Applied Computer Science, vol. 14, no. 4, pp. 5–11
doi:10.23743/acs-2018-25

Submitted: 2018-06-03
Revised: 2018-09-20
Accepted: 2018-10-15

BOVW, classification, codebook

Baldemar ZURITA [0000-0002-1443-8260]*,
Luís LUNA *, *José HERNÁNDEZ* *, *Federico RAMÍREZ***

BOVW FOR CLASSIFICATION IN GEOMETRICS SHAPES

Abstract

The classification of forms is a process used in various areas, to perform a classification based on the manipulation of shape contours it is necessary to extract certain common characteristics, it is proposed to use the bag of visual words model, this method consists of three phases: detection and extraction of characteristics, representation of the image and finally the classification. In the first phase of detection and extraction the SIFT and SURF methods will be used, later in the second phase a dictionary of words will be created through a process of clustering using K-means, EM, K-means in combination with EM, finally in the Classification will be compared algorithms of SVM, Bayes, KNN, RF, DT, AdaBoost, NN, to determine the performance and accuracy of the proposed method.

1. INTRODUCTION

The classification of forms is an intriguing and challenging problem found in the intersection of computer vision, geometry processing and machine learning (Li, & Ben Hamza, 2014; Ben Hamza, 2016; Ye & Yu, 2016). The form is an intrinsic characteristic for the understanding of the image, which is stable to illumination and variations in the color and texture of the object. Due to these advantages, the form is widely considered for object recognition (Shaban, Rabiee, Farajtabar & Ghazvininejad, 2013; Wang, 2014; Jia, Fan, Liu, Li, Luo & Guo, 2016). The contours of a form are main characteristics and of great importance for their classification, from these characteristics we can describe the form.

* Apizaco Technological Institute, Computer Systems Department, Apizaco, Tlaxcala, Mexico, e-mail: baldemar.zurita@gmail.com
** Apizaco Technological Institute, Departament of Computer and Systems, Apizaco, Tlaxcala, Mexico

The Bag of Visual Words (BOVW) model proposed by Szelinski (Szelinski, 2011), is based initially on the work of Sivic and Zisserman (2003) as Bag of Words for natural language processing. This method is widely used in classification of objects with excellent results, the main idea of this method is to obtain the description of the images of a training set to generate a codebook or book of visual words, later a classification algorithm can decide the class to which it belongs.

2. CLASSIFICATION BY BOVW METHOD

A classification system that uses the BOVW method has the following phases:

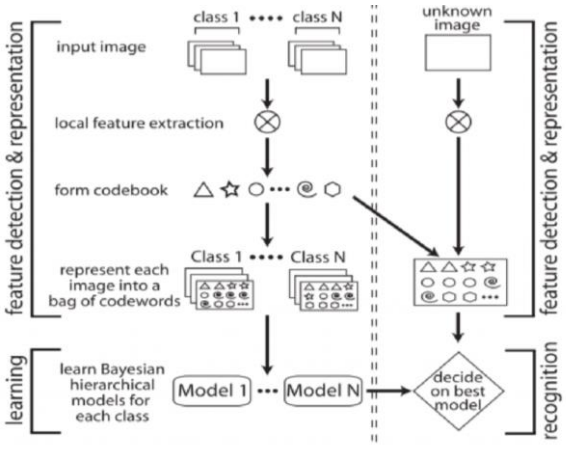

Fig. 1. BOVW model

A classification system that uses the BOVW method has the following phases:
1 – Collect a data set of examples.
2 – Partition the data set into a training set, and a cross validation set (80%–20%).
3 – Find key points in each image, using SIFT.
4 – Take a patch around each key point, and calculate it's Histogram of Oriented Gradients (HoG). Gather all these features.
5 – Build a visual vocabulary by finding representatives of the gathered features (quantization). This done by k-means clustering.
6 – Find the distribution of the vocabulary in each image in the training set. This is done by a histogram with a bin for each vocabulary word.
7 – Train an SVM on the resulting histograms (each histogram is a feature vector, with a label).
8 – Test the classifier on the cross validation set.

If results are not satisfactory, repeat 5 for a different vocabulary size and a different SVM parameters or different classifier.

3. METHOD

We will work with one sets of training, it was obtained from 50 individuals tracing different geometric shapes, 210 images, using 90% for training and 10% for testing, is divided into three classes: triangle, circle and square.

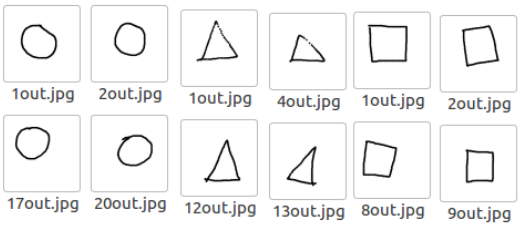

Fig. 2. Dataset of geometric figures

For the modeling of BOVW, a detector and descriptor SIFT and SURF with grouping by K-means, EM and its combination of both K-means + EM classifiers will be used, for the classification step seven algorithms were chosen: vector support machines (SVM) - Supor Vector Machine), random forests (RF – random forest), decision trees (DT – desicion tree), adaptive increase (AdaBoost – adaptive boosting), Bayesian (Gaussian NB), neural network (Neural Network) and nearest neighbors (KNN – k-nearest neighbors) In order to evaluate the performance of each model, each method will be executed 6 times with different values of k (50, 100, 200, 300, 400, 500) in order to obtain the average precision of classification followed by the generation of performance graphs (precision and k-value), cross-validation and confusion matrix analysis will be used to determine the best geometric shape classification model.

All experiments will be executed in a linux operating system, CPU 2.5 Quadcore 64bits, 8 GB of RAM.

4. RESULTS

In the following graphs, the performance of each model with detector and descriptor SIFT and SURF with Clustering Kmeans, EM and Kmeans + EM is shown. Each model was executed with different values of k (dictionary size).

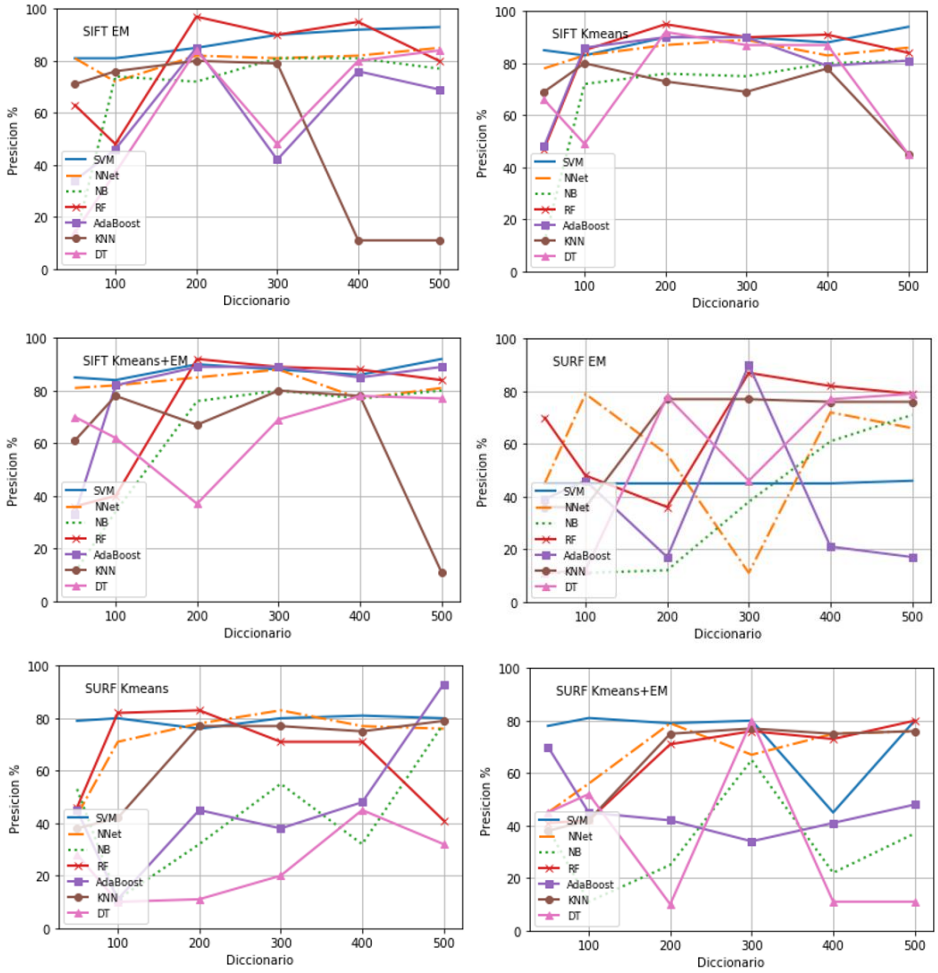

Fig. 3. Comparative chart of the performance of each model

It can be seen that the SIFT descriptor and detector is more stable with the clustering algorithms EM, Kmeans and Kmeans + GM, since a better precision is obtained at the time of the classification for each value of k, in appearance, being gradually more accurate to as the dictionary size increases. Obtaining the values of the cross validation (accuracy, recall, f1 score) can have a better decision on which is the best combination of descriptor, grouping algorithm and classifier, the following comparative table was generated choosing the models with greater accuracy than 90%.

Tab. 1. Results

Detector Cluster	Classifier	K	Precision	Recall	F1-score	Matrix
SIFT EM	RF	200	97	97	97	[20 0 0] [1 19 0] [0 1 19]
SIFT Kmeans	RF	200	95	95	95	[19 1 0] [0 19 1] [0 1 19]
SIFT Kmeans	SVM	500	94	93	93	[18 0 2] [0 18 2] [0 0 20]
SURF Kmeans	AdaBoost	500	93	92	92	[19 1 0] [0 20 0] [0 4 16]
SIFT Kmeans+EM	RF	200	92	90	90	[15 5 0] [0 20 0] [0 1 19]
SIFT Kmeans+EM	SVM	500	92	92	92	[18 1 1] [0 18 2] [0 1 19]
SIFT Kmeans	DT	200	92	92	92	[20 0 0] [0 17 3] [0 2 18]
SIFT EM	SVM	400	92	92	92	[18 0 2] [0 18 2] [0 1 19]
SIFT Kmeans	RF	400	91	90	90	[20 0 0] [3 17 0] [0 3 17]
SIFT Kmeans+EM	SVM	200	90	88	88	[15 3 2] [0 18 2] [0 0 20]
SIFT Kmeans	SVM	200	90	88	88	[14 4 2] [0 19 1] [0 0 20]
SIFT EM	SVM	300	90	88	88	[15 2 3] [0 18 2] [0 0 20]

In this table we can see the best classification results by means of the confusion matrix, which is a tool that allows to visualize the level of precision of a classifier, the rows of the matrix represent the images of the evaluation set (ground-truth) , each row is a different class: circle, square and triangle in a descending order, in the columns we have the same order of the classes from left to right.

Accuracy is measured by calculating the sum of the diagonal of the matrix, which represents the correctly classified images, among the total number of images in the matrix. This table shows the average of the accuracy of the three classes that represents the quality of the classifier response. Precision = TP / TP + FP.

The sensitivity (recall) measures the efficiency in the classification of all elements of the class by means of the calculation of the real positives between the sum of the real positives and the false positives. The average of the sensitivity of the three classes is shown in the table. Recall = TP / TP + FN.

The F1 score can be interpreted as a weighted average of the precision and sensitivity, where a score F1 reaches its best value at 1 and the worst score at 0. F1 Score = 2 * (Recall * Precision) / (Recall + Precision) .

The matrix shows us in its diagonal, the number of correctly classified images. In this case, the model that proved to have the best accuracy is the SIFT detector combination, EM grouping algorithm, k = 200 (200 word dictionary) and the Random Forest classifier. For this model, an additional validation method was added: cross validation.

SIFT EM k = 200 Random Forest
Classification Report

	precision	recall	f1-score	support	Confusion Matrix
Circle	0.95	1.00	0.98	20	[20 0 0]
Square	0.95	0.95	0.95	20	[1 19 0]
Triangle	1.00	0.95	0.97	20	[0 1 19]
avg / total	0.97	0.97	0.97	60	Accuracy Matrix: 96.6 %
					Cross Validation RF: 95.0 %

Fig. 3. Results of the model with better precision with cross validation

Cross-validation is a technique used to evaluate the results of a statistical analysis and ensure that they are independent of the partition between training and test data.

5. CONCLUSION

The classification of images with the BOVW model is fascinating, since it shows an acceptable precision in its labels, in this case, only a dictionary size of 200 words (k = 200) was necessary to be able to classify with accuracy greater than 95% the dataset of geometric shapes exposed in this project, which leads to a lower computational effort, shorter processing time, smaller size in Kb of the dictionary and a faster response by the classifier.

Because the dataset does not contain much information, only a shape descriptor was used, there are other descriptors such as color, which in combination make a more robust system, even classifying more complex images, such as training the algorithm that can discriminate different types of roses for their color, since the shape would be the same, but its variation in color makes it different from the others and therefore, it is more appropriate to use the description of the color. It can be inferred that the success of this model is based on the objects to be classified, since each phase of the method has to be adjusted, changing descriptors, algorithms, classifiers, until obtaining the best result, by means of evaluation methods.

As future work, you could use other grouping algorithms, such as BIRCH or Fuzzy Kmeans, with the aim of raising the accuracy rate to 99%, as well as increasing the size of the dataset and testing this model with other datasets used in computer vision like Caltech101.

REFERENCES

Ben Hamza, A. (2016). A graph-theoretic approach to 3D shape classification. *Neurocomputing*, *211*, 11–21.

Jia, Q., Fan, X., Liu, Y., Li, H., Luo, Z., & Guo, H. (2016). Hierarchical projective invariant contexts for shape recognition. *Pattern Recognition*, *52*, 358–374. doi:10.1016/J.PATCOG. 2015.11.003

Li, C., & Ben Hamza, A. (2014). Spatially aggregating spectral descriptors for nonrigid 3D shape retrieval: a comparative survey. *Multimedia Systems*, *20*(3), 253–281. doi:10.1007/s00530-013-0318-0

Shaban, A., Rabiee, H., Farajtabar, M., & Ghazvininejad M. (2013). From local similarity to global coding; an application to image classification. In: *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2794–2801). Portland, USA: IEEE.

Sivic, J., & Zisserman, A. (2003). Video Google: a text retrieval approach to object matching in videos. In: *Proceedings of the Ninth IEEE International Conference on Computer Vision – Volume 2* (pp. 1–9). USA: IEEE Computer Society Washington.

Szelinski, R. (2011). *Computer Vision: Algorithms and Applications* (pp. 658–729). Springer Verlag.

Wang, X., Feng, B., Bai, X., Liu, W., & Latecki, L. J. (2014). Bag of contour fragments for robust shape classification. *Pattern Recognition*, *47*(6), 2116–2125.

Ye, J., & Yu, Y. (2016). A fast modal space transform for robust nonrigid shape retrieval. *The Visual Computer*, *32*(5), 553–568. doi:10.1007/s00371-015-1071-5