaning. The problem that lies with the LSA algorithm is one of multiple meanings of words. If each word has only one meaning to the LSA, then documents with words, given the context of the document, have different meanings, the results for topic classification will be skewed and incorrect. A solution that has a better performance that solves the multiple meaning problem is the pLSA[1].

Probabilistic Latent Semantic Analysis is an algorithm that uses the Latent Semantic Analysis as a basis but uses the concept of probability to create a more accurate topic classification[1]. The algorithm assumes a collection of documents D={d1,d2,..dn}. In each document, there are a collection of words W={w1,w2,...wn). As described in the LSA a bag of words is created with a co-occurrence table by documents and words with occurrences. Using this bag-of-words, we use probability of a latent variable z (or topic). Using probability makes it possible that the document can have multiple topics given different contexts. One can now create an observable pair <di,wj> by selecting a document di with the probability P(di).Then we pick a latent topic zk with the probability of P(zk|di). Now generation of an observable word wj with the probability of P(wj|zk) is possible. This algorithm creates the probabilistic model in Figure 1.

$$P(d_i,w_j)=\sum^{K} P(d_i)P(z_k|d_i)P(w_j|z_k)$$

$$P(z_k|d_i,w_j)=\frac{P(w_j,z_k|d_i)}{P(w_j|d_i)}=\frac{P(w_j|z_k,d_i)P(z_k|d_i)}{P(w_j|d_i)}=\frac{P(w_j|z_k)P(z_k|d_i)}{\sum^{K} P(w_j|z_l)P(z_l|d_i)}$$

$$P(w_j|z_k)=\frac{\sum_{i=1}^{N} n(d_i,w_j)P(z_k|d_i,w_j)}{\sum_{m=1}^{M}\sum_{i=1}^{N} n(d_i,w_m)P(z_k|d_i,w_m)} \quad , \quad P(z_k|d_i)=\frac{\sum_{j=1}^{M} n(d_i,w_j)P(z_k|d_i,w_j)}{n(d_i)}$$

Figure 1: Equation for pLSA model[1] includes the E and M step of the Expectation-Maximization Algorithm

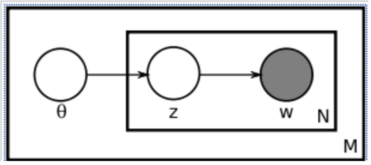

Figure 2: pLSA Model[1]

Table 1: Term-document matrix

| | w1 | ... | wi | ... | wk |
|---|---|---|---|---|---|
| d1 | | | | | |
| ... | | | | | |
| dj | | | n(dj,wi) | | |
| ... | | | | | |
| dv | | | | | |

Creating a term-document matrix given a training set of documents with words is also necessary to use the pLSA. If n(dj,wi) is the number of times a word wi occurs in a document di, and the document vector is defined as (n(di,w1),...,n(di,wk)), then we can use this to create the estimated mass probability distribution P(wi|dj). The total number of documents is denoted M and the total number of words(a vocabulary) is denoted by N. Thus the Term-document matrix can be seen as a M x N matrix that contains n(dj,wi).

The algorithm now needs to create the topic probability distribution (the latent topic). It does this by using the Expectation-Maximization Algorithm. This algorithm works by creating an E-step which calculates expectations for latent variables given observations by using the current estimates of parameters[5]. The M-step is used to update parameters pertaining to the data likely hood.

In the Matlab code used in this study, a visual bag-of-words is created[4]. First the algorithm randomly picks images from the training set. Next, it does a resize if needed to make sure all images have the same height and width. This is important if we wish to have accurate results that are not based on just the image size. An important step is to now section off certain areas of the image. This will be he visual words that will be used later in the pLSA. It does this by looking at points of interest using a Canny Edge detector. This specific algorithm was suppled by Yu Cao in Matlab code that replaced the original binary provided by the model by Rob Fergus[4,6]. Another algorithm called the SIFT creates categorizations based on the given points of interest. This creates the visual words given the image (or document). Now the pLSA is used on the images to create the topics. In this implementation, there are two sets of images, one with the training set and the other as a set of random images, comprised of images that do not belong to the set at all (such as desks and books) and contain the original training set of images. The goal of which is for the pLSA to recognize the high probability of the training set compared to the random images that have very little in common with the training set[1]. The final step is to visualize this data to see the correctness of the overall algorithm.

## 2.2 Case study #1: Face or No Face

The first study that was conducted regarded recreating the original study that was done by Rob Fergus. In this study, a collection of 100 images were taken of people's heads and then another set of 100 images was created with background images. In this case, the head images were used as the training set and the background images were used as the random set. The goal of this case study was to categorize the images of heads as heads by the algorithm and everything else as not heads. Another goal was to see what the pLSA chose as visual words and if those choices were associative with faces (such as mouth, nose, and ears).

## 2.3 Case Study #2: Recognizing other objects in smaller datasets

The second case study was conducted to see if smaller sets of training data would produce an accurate set of results. In this case, 5 images of shoes were used as the training set and 5 pictures found randomly online were used. Again the goal was to see what the pLSA chose as visual words and to see how correct the results were given the dataset size.

## 2.4 Case study #3: Smaller dataset for Face or No Face

After the first two Case studies results were found a third case study was conducted on the accuracy of the implementation given a specific dataset that has had results before. Half of the face images were used from case study #1. The goal of this study is to find out if there is a direct correlation between the volume of the training data and the accuracy of the results.

### 2.5 Case study #4: Other Objects with large training set

If results showed that there was a direct correlation to the volume of the training data to the accuracy of the image retrieval, then, intuitively, one could postulate that the inverse could be stated as well. Case study #4 was used to verify if there is a implication that bigger training sets with a more diverse image set can produce better image retrieval. A training set of 50 shoes was used that included the case study #2 training set.

## 3. CASE STUDY RESULTS

The case study of Face and No face showed results from the interesting points that the outline of the head and other outlines pertaining to the head (nose mouth, eyes, neck) were also present. There was some noise in the algorithm that was in the form of background components such as fixtures and walls as can be seen in figure 3.

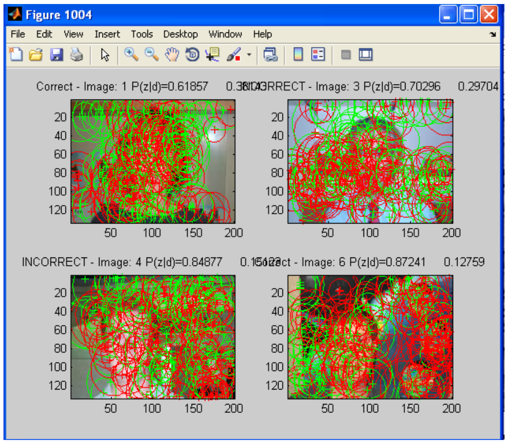
Figure 1004
File Edit View Insert Tools Desktop Window Help
Correct - Image: 1 P(z|d)=0.61857 0.38143
INCORRECT - Image: 3 P(z|d)=0.70296 0.29704
INCORRECT - Image: 4 P(z|d)=0.84877 0.15123
Correct - Image: 6 P(z|d)=0.87241 0.12759
20
40
60
80
100
120
50
100
150
200

Figure 3: Individual results for Face/No Face. P(d|z) is shown

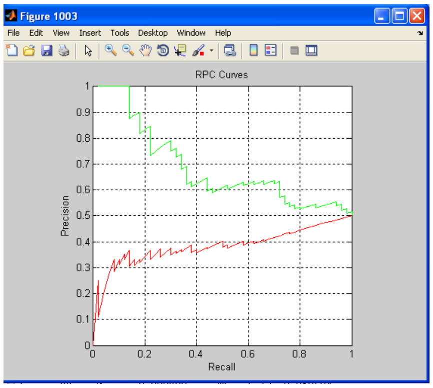

Figure 4: Face/ No Face Line Graph of Results in Precision and Recall

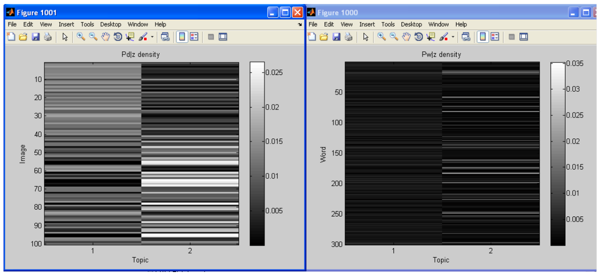

Figure 5: Face/ No Face P(d,z) density graph on the left and the P(w|z) graph on the right.

The case study number 2 of the smaller set of shoe images produced a very inaccurate result. The shoes were outlined by the canny edge algorithm and visual words were made from the shoe. The implementation did not recognize the shoe image outlines. This created a very low probability of the image being a shoe even if it was an image of a shoe. Interestingly, the shoes visual words like shoelace, toe, and back of foot can easily be seen in figure 6, even though the low probability was associative to the outline. Figure 6 has a P(d|z) value of .09 were it should be higher because it was part of the original training set. The overall precision and recall is significantly lower than the Face and No Face study. Most of the images were categorized into the no shoe category seen in Figure 8.

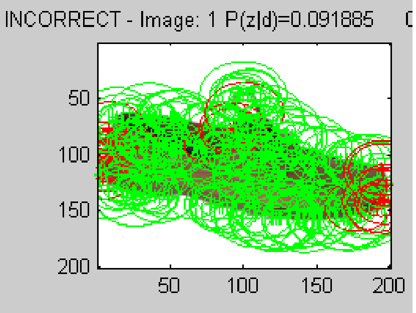

Figure 6: Shoe/No Shoe sample of P(d|z)

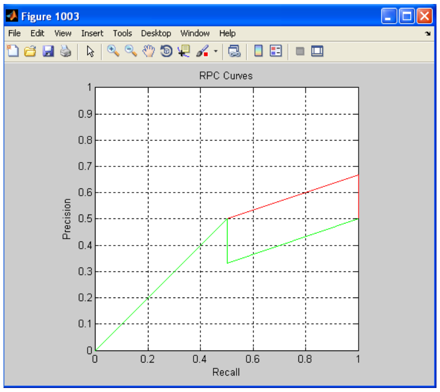

Figure 7: Graph of Results in Precision and Recall

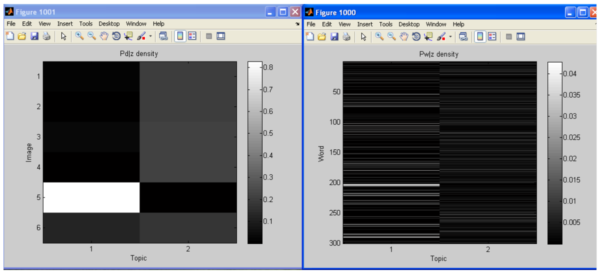

Figure 8: Shoe/No Shoe P(d,z) density graph on the left graph and the P(w|z) graph on the right.

Case study number 3 in the Face or No Face study showed a correlation between smaller datasets and the pLSA ability to categorize. My hypothesis seeing the performance of Case study #2, was smaller datasets produce results that are suboptimal. In this case study, figure 11, image 24 shows a P(d|z) of only .41 which, because the image was part of the training set, should be much higher as seen in figure 3. This is reflected in the other samples of P(d|z).

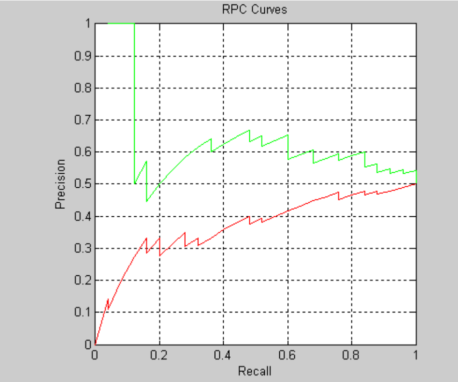

Figure 9: Case study #3 smaller set of training images Recall and Precision

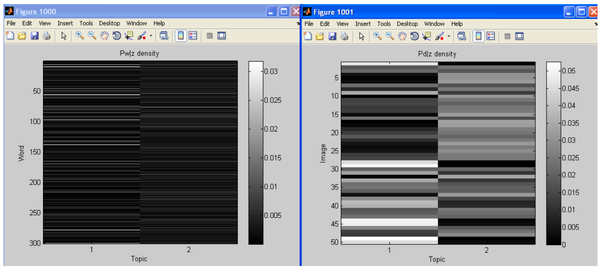

Figure 10: Case study #3 smaller set of training images P(d,z) density graph on the left and the P(w|z) graph on the right.

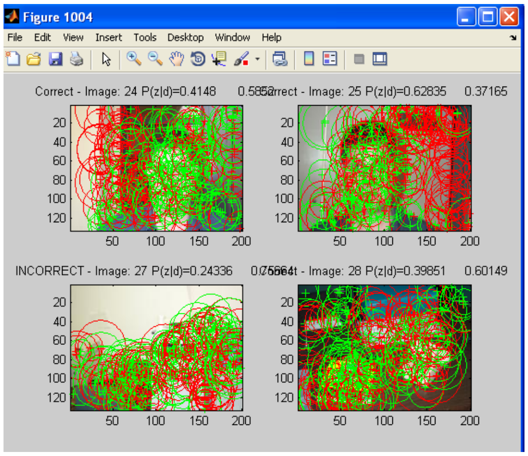
Figure 1004
File Edit View Insert Tools Desktop Window Help
Correct - Image: 24 P(z|d)=0.4148 0.5852
Correct - Image: 25 P(z|d)=0.62835 0.37165
INCORRECT - Image: 27 P(z|d)=0.24336 0.75664
Correct - Image: 28 P(z|d)=0.39851 0.60149
20
40
60
80
100
120
50
100
150
200

Figure 11: Face or No Face sample of P(d|z)

Another hypothesis that was implied by case study #3 was that a bigger dataset of an object can lead to a better categorization of said object. In case study #4, 50 images of shoes were used as the training set. One can see a difference in the number of visual words gathered from figure 12 comparable to figure 6. Whereas figure 6 had a P(d|z) of .09, figure 12 showed a P(d|z) of .51, a much higher percentage. Both of these figures are identical pictures except one has a bigger dataset to pull visual words from and the other has a very limited dataset.

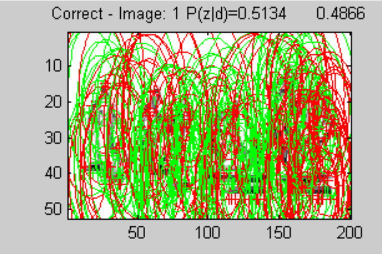

Figure #12: Same picture as Case study #2 but with bigger training set

## 4. CONCLUSIONS

These four case studies showed that objects other than faces can be identified. It also showed a correlation between the actual training data and the overall performance of the algorithm. There seemed to be a large correlation between the volume of the training set and its accuracy. This correlation seems to be one of an inverse relationship. Case study #4 showed a possible correlativity that, if the dataset is sizable, then the algorithm can categorize more accurately. The inverse seemed to be recognized by Case study #3. A smaller dataset was used and less accurate results were found. This leads to an almost intuitive result: that the bigger and more diverse the training set is, the better the algorithm can produce accurate results to the user.

## 5. FUTURE WORK

One of the side effects of the current implementation of the pLDA in image retrieval is that it is very slow to the point were modern computers (as of date 2010) takes as much as 2-3 hours to run prepossessing stages as well as the pLSA implementation. The actual pLSA is one of the slowest components of the system. A faster pLSA algorithm might be possible with a visual bag-bag-words model already put into a tree like structure[7,8]. This could possibly create less overall lookup time. Also, in the real world, there are specific rules that we already know about images such as faces contain a nose, mouth, and eyes. If categorization methods were already in place, a categorization graph could be used in combination with the pLSA to produce a more accurate result[7]. If one knows that eyes are located inside of a face and the pLSA does not find such a visual word, the categorization method might not retrieve that result or give it a low value in the overall image retrieval.

One proposed implementation that could improve the current implementation of the pLSA for image retrieval is the Multilayer pLSA for multimodal image retrieval(mm-pLSA)[8]. This algorithm would call for a two step pLSA in two layers. The first layer of the pLSA on images is to create the modal using a bag-of-words like the one described in this paper. The second pLSA would be created using tags that a photographer would supply. This would create an another pLSA that might be more accurate than the original pLSA in the visual words model. These two pLSA algorithms can be used to create a co-occurrence table comprised of the merged co-occurrence tables generated by the first two pLSA's. This table can be used to create another pLSA and produce another level of complexity. It should be noted the that original authors stated that the results of this mm-pLSA was sub-optimal because of the initialization training sets. This reflects this current papers conclusion about the The PLSA Bag-of-Words Classifier for Image Classification.

## 6. References

[1] T. Hofmann. *Probabilistic Latent Semantic Analysis,* pages 1-8. 1999

[2] T. Landauer, P. D. Foltz, and D. Laham. *Introduction to Latent Semantic Analysis.* Discourse Process, pages 3-18. 1998

[3] L. Fei-Fei, R. Fergus, and A. Torralba. Recognizing and Learning Object Categories Slides CVPR 2009. Part 1 and Part 2. 2009

[4] R. Fergus. *Two bag-of-words classifiers.* ICCV 2005 short courses on Recognizing and Learning Object Categories. 2005

[5] R. Lienhart. *Probabilistic Latent Semantic Analysis*, SS 2008 - Bayesian Networks. Pages 3-9. 2008

[6] Y. Cao. *Scale Invariant Feature Transform (SIFT) developed by David G. Lowe.* 2006

[7] Y. Cao. *IMIS - Intelligent Medical Image Retrieval System,* CSCI 154-Simulation. pages 9-11. 2010

[8] R. Lienhart, S. Romberg, and E. Horster. *Multilayer pLSA for Multimodal Image Retrieval.* pages 3-9. 2009

[9] K. Bengler, R. Passato. Augmented Reality in Cars Requirements and Constraints, BMW Group, pages 4-15. 2004