# 08: Pyramid Histogram of Words (PHOW)

Sergio Steven Leal Cuellar* and Mateo Rueda Molano†,
Department of Biomedical Engineering, University of the Andes, Bogota, Colombia
Email: *ss.leal10@uniandes.edu.co, †ms.rueda10@uniandes.edu.co,

***Abstract*—The PHOW algorithm is based on extracting visual words to get spatial information of the image descriptor using pyramid matching. Using the previous mentioned method two SVM model were trained in two object recognition datasets; Caltech 101 and ImageNet200. Some experiments were ran in order to identify the optimal parameters (e.g Train samples, size, step, number of words, SVM confidence, etc.) on Caltech101 and ImageNet200. The results obtained for the Caltech101 dataset were around an ACA of 0.7068 and for the ImageNet200 dataset the mean ACA obtained was 0.2458. In conclussion, the PHOW algorithm presents some limitations due to the meaningless information that often is taking from non-interest zones like backgrounds, noise or details which difficults the SVM model to properly classify the data samples. As a proposal, this problem could be diminished by using some previous operations and pixel selection on the images.**

## I. Introduction

The development in multimedia technology and internet has risen the amount of visual information available. Therefore, it is more challenging for worldwide web to locate accurate and efficient visual information from big data of images. The image processing has partially solved this issue by introducing numerous image classification algorithms such as BOW which is recognized in the scientific community [1]. BOW algorithm [1] was originally introduced in document mining. At present, the concept of BOW paradigm is shifted from text mining to image processing through SIFT feature extraction algorithm [2]. In this algorithm, in the bag-of-words method to get the image features, the SIFT algorithm is used to extract the feature points of images and the obtained SIFT features are used to generate a fixed number of the typical local features as visual words, which did not take full advantage of the spatial distribution of image information and the global information. It is exactly based on this consideration that many researchers propose a patch-based local and global bag-of-words features with spatial pyramid matching to get spatial information of the image descriptor (Pyramid Histogram of Words : PHOW) [3]. Researchers also improve the BOW model by using dense sample based on patches of the original sample and an improved k-means clustering method to construct the visual dictionary to overcome the problem of different results due to the initial seed [4]. After the use of kmeans, the descriptors of the images are constructed using the spatial histograms generated from the visual dictionary. Finally, these dictionaries are separated with support vector machine (SVM) multi-class classifiers by using a kernel map to transform the (SVM) into a linear one [3].

Nowadays, the use of SIFT algorithm is more common for achieving results in the medical field [2]. In recent years, some extended versions of SIFT have been developed such as PCA-SIFT, GLOH and SURF. They provide dimension reduction and much other functionality. Although, traditional SIFT algorithm is helpful in image recognition there exist some issues in image classification such as running time and accuracy is not high enough. Additionally, traditional SIFT algorithm is not efficient to extract the features from noisy image.

In this order of ideas, in this laboratory the BOW-SIFT method presented in [4] will be used for the classification of images from the Caltech 101 database. This algorithm reached an accuracy of 65% using 15 images per class for the training group. After evaluating the best parameters for this model, the classification result will be evaluated in the 101 categories and the results obtained will be compared with the developers of the method. Subsequently, this classification method will be evaluated in a larger database, imageNet200, composed of 200 categories with 100 images for train and test.

## II. materials and methods

### A. Dataset description

Caltech-101 is a dataset of picture objects belonging to 101 categories. There are bout 40 to 800 images per category and categories have about 50 images [5]. There are also the outlines of each object in these pictures included under the 'Annotations.tar'. Caltech-101 is used for classification tasks and the developers of the dataset suggest training and testing on fixed number of pictures and repeating the experiment with different random selections of pictures in order to obtain error bars. Popular number of training images: 1, 3, 5, 10, 15, 20, 30. Popular numbers of testing images: 20, 30 [5].

ImageNet is a big image dataset organized according to the WordNet hierarchy. Each meaningful concept in WordNet, possibly described by multiple words or word phrases, is called a "synonym set" or "synset". There are on average 1000 images to illustrate each synset. Images of each concept are quality-controlled and human-annotated. In this article, we use a comprised version of ImageNet (imageNet200) with 200 categories and 100 images per categoty in train and test [6]

### B. Methods description

As said before, PHOW uses the bag-of-words method to get the image features, patch-based local and global bag-of-words features with spatial pyramid matching to get spatial

information of the image descriptor (Pyramid Histogram of Words : PHOW). K-means is then applied clustering method to construct the visual dictionary to overcome the problem of different results due to the initial seed [VLFEAT]. After the use of kmeans, the descriptors of the images are constructed using the spatial histograms generated from the visual dictionary. Finally, these dictionaries are separated with support vector machine (SVM) multi-class classifiers by using a kernel map to transform the (SVM) into a linear one [CARITAFELIZ].

In contrast to the SIFT algorithms used to extract the feature points of images and the obtained SIFT features are used to generate a fixed number of visual words, PHOW uses this bag of words with pyramid matching. PHOW eventually has more information of the image spatially and globally.

The texton methodology is based on detecting the response of an image to a filter bank (which often is based on horizontal, vertical, diagonal lines) and it represent an image by the overall distribution (histogram) of the textons (k-means of responses of the filters, per pixel) , In that sense, an object is distinguishable from other by its global distribution of textons, ignoring spatial information. On the other hand, PHOW establishes object specific features at local scale (patches) which highly associate spatial related objects. Also, it considers several scales of the image in order to obtain features at diverse resolution sizes of the image. Because PHOW doesn't ignore spatial information (a limitation of histogram representation) and relates features at different scales, PHOW is a worth strategy as a classifier of various categorical objects. Additionally, some parameters determined the way PHOW strategy works e.g Size, Step, numSpatialX/Y and others are related with the amount of information that was to be used for the training method e.g. number of train images, number of the dictionary of words, number of categories to train the model and others like the parameter C are inherent for the multiclass SVM that is going to be trained. The parameters Size, Step determine the size of the window and the number of steps in which the window is going to be moved. The numSpatial parameter determines the number of scales that are going to be used on the spatial pyramid. The number of Words determines the number of features (visual words) that are going to be extracted. The number of categories and the number of train images determine how many classes and how many images from each of those classes are going to be used to train the final model. In addition the parameter C determines the desired confidence to classify the samples, in other words it determines how susceptible is going to be the model to misclassify the train samples.

*1) Caltech101 Methodology:* The best parameters for Caltech101 data were set by running several experiments starting from the default parameters and then varying one parameter at the time and analyzing it's effect on the ACA for the test set. First, it was necessary to understand the relative behavior of the parameters Size and Step on the ACA, for that purpose a specific value of Size stayed constant while the number of steps were increasing, then the Size was increased and test for the same values of the steps. Then, the number of classes/categories was increased from 5 (tiny problem) to 102 (whole problem). Once the number of categories was set on the maximum (whole problem) the number of images to train for every class was increased from 15 to 31, because in the Caltech101 dataset many classes were unbalanced in their number of images and classes like "inline_skate", "metronome" and "binocular" had around 31 and 35 images. For that reason, 31 was the maximum number of images to train in order to maintain balanced classes. After that the numSpatial parameters for X and Y were changed from [2] to [2 4 6 8 10 12] simultaneously and their effect on the ACA was evaluated, the value was raised until 12 because this parameters increases severely the computation time (due to the amount of scales asking to be computed). Next, the number of words in the dictionary (can be also seen as the K) was adjust from a lower level than the default and slowly raised until 3500 in order to see the which value gives the higher ACA. Finally a re adjustment was done to the parameters Size and Step in order to establish the final values for this parameters and values from 0.1 to 50 were tested for the confidence parameter of the SVM model. At the end, the final parameters that were selected were those which presented a significant and positive effect on the resultant ACA of the model.

*2) ImageNet200 Methodology:* To vary the parameters of ImageNet200, we took as a base some parameters that were found as better in the Caltech101 database. For example, for the number of steps and their influence on the ACA of the test group, only the size of 3 and 5 were taken, since they were found to be better in Caltech101. Likewise, the parameters were not varied in very distant ranges where the results in Caltech101 were too low with respect to the best obtained. In ImageNet200 the number of train images was first varied to obtain the best value for the following experiments, since it is a key parameter as it was found in the Caltech101 experiments (fig 3). Subsequently, the number of steps was evaluated which was also one of the most significant parameters for the development of bag-of-words (fig 1). Finally, with the best parameters previously found, numSpatialx-y was associated with the scales of the pyramid in PHOW and the C associated with the SVM classifier.

# III. Results

## A. Caltech101 Results

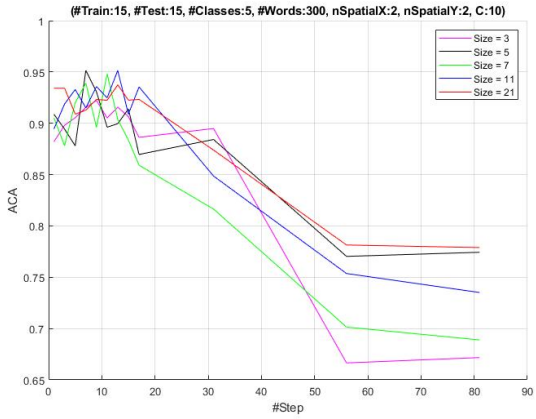



Figure 1. Comparison of various Size and Step values and their effect over the ACA.

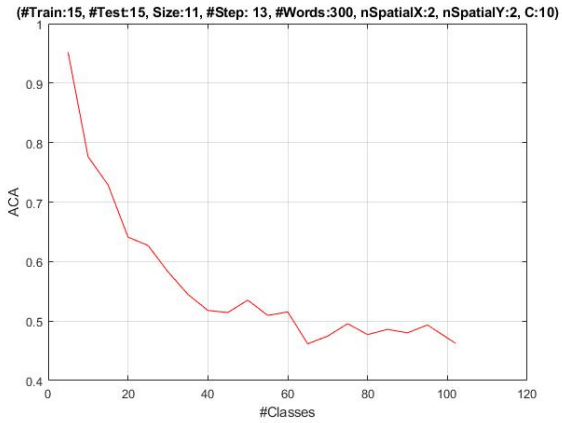



Figure 2. Effect of the number of categories/classes over the ACA.

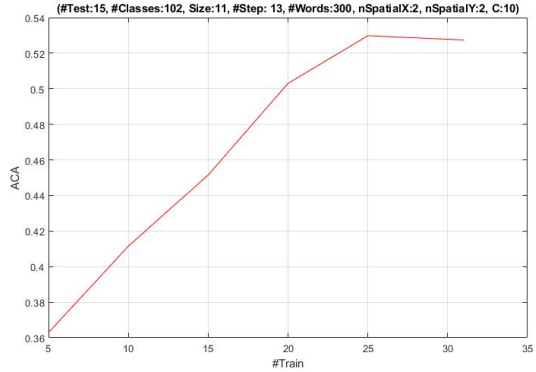



Figure 3. Effect of the number of training samples per class over the ACA.

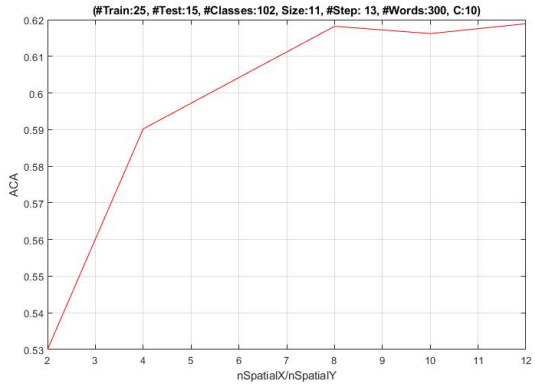



Figure 4. Effect of the number of numSpatial on X and Y over the ACA.

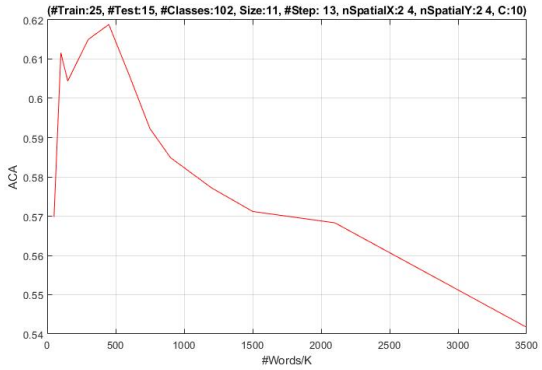



Figure 5. Effect of the number words used (K) on the ACA

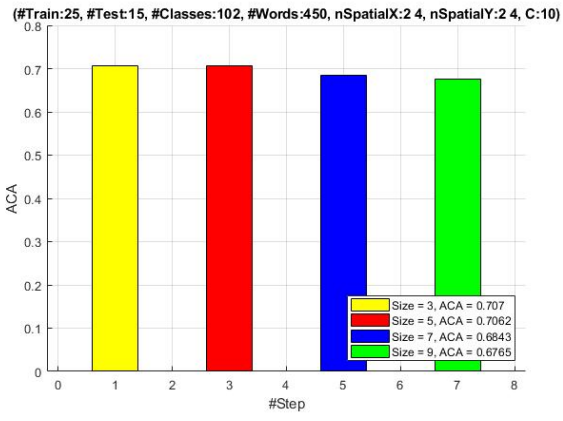



Figure 6. Recalculation of small Size and Step values and their effect over the ACA.

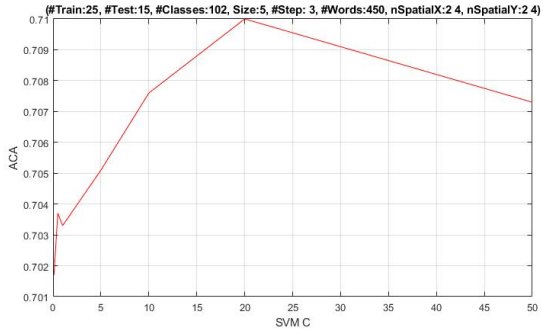
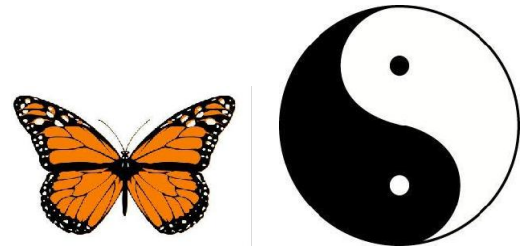



Figure 7. Effect of the SVM parameter C on the ACA.

| Size | Step | Classes | Train | Words | C | nSpatX/Y | ACA |
|---|---|---|---|---|---|---|---|
| 5 | 3 | 102 | 25 | 450 | 20 | [2 4] | 0.7068 |

Table I
FINAL (BEST) PARAMETERS USED ON CALTECH101 DATASET.



(a) Butterfly class, slightly challenging (b) Byinyang class, slightly challenging

(c) Accordion class, slightly challenging (d) Beaver class, highly challenging

(e) Water lily class, highly challenging (f) Bass class, highly challenging

Figure 8. Comparison of some slightly challenging and high challenging classes.

## B. ImageNet200 Results

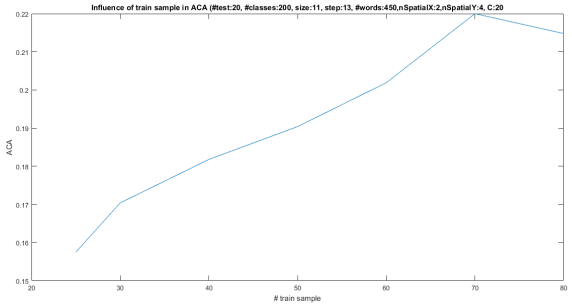



Figure 9. Effect of the train sample on the ACA- ImageNet200

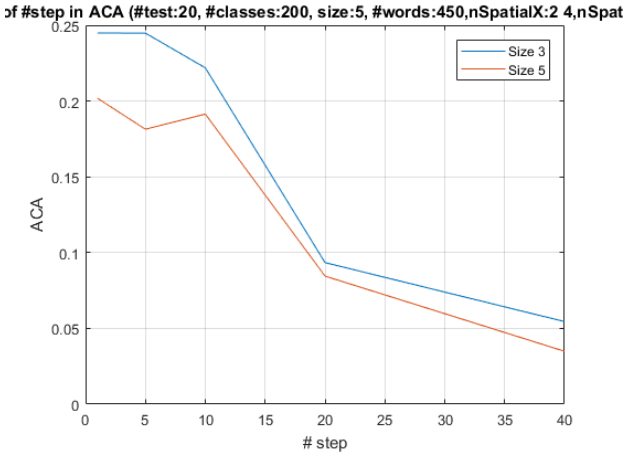



Figure 10. Effect of the #step on the ACA- ImageNet200

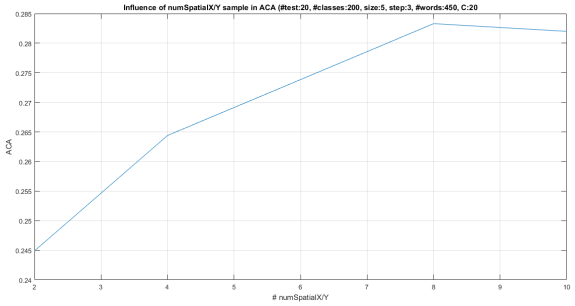



Figure 11. Effect of the /Y on the ACA- ImageNet200

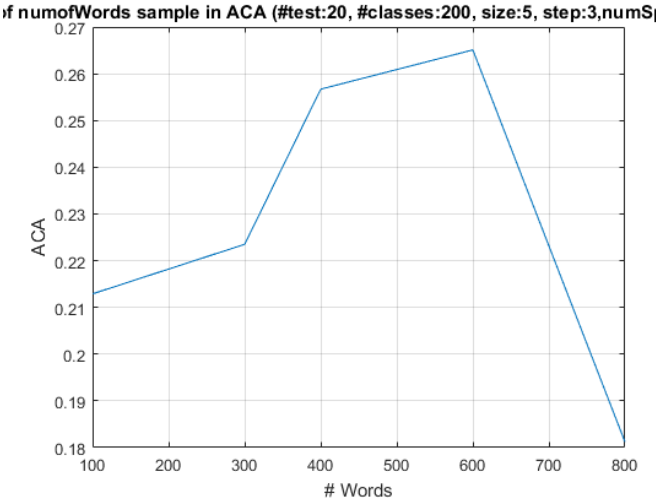



Figure 12. Effect of the #Words on the ACA- ImageNet200

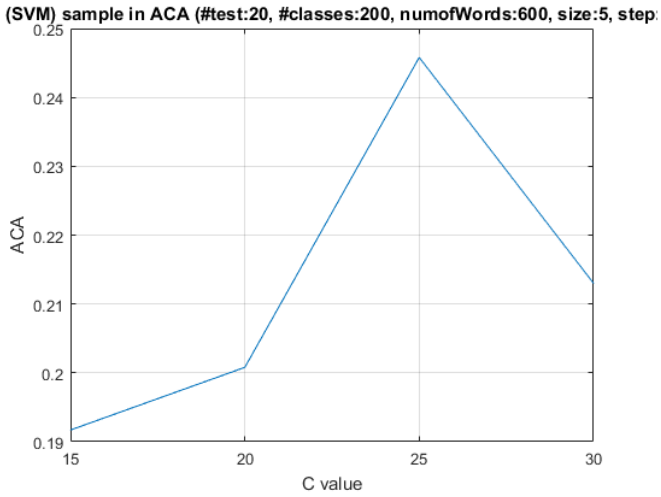



Figure 13. Effect of the #C on the ACA- ImageNet200

| Size | Step | Classes | Train | Words | C | nSpatX/Y | ACA |
|---|---|---|---|---|---|---|---|
| 5 | 3 | 200 | 25 | 600 | 25 | [2 4] | 0.2458 |

Table II
FINAL (BEST) PARAMETERS USED ON IMAGENET200 DATASET.

## IV. DISCUSSION

### A. Caltech101

As can be seen from the figure 1 the parameters Step and Size gave the best results when they were set close to each other. Also, in that figure can be seen that small values (¡11) of steps and size presented overall better ACA. This may happen because bigger sizes of window tend to normalize and ignore small details due to the amount of pixels that are being taken into account at the same tame also if the number of steps is bigger than the window, the amount of pixels that are being ignored increases and because of that many details are getting lost and as a consequence the overall ACA decays. The figure 2 presents an inversely relation between the increase of categories and the ACA, this happens because of the few amount of training samples per class compared to the number of classes that are being used to the train the model, more classes represent more variation (error is introduced to the model) and no more samples are being provided in order to diminish that variation and avoid misclassification. In contrast, as shown in figure 9 increasing the number of training samples for every class (102) also increases the ACA. In addition, considering more scales for the pyramid as shown in figure 4, as expected increases the ACA. Nevertheless, the ACA increases considerably when more than 1 scale is considered but there is no significant difference in considering too much scales because when there are many scales considered there's no new information being taken into account but there's just redundant information. As shown in the figure 5 taking into a account to few visual words creates a poor model to represent the samples because dissimilar groups are trying to be explained by the same set of visual words, too few of visual words that are not representative for the samples. Nevertheless, using too much numbers of visual words generates that the same type of samples are often described by noise (meaningless details) which makes the SVM model more susceptible to misclassify, same categorical samples are being explained by different visual words because of the amount of them. Finally, as expected, the figure 6 a small size and small steps consider small details without ignoring relevant pixels and because of that the obtained ACA is better than experiments with bigger size and steps. Also, as seen in the figure 15, there's no significant difference between models with different values of C (confidence) for the construction of the hyperplane, higher values of C hypothetically should be decreasing the tendency to misclassify and as a consequence increase the ACA. Nevertheless, this is not happening because the data that is being misclassified is insensitive to the margin of the hyperplane, i.e. the misclassified samples cannot be discriminated due to their location (mixed between properly classified samples). This may be a direct limitation of the algorithm itself and the parameters selected (because of it's inability to obtain discriminative features) rather than a problem of the SVM model in classifying the samples.

Some of the classes that presented to be the less challenging, i.e. presented the overall best results for the parameters that were set, are the classes 2, 3, 99 (butterfly, accordion and yin yang, refer to the annex to see the confusion matrix of numeric classes). On the other hand, some of the most challenging classes are 0, 7, 98 (beaver, bass and water_lily, refer to the annex to see the confusion matrix of numeric classes). The reason why classes like butterfly, accordion and yin yang tend to be "easier" classes than beaver, bass and water lily is because, as can be seen in figures 8(a), 8(b) and 8(c), classes like yin yang present very distinguishable texture pattern and pretty standard shape and color distribution (black and white basically), other like the butterfly class have representative patterns in their wings. Also, the accordion has a rigid shape (most of the time boxes) and the images of this class also tend

to have distinguishable edges that help to distinguish these objects. Additionally, most of the images of these classes are objects in front of a white background, which makes them easier to identify. In contrast, classes like beaver, water lily and bass are difficult classes because, as can be seen in figure 8(d), 8(e) and 8(f), these classes tend to have particular colors that make them be mixed with their background, a certain type of concealment with their environment due to the place the photos were taken. Also, classes like beaver and bass have diverse shapes (some of the images of bass and beaver are mosaic-like from diverse views) because their bodies are not rigid (not objects, animals).

### B. ImageNet200

The number of categories for ImageNet200 was fixed at 200 since the objective was to show how a greater number of categories and a different origin of the images could affect the performance of the algorithm. In addition to this, it was clearly evident in Caltech101 that while the number of categories increased, the resulting ACA decreased remarkably due to the diversity of images and that as the categories increase the problem becomes more complex.

In relation to the number of train images for training, it is evident that increasing this number increases the ACA since a greater variability between images is obtained that our model can describe. However, this number as shown in Figure 9 tends to stabilize because the problem and the error is not due to the variability between data but to the shortcomings in the descriptors of the images that fail to get all the attributes of them . In general, the results are similar to those of Caltech101 and from 70 train images this number tends to stabilize.

The behavior of the number of words was slightly different from that presented in Caltech101, requiring a greater number of word dictionaries to achieve a better result in the classification (600 vs 450, fig 5, fig 12). This may be due to the greater diversity of classes (200 vs 100) and greater complexity of images in this database of natural objects.

The results of the other parameters were similar in behavior to the one presented by Caltech101, however, it is worth mentioning the large difference that was present in both datasets. This is mainly due to the fact that in ImageNet200 the number of images is much greater, increasing the complexity of the problem in terms of variability and the complexity and diversity of the images. PHOW gives encouraging results, however, the results of ImageNet decrease drastically, evidencing the limitations of the PHOW method. It may be that only the descriptors of word numbers are not enough and more descriptors are required (eg color histograms, texture, intensity) to better represent the variability between the images and obtain more encouraging results than those achieved with the bag-of-words

## V. Conclusions

Due to the way PHOW algorithm works, all regions of an image are being analyzed and this might be not the optimal choice because often the object of interest is not occupying the whole image. In reality, many of the pixels of the image might be occupied by the background and just a small portion of the total image size will be pixels related to the object. Because of that, PHOW is frequently taking a lot of noise information from non-interest zones and inducing error into the final model which may represent a limitation on the classification of the image categories. A possible solution for that might be to establish a pre-processing phase which eliminates noise due to the photo itself (illumination, blurred, etc.) but in addition a robust probabilistic model and a segmentation method (or a boundary detector gPb might be a suitable option that haves a probabilistic characteristic) that establishes the probability of each pixel to be part of an object of interest. With that method some inferences can be done over the image and a list of potential candidate regions of pixels ( frequent pixels surrounded by various boundaries if using gPb) can be generated in order to refine the sub sequential method. Then, in the actual PHOW algorithm, features must be obtained just on those pixels with a high confidence (given by the probabilistic model or using an optimal ultrametric distance on gPb) of being part of objects. This might cause that the relative noise decrease and more significant visual words are being obtained thus increasing the ACA. Finally, it may be appropriate to use more representation descriptors to overcome the diversity between images. Authors such as those of [Naeem] report the use of PHOW of colors, intensity and even of textures with a more satisfactory performance in the classification of larger datasets such as ImageNet

## References

[1] Wang, Yutian, et al. “Algorithm of near-Duplicate Image Detection Based on Bag-of-Words and Hash Coding.” Journal of Computer Applications, vol. 33, no. 3, 2013, pp. 667–669., doi:10.3724/sp.j.1087.2013.00667.

[2] Y. Ke, R. Sukthankar, PCA-SIFT: A more distinctive representation for local image descriptors, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2004) 506–513

[3] Malki, Zohair. “Shape and Geometric Features-Based Semantic Image Retrieval Using Multi-Class Support Vector Machine.” 2017, doi:10.20944/preprints201702.0077.v1.

[4] Vedaldi, Andrea, and Brian Fulkerson. “Vlfeat.” Proceedings of the International Conference on Multimedia - MM 10, 2010, doi:10.1145/1873951.1874249.

[5] Ajeesh, S. S., et al. “Performance Analysis of Classification Algorithms Applied to Caltech101 Image Database.” 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014, doi:10.1109/icicict.2014.6781364.

[6] Deselaers, Thomas, and Vittorio Ferrari. “Visual and Semantic Similarity in ImageNet.” Cvpr 2011, 2011, doi:10.1109/cvpr.2011.5995474.

## Annex: Extra figures

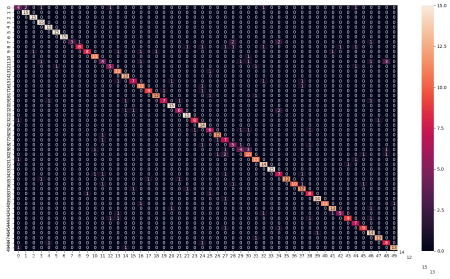


Figure 14. Confusion Matrix for the final parameters (first 50 classes),

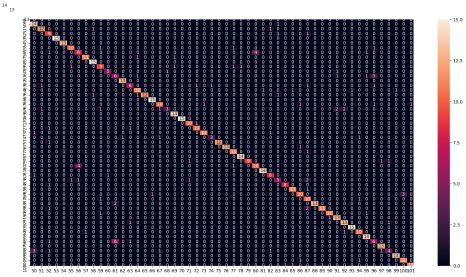


Figure 15. Confusion Matrix for the final parameters (last 52 classes),