Aggregated Color Descriptors for Land Use Classification

Vedran Jovanović and Vladimir Risojević

Abstract — **In this paper we propose and evaluate aggregated color descriptors for land use classification in aerial images. First, global and local Bag-of-Colors (BoC) descriptors are evaluated for land use classification. Influence of different parameters on performance and efficiency of classification were tested. Small modification in process of computing of BoC is introduced, which improves the overall classification performance. We also present new, very simple color descriptor, termed Vector of Locally Aggregated Colors (VLAC) which has as good classification results as modified BoC, but using linear support vector machines.**

Keywords **— BoC descriptor, VLAC descriptor, color descriptors, Land Use Classification**

I. Introduction

In this paper we consider the problem of land use classification in high-resolution overhead imagery. The main challenge in this area consists of finding powerful descriptor and classifier, which will distinguish different types of images, based on their content and assign them into different classes, and on the other hand, be invariant to all transformations and factors that can decrease quality of image such as cropping, rotation, scaling, noise, etc.

There are a vast number of papers related to descriptor generation. Many of them are based on using descriptors extracted locally from image patches, in particular Scale Invariant Feature Transform (SIFT) descriptor and its modifications [1], [2]. These descriptors are often jointly used with Bag-of-Visual-Words (BoVW) framework, where, starting from a set of local descriptors, fixed size vector for each image were computed.

Another way for combining local descriptors into fixed size vector is Vector of Locally Aggregated Descriptors (VLAD) framework, introduced by Jegou *et al*. [3], [4], which reduces the amount of data needed to obtain final descriptor, without noticeably impacting its accuracy [5], [6]. VLAD descriptors show excellent performance and accuracy in systems for large-scale image search and retrieval [3] - [6].

Although very powerful, most of descriptors based on SIFT work only with gray-level images, and do not take into account color information.

Color is a very expressive visual feature, and as such, it could be a very helpful cue in systems for recognition and object detection, since different object types have different colors. One notable example is land cover/land use classification in remote sensing images, where different land cover types have different colors (forest and grass is green, water is blue, etc.).

One of the first color descriptors was described in [7], where color histogram was introduced. Since then, different color descriptors were presented, and some of them are also included in MPEG-7 standard for description of audio-visual content [8].

In [9] Bag-of-Colors (BoC) descriptor has been introduced that uses adaptive approach for color codebook computation, in contrast to fixed color partition, traditionally used with color descriptors. Then, this descriptor was tested in [10] for use in aerial image classification.

The main contributions of this paper are:

- BoC descriptor for land use classification is evaluated,
- Modification in the BoC computation process is proposed, which improves classification performance,
- New color descriptor, termed Vector of Locally Aggregated Colors (VLAC) is introduced,
- Influence of different parameters and normalization techniques on classification accuracy is tested.

Paper is organized as follows. In Section II we describe the ways of computing global and local BoC descriptors. Procedure for computing VLAC descriptor is introduced in Section III. In Section IV we describe used image dataset, methodology of testing, and present the results obtained by testing both descriptors. In Section V. we give concluding remarks and suggest future work.

II. BoC Descriptor

There are two different ways of BoC computation. If the descriptor is extracted from the whole image, global BoC is obtained, while if the descriptors are computed for a set of image patches, we obtain local descriptors.

Paper received July 21, 2015.

This paper is revised and expanded version of the paper presented at the 22nd Telecommunications Forum TELFOR 2014.

This research was supported in part by Norwegian Ministry of Foreign Affairs through HERD ICT NORBOTECH (NORwayBOsnia TECHnology Transfer) project for Programme in Higher Education, Research and Development under contract 2011/1370.

V. Jovanović, University of Banja Luka. Faculty of Electrical Engineering, Patre 5, 71000 Banja Luka, Bosnia and Herzegovina (tel: +38751-221-874, e-mail: vedran.jovanovic@etfbl.net).

V. Risojević, University of Banja Luka, Faculty of Electrical Engineering, Patre 5, 71000 Banja Luka, Bosnia and Herzegovina (tel: +38751-221-847, e-mail: vlado@etfbl.net).

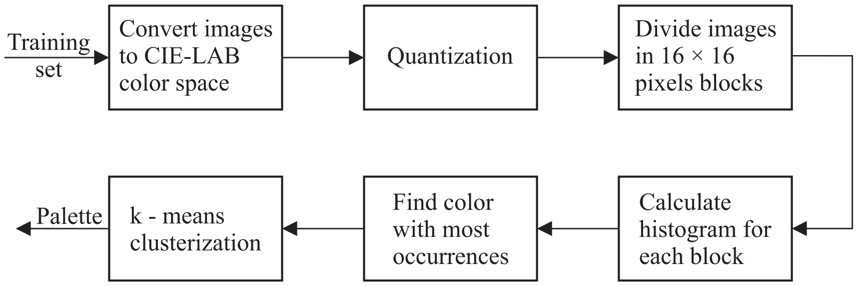

Fig. 1. Block scheme for palette computation process

A. *Global BoC descriptor*

Global descriptor is generated from the whole image. The first step in the process of descriptor computation is generation of a color codebook (*palette)*. Process of palette generation is shown in Fig. 1. First, each image from training set is converted to CIE-LAB color space and each color component is uniformly quantized in N bins. After this, each image is divided in blocks of size 16 × 16 pixels, and the color with most occurrences is extracted for each block. Then, all colors extracted in this way were clustered using k-means algorithm producing a color codebook with k_c colors. In this way we have learned a palette which is more adjusted to real-world images than fixed color space quantization. Palette learned with parameters N = 8 and k_c = 100 is shown in Fig. 2. As a product of palette learning and unbalanced color statistic of aerial images, some colors such as green, gray and brown have bigger contributions to the palette than others, e.g. red or blue.

When the palette is obtained, the next step is BoC generation. Similar to BoVW, we compute image representation as a histogram of codeword occurrences for each image with regard to the specified palette. It is done in a way that for each pixel from image we calculate its Euclidean distances to the colors in the palette, whereupon we increment the histogram bin corresponding to the color with the shortest distance. After this, each image is represented with a fixed-size vector of length k_c - global BoC descriptor.

As the last step, different types of normalization techniques were tested in order to achieve better results. Many of these techniques are also used jointly with SIFT-based descriptors. We tested inverse-document frequency (IDF), power-law, L1 and L2 normalizations, and their combinations. The best results were achieved using only L1 normalization, which is used in further work.

B. *Local BoC descriptor*

The process of computing image representation based on local BoC descriptor is summarized in Fig. 3. The first step is palette generation, which is performed in the same way we described before. This palette will be used later for computation of patch descriptors. The second step is preparation of the input image for descriptor computation, where the image is converted into CIE-LAB color space and quantized by each color component regularly in N bins. In this part we also tested the influence of using power-law transform before descriptor computation [2]. We came to a conclusion that using square root value of each color

Fig. 2. Color palette obtained after learning on real-world aerial images (N = 8, k_c = 100)

component instead of original color information decreases the performance of the final descriptor.

Similarly to global BoC, the next step is the computation of BoC descriptor for each patch of the image, with regard to the palette. These patches can be obtained in many ways. They can be extracted by a region detector, regularly sampled, etc. In this paper, patches are regularly sampled.

Now, each image is represented with a bag of local BoC descriptors. In order to compute the codebook of local BoC descriptors, we apply k-means algorithm on descriptors obtained from images from training set. Length of obtained codebook after clustering is M.

The last step of this process is image representation extraction. In this case image representation is actually a histogram of codeword occurrences with regard to the codebook of local BoC descriptors. For each image, which is represented with a bag of local BoC descriptors, we calculate Euclidean distances between these descriptors and codebook centroids and increment the bin in the final histogram, corresponding to the centroid with the shortest distance. At the end of this process each image is represented with a single vector of length M.

III. VLAC DESCRIPTOR

Similarly to BoC, we propose both global and local versions of the VLAC descriptor.

A. *Global VLAC descriptor*

In order to extract global VLAC descriptor, we need to generate the color palette. After palette generation, instead of histogram computation as we did in BoC framework, we use VLAD approach in order to produce the final descriptor. Standard VLAD representation is formed as a sum of all residual vectors, which represent differences between the local descriptor and the centroid

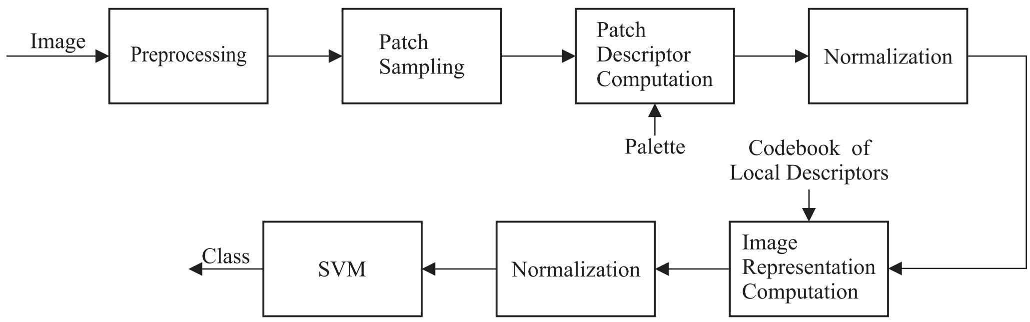

Fig. 3. Block scheme for computing of image representation based on local BoC and VLAC image descriptors

it is assigned to [3]. In our case, for each of k_c colors in palette $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots \boldsymbol{\mu}_{k_c}\}$, the differences $\mathbf{x}_t$ - $\boldsymbol{\mu}_i$ of the vectors $\mathbf{x}_t$ assigned to $\boldsymbol{\mu}_i$ are accumulated:

$$\mathbf{v}_i = \sum_{\mathbf{x}_t : NN(\mathbf{x}_t)=i} \mathbf{x}_t - \boldsymbol{\mu}_i \qquad (1)$$

where $\mathbf{x}_t$ represents the color of t-th pixel, and NN($\mathbf{x}_t$) represents its nearest cluster centroid. After this, global VLAC descriptor is obtained by concatenating the residual vectors $\mathbf{v}_i$ into a single vector. Dimensionality of $\mathbf{x}_t$ is 3 (*L, a, b* color component), and it will define dimension of each vector $\mathbf{v}_i$. Since there are k_c centroids (one for each color in palette), length of the final vector obtained after concatenation is $k_c \times 3$.

After this we tested different normalization methods which are commonly used with VLAD framework. L2, power-law transform and normalized components [5] were tested and the best results were achieved using only L2 normalization.

B. *Local VLAC descriptor*

Block diagram which describes the computation of local VLAC descriptor is shown in Fig. 3. This process starts with color palette generation in the same way we already explained. Preprocessing methods and patch extraction for this method are the same as described in Section II.B. The only difference is that, in this case, contrary to BoC, use of square rooted values of colors does not influence the classification results.

In the patch descriptor calculation block, instead of calculation of local histograms for each patch, as we did in BoC framework, in this method, we calculate VLAC representations of patches, given the color palette in a same way as described in (1). Now, each patch is represented with a VLAC descriptor of length $k_c \times 3$. Each image is represented with a bag of local VLAC descriptors, whose number is equal to the number of patches extracted from the image.

Applying k-means algorithm on all local VLAC descriptors computed for training set images, will result in a codebook of local VLAC descriptors of length M. In order to produce final fixed-size image descriptor, in image representation calculation block, we extract VLAD descriptor for each image using its bag of local VLAC descriptors and codebook of local VLAC descriptors obtained using k-means algorithm. The obtained vector is image representation based on local VLAC descriptors and its length is $K = 3 \times k_c \times M$.

IV. EXPERIMENTS

A. *Dataset and methodology*

In order to test effectiveness of the proposed descriptors we chose UC Merced Land Use (UCMLU) dataset. UCMLU is a set of aerial images, manually extracted from USGS National Urban Area Imagery collection, and divided in 21 classes, depending on different land covers and objects shown in them. Each class consists of 100 images, with size of 256 × 256 pixels, and spatial resolution of 30cm per pixel. Name and one sample of each class are depicted in Fig. 4.

For classification with both descriptors, we chose Support Vector Machine (SVM) classifier. We tested linear SVM without and with nonlinear χ^2 descriptor mapping [11]. Drawbacks of nonlinear mapping are higher memory usage and slower training, since it increases dimensionality of final descriptor. Process of testing starts with dividing images randomly into training and test sets, where 80 images from each class were assigned to training and 20 to test set. After this, we extract the color palette (for both BoC and VLAC descriptors), codebook of local BoC descriptor, and codebook of local VLAC descriptors only using training images, whereupon we calculate the final descriptor as a representation for each image.

Then, normalization is applied to descriptors $\bar{\mathbf{x}} = \mathbf{x} / \|\mathbf{x}\|_p$, where p = 1 for L1, and p = 2 for L2 normalization. We found out that L1 normalization is better in combination with BoC descriptor, while L2 normalization is preferred with VLAC descriptor.

As an accuracy measure, we calculated ratio of correctly classified test images and their total number. Ratio is computed for ten different training/test set splits, whereupon the classification accuracy for each set of parameters is presented with mean value and standard deviation of accuracies for different dataset splits.

B. *BoC descriptor*

Summary of preliminary results obtained using BoC descriptor presented in [10] are shown in Table 1. As we can see, classification accuracy of standard BoC descriptor (S), is very dependent on both the number of quantization levels, N, and size of palette, k_c. The reason for that is the following. With increasing k_c we increase the number of different colors that could be found in palette, and also we

Fig. 4. UC Merced Land Use Dataset classes with an example

increase diversity among the shades of colors which is very useful in order to distinguish images which are globally colored same. For example, to distinguish grass from forest, information only about green color is not enough. However, if we have available more shades of green, we will probably could distinguish this two classes since the grass is much lighter than forest on images. On the other hand, increasing the number of quantization levels decreases loss of color information on image, which also makes influence on different color shades in palette. Besides influencing the classification accuracy, these parameters also influence classification time, which increases with increasing of N or k_c. For example, if k_c increases, the number of centroids in process of k-means clustering increases, and it will take more time to find nearest clusters. Also, greater k_c means higher dimensionality of final image descriptor, which will also increase time of classification.

In order to eliminate dependence of classification accuracy on parameter N, we introduce a small modification in the process of palette generation. Instead of using the whole color space range of an image in process of quantization (in MATLAB for CIE-LAB color space this range is [0, 100] for L component and [-128, 127] for a and b color components), we compute the color range for quantization of each image dynamically, using the extreme intensity values for each image component separately (for image i, quantization range is calculated as [min(L_i), max(L_i)] for L, [min(a_i), max(a_i)] for a, and [min(b_i), max(b_i)] for b color component, where L_i, a_i, b_i are color components of i-th image in CIE-LAB color space). As we can see in Table 1. classification accuracy of this modified method (M) is still very dependent on parameter k_c, while parameter N almost does not have any influence on accuracy. Reason for that lies in the fact that each image brings its own range of colors, which differs from image to image, and therefore, a large number of different colors contribute to the process of clustering. For standard BoC, the number of different colors in the process of clustering was always lower or equal to N^3, and from the results it is visible that increasing length of the palette over N^3 does not improve the accuracy of classification. The advantage of our modified method is not only in improving the accuracy, but also in improving of the performances of the whole system. In palette generation process one step is finding color with most occurrences in the histogram with N^3 elements (colors). If N increases more memory is needed, and also it takes more time to search the histogram. In Table 1. we can see that classification accuracy over 86% for modified method is achieved with $N = 4$, while for standard BoC, N should be greater than 64 in order to achieve same accuracy.

Results obtained for local BoC descriptor are shown in Fig. 5. Some of this results were also presented in [10]. According to the conclusions we made for global BoC descriptor, we chose modified method for palette generation, in order to eliminate dependency of results on parameter N. Number of quantization levels per color component was $N = 8$. Patch was defined as a block of 16 × 16 pixels, which was moved over the image with a fixed step of 4 pixels.

From these results we can see that, similarly to global BoC, length of final descriptor M has a huge influence on classification results, and for better results we should choose larger M. But, on the other hand, contrary to global BoC, maximum of classification accuracy, in this case, is obtained with relatively small number of colors in the palette. We can see that maximum of each curve is attained approximately for $k_c = 50$. This is a consequence of patch size. In fact, size of a patch is much smaller from the size of an image, and choosing large value for k_c will result with sparse local histograms in the patch descriptor computation part. Clustering of sparse vectors then results in large quantization errors, which reduces the classification accuracy. Influence of this is best visible on Fig. 5. for $k_c > 100$, where accuracy of classification drastically decreases, when k_c increases.

C. VLAC descriptor

For testing of VLAC descriptor we used similar procedure as we did for BoC descriptor in order to compare the results obtained using both methods. As we described before, difference in generation process between global BoC and global VLAC descriptors, is in the way

TABLE 1: CLASSIFICATION ACCURACIES (%) OBTAINED USING STANDARD GLOBAL BOC (S) AND MODIFIED GLOBAL BOC (M) DESCRIPTOR

			N - Number of quantization levels per color component						
			4	8	16	32	64	128	256
k_c - Length of palette	50	S	65.2 ± 1.4	69.4 ± 2.0	71.6 ± 1.1	72.8 ± 1.5	73.1 ± 1.7	74.0 ± 2.0	73.4 ± 1.7
		M	72.8 ± 1.5	72.2 ± 1.9	73.4 ± 1.5	72.9 ± 1.6	72.6 ± 1.3	74.5 ± 2.4	73.6 ± 1.7
	100	S	63.3 ± 2.8	72.2 ± 2.0	74.1 ± 1.5	76.7 ± 1.5	76.0 ± 1.9	75.9 ± 2.1	77.1 ± 1.7
		M	77.1 ± 2.1	75.8 ± 1.9	76.5 ± 1.7	76.3 ± 1.9	77.3 ± 1.9	77.1 ± 2.1	76.1 ± 2.5
	500	S	63.9 ± 2.1	73.9 ± 1.4	80.0 ± 2.1	82.0 ± 2.1	83.8 ± 1.6	84.0 ± 1.9	84.5 ± 1.8
		M	84.0 ± 2.2	84.5 ± 2.5	84.3 ± 2.2	83.2 ± 1.6	83.8 ± 1.7	84.3 ± 2.0	85.0 ± 2.1
	1000	S	63.8 ± 1.7	72.5 ± 1.9	81.5 ± 2.6	84.4 ± 1.9	85.5 ± 1.9	85.7 ± 2.5	85.8 ± 1.8
		M	85.4 ± 2.0	85.0 ± 2.5	85.2 ± 2.3	85.0 ± 1.8	85.1 ± 1.9	85.5 ± 2.1	84.8 ± 2.1
	2000	S	64.9 ± 2.8	72.4 ± 2.1	82.1 ± 2.6	85.6 ± 2.1	85.9 ± 1.7	85.7 ± 1.7	85.7 ± 1.8
		M	85.8 ± 1.6	86.1 ± 2.0	85.7 ± 1.7	85.8 ± 1.7	85.7 ± 1.7	86.0 ± 1.5	85.9 ± 1.9
	5000	S	67.0 ± 2.0	71.6 ± 2.6	81.5 ± 1.8	85.3 ± 1.8	**86.5 ± 1.8**	85.9 ± 2.2	86.2 ± 1.6
		M	86.3 ± 1.9	86.5 ± 1.9	86.0 ± 1.7	86.2 ± 1.5	86.2 ± 1.5	**86.5 ± 1.7**	86.1 ± 1.7

how a final descriptor is generated from the palette of colors. In BoC framework we calculated histogram of colors with regard to the palette, resulting in the final descriptor of the same length as the palette, while in VLAC framework we used VLAD approach, which produces a final descriptor with three times larger size than the palette. In order to compare the results obtained using different frameworks, the palette used for global VLAC descriptor computation has three times less colors, compared to the palette used for global BoC descriptor computation, so the lengths of the final descriptors are the same. Comparative results are shown in Fig. 6. We can see that, for this dataset, global BoC descriptor result in better performance than VLAC. Reason for that lies in the fact that all images in dataset are colorful, and many classes differ from others mostly in dominant colors, so color histogram gives much more information about picture than similarity to some color (which is the case in VLAC). In this case, we are not really interested in finding the shades of color, because if we want, for example, to distinguish trees from roads, enough is to have information about contribution of green and gray colors in BoC framework.

From Fig. 6. it can also be seen that if we only L2 normalize global VLAC descriptor better results are obtained using nonlinear SVM classifier. However, when we apply square root transform before L2 normalization, linear SVM shows better performance. One of the greatest advantages of using VLAC descriptor is that it gives as good results as BoC, but using linear SVM, which takes less time to train compared to nonlinear SVM, which is visible in Table 2. This will be even more significant when local VLAC descriptor is considered.

For testing of local VLAC descriptor the two main parameters, that were tested, were size of palette k_c and size of codebook of locally aggregated colors M. The value of parameter M was varied between 10 and 100, while the value of parameter k_c was calculated in such a way that the length of the final descriptor K is closest to 5000. Obtained

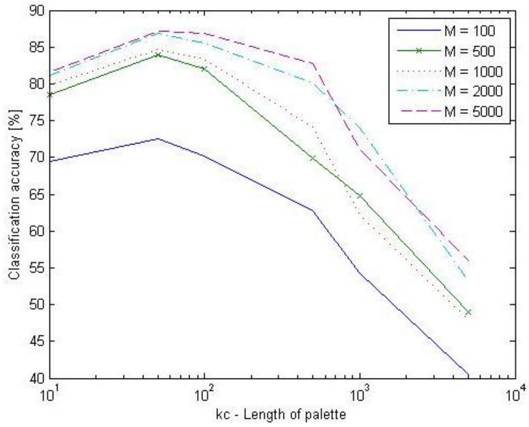

Fig. 5. Classification accuracies (%) obtained using local BoC descriptor

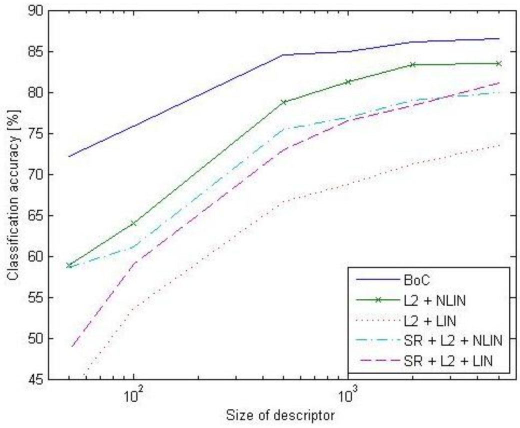

Fig. 6. Comparative overview of classification accuracies (%) obtained using global BoC and global VLAC descriptor with different normalization techniques

results are shown in Fig. 7. For classification we used linear SVM, and according to the conclusions we made for global VLAC descriptor, we used square root transform and L2 normalization before SVM classifier.

Descriptor computation and classification is implemented in MATLAB 2013a, and tested on a

TABLE 2: COMPUTATIONAL TIMES

Palette	$N = 8$, $k_c = 5000$	81 s
	$N = 8$, $k_c = 50$	27 s
Global BoC	0.3 s	
Local BoC	1.3 s	
Global VLAC	2.0 s	
Local VLAC	2.1 s	
Linear SVM	Training	34 s
	Classification	0.2 s
Nonlinear SVM	Training	79 s
	Classification	0.6 s

TABLE 3: CLASSIFICATION ACCURACIES (%) OBTAINED WITH BOC AND VLAC DESCRIPTORS

BoC	Global	Linear	62.8 ± 2.4
		Nonlinear	86.5 ± 1.7
	Local	Linear	78.1 ± 2.6
		Nonlinear	87.1 ± 1.0
VLAC	Global	Linear	81.1 ± 2.1
		Nonlinear	83.5 ± 2.0
	Local	Linear	86.4 ± 1.9
		Nonlinear	**87.6 ± 1.8**

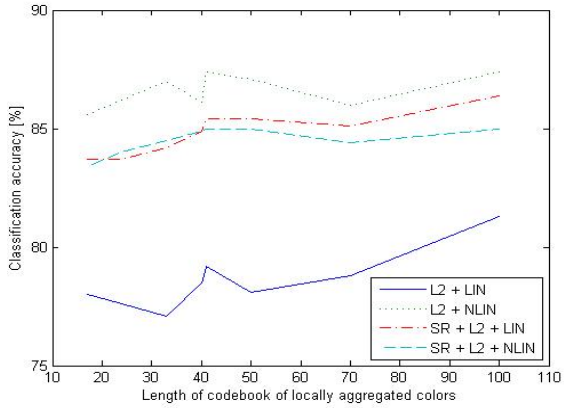

Fig. 7. Classification accuracies (%) obtained using local VLAC descriptor with different normalization techniques

computer with Intel Core i5 - 4300M CPU, 2.6 GHz, 4 cores, 8GB RAM, Windows 7 - 64-bit.

Best results obtained using all four approaches is shown in Table 3. We can see that the best results are again obtained using nonlinear SVM, but the results obtained using local VLAC descriptors and linear SVM are within one standard deviation. Moreover, they are similar to the best results obtained using BoC descriptor with nonlinear kernel. This can be visible in Fig. 8. and Fig. 9. where the confusion matrices for BoC and VLAC descriptors were depicted respectively.

V. CONCLUSION

In this paper we proposed two very simple color descriptors for use in image classification. We find out that descriptors, obtained using only color information, can be a very powerful tool for achieving high classification accuracies. Results show that BoC and VLAC are very effective descriptors, and best of all is that they are very easy to calculate. Since the generation process of these descriptors takes into account only color information, in order to increase performance and accuracy, they could be combined with other descriptors as SIFT, texture descriptors, etc.

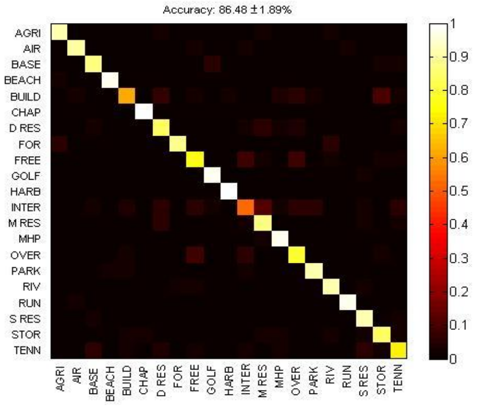

Fig. 8. Confusion matrix for global BoC descriptor with $M = 5000$ and nonlinear SVM.

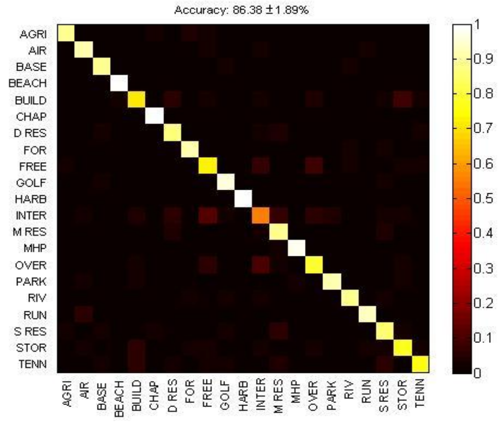

Fig. 9. Confusion matrix for VLAC descriptor with $K = 5100$ and linear SVM

REFERENCES

[1] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification", in ACM SIGSPATIAL GIS, 2010.

[2] R. Arandjelović, A. Zisserman, "Tree things everyone should know to improve object retrieval", *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., pp. 2911-2918, 2012.

[3] H. Jegou, M. Douze, C. Schmid, P. Perez, "Aggregating local descriptors into a compact image representation", *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*., pp. 3304-3311, 2010.

[4] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, C. Schmid, "Aggregating local image descriptors into a compact codes", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.34, no.9, pp. 1704 – 1716, 2012.

[5] R. Arandjelović, A. Zisserman, "All about VLAD", *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 1578-1585, 2013.

[6] J. Delhumeau, P.H Gosselin, H. Jegou, P. Perez, "Revisiting the VLAD image representation", in ACM Multimedia, 2013.

[7] M. J. Swain and D. H. Ballard, "Color indexing", *Int. J. Comput. Vis.*, vol. 7, no. 1, pp.11-32, 1991.

[8] Manjunath, Bangalore S., et al. "Color and texture descriptors" *IEEE Trans. Circuits Syst. Video Technol.*,vol.11, no.6, pp. 703-715, 2001

[9] C. Wengert, M. Douze, H. Jegou, "Bag-of-colors for improved image search" *Proc. ACM International Conference on Multimedia*, pp. 1437 - 1440, 2011.

[10] V. Jovanović, V. Risojević, "Evaluation of Bag of Colors Descriptor for Land Use Classification", *In Proc. of TELFOR 2014*, pp.889 - 892.

[11] A. Vedaldi and A. Zisserman, "Efficient additive kernels via explicit feature maps", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.34, no.3, pp. 480 – 492, 2012.