Urban Area Segmentation Using Visual Words

Lior Weizman Jacob Goldberger

***Abstract*—In this paper we address the problem of urban areas extraction by using a feature-free image representation concept known as "Visual Words". This method is based on building a "dictionary" of small patches, some of which appear mainly in urban areas. The proposed algorithm is based on a new pixel-level variant of visual words and is based of three parts: building a visual dictionary, learning urban words from labeled images, and detecting urban regions in a new image. Using normalized patches makes the method more robust to changes in illumination during acquisition time. The improved performance of the method is demonstrated on real satellite images from three different sensors: LANDSAT, SPOT and IKONOS. To assess the robustness of our method, the learning and testing procedures were carried out on different and independent images.**

Keywords - object detection, segmentation, remote sensing, urban areas, map updating, visual words.

I. Introduction

In the last few years, urban zone detection from satellite sensor imagery has become crucial for several applications. The main one is Geographic Information Systems (GIS) update, which enables efficient study and planning of urban growth, a continual need. GIS data can also help government agencies and other policy makers make decisions about regional issues. In most cases, humans are not a satisfactory resource to handle the enormous number of satellite images acquired for urban detection. Therefore, it is essential to have efficient tools for automatic detection and segmentation of urban areas. Because of the unique texture of urban scenes with respect to natural scenes, the main approaches for segmentation of urban zones are based on texture analysis. Texture operators are either gray-level-based or structure-based. Gray-level-based texture operators are based on a co-occurrence matrix (GLCM) [1], or a normalized gray-level histogram [2] and common structure-based operators are be Gabor wavelet [3] and the gradient-based feature [4]. A recent work by Zhong and Wang [5] combines low and high levels of structure-based texture for urban detection. Some approaches use spectral data in the image to improve the detection rate e.g. [6]. Urban areas can also be extracted by classification of the entire image [7] or by neural network based methods [8]. Although very different in approach, all the currently used methods for urban detection suffer from a major drawback - the absence of robustness.

This paper presents a new approach to the task of urban detection and segmentation which we dub *Visual Word Region Detection* (VWRD). The method is based on the "Visual Words" paradigm which is a recently introduced concept that has been successfully applied to scenery image classification tasks (see e.g. [9], [10]). The visual words model is based on the idea that it is possible to transform the image into a set of visual words and to represent the image (and objects within the image) using the statistics of the occurrence of each word as feature vectors. These visual words are image patches (small sub images) that are clustered to form a dictionary consisting of a small set of representative patches. We apply a pixel-level variant of this approach to urban area extraction, by adapting it to meet the demands of urban segmentation.

II. Bag of Visual Words

In this study we show that a highly successful text retrieval approach, known as "Bag-of-Words" (BoW), can be used for detecting urban areas in satellite images. The BoW model is a simplifying assumption used in natural language processing and information retrieval. A text (such as a sentence or a document) is represented as an unordered collection of words, disregarding grammar and even word order. We only retain information on the number of occurrences of each word. For example, "a big house" and "house big a" are the same in this model. The BoW model can be used for a dictionary-based modeling. A document is represented by a vector where each entry of the vector refers to the count of the corresponding entry in the dictionary. An excellent introduction to the BoW concept and its applications can be found in [11].

To represent an image using the BoW model, an image has to be treated as a document. This means we need to define a visual analogy for a word and a visual analogy for a code-book or a dictionary that contains a list of all possible words. However, a "word" in images is not an off-the-shelf entity like word in text documents. There is no natural visual analog to the concepts of a word and a dictionary. Hence, to apply the BoW approach to image analysis tasks, we first need to define a visual analogy for word and dictionary. This is usually done in a three-step procedure: feature detection, feature description and codebook generation. The visual word model is thus an image histogram representation based on independent local features.

Given an image, feature detection is used to extract several local patches (or regions), which are considered as candidates for basic elements, or "words". Taking every pixel (or pixels on a regular grid) is probably the most simple yet effective method for feature detection. Other approaches that are based on interest point detectors, try to detect salient patches, such as edges and corners. Following the feature detection step, feature representation methods deal with how to represent the patches using feature descriptors. In the next section we describe our descriptor based on PCA applied to the patch pixel values. A popular alternative approach to patches is the SIFT representation [12] which can be beneficial is scenery

L. Weizman is with the School of Engineering, Bar-Ilan University, Ramat-Gan 52900, Israel (e-mail: weizmanl@gmail.com).

J. Goldberger is with the School of Engineering, Bar-Ilan University, Ramat-Gan 52900, Israel (e-mail: goldbej@eng.biu.ac.il).

images. The final step of the visual bag of word model is to convert vector represented patches into visual "words" which also produces a "dictionary". A visual word can be considered as a representative of several similar patches. A simple method is performing K-means clustering over all the vectors to form the words.

Current applications of visual words are image classification, clustering and retrieval in areas of large image datasets, video data and medical image datasets. These tasks are all based on a single BoW representation for the entire image. A more refined task is object (or event) detection where an object is treated as a sub-image. Urban-zone detection is different in the sense that we are not looking for urban objects. Instead, we want to detect urban zones at a single pixel resolution. Hence every point is interesting and we have to compute a feature vector for every pixel in the image. In the task of urban-zone detection, it is meaningless to represent the objects we want to detect as a frequency of occurrence word histogram since each pixel corresponds to a single word. Instead, we introduce a pixel-level variant of the visual words concept. in the training step we use labeled data to build visual word histograms for urban and non-urban areas. These histogram models are used at the test step to detect pixel-based resolution urban zones.

III. The Urban Detection Algorithm

The first step of our urban detection system is to compile a dictionary of visual words. This step forms a bridge from the image processing world to the world of text processing. In the next step we create visual word histograms for urban and non-urban areas. This yields a set of "urban words". These words occur much more frequently in urban areas and detection of such words is a strong indication of the presence of an urban region. Given a new unlabeled test image we look for visual-words that correspond to urban detection words as a first step for detecting urban areas. A postprocessing step applies spatial consistency constraints on the detected urban patches to obtain a global decision on urban regions. The Visual Word Region Detection (VWRD) algorithm is composed of the following parts: compiling a visual dictionary, learning urban words from labeled images and detecting urban regions in a new image. Following is a detailed description of the urban detection algorithm.

A. *Compiling a dictionary*

The task of compiling a visual dictionary is the process of creating a vocabulary of words that will be further used to represent primitives in the image. To develop a comprehensive dictionary, one or more images with urban and non-urban areas are required. The first step towards obtaining visual words is extracting local features from the images. We represent each image as a collection of spatially adjacent pixels (patches) which are treated collectively as a single primitive. We view patches of size $n \times n$ as one-dimensional vectors of size n^2. To increase the robustness of the algorithm and to avoid the need for atmospheric/radiometric calibration, each vector is first normalized. Combinations of two kinds of normalization can be considered such as subtracting the vector mean and dividing by the vector standard deviation. Generally, the normalization process is expected to reduce the differentiation capabilities between urban and non-urban zones, while increasing the robustness of the algorithm to different acquisition conditions. The decision regarding the optimal normalization depends on the trade-off between the informativeness and the robustness and is data driven. In our approach, the normalization step is based on subtracting the patch mean. This makes the features invariant to gray-level scale differences between images. We further explore this point in the experiment section.

Taking into consideration the ground sampling distance (GSD) of commercial imaging satellites, a spatial patch size which is smaller than 11×11 pixels does not contain enough information for the task of urban zones detection. A 11×11 patch is large enough to preserve urban elements such as straight lines and corners and is not too large so that there are other patches similar to that patch. This claim is supported by experimental results in the next section.

To reduce both the algorithm's computational complexity and the level of noise, a feature extraction method is applied. Generally, urban zones, in contrast to non-urban zones, are characterized by high spatial frequencies. Therefore, we apply a principal component analysis procedure (PCA) to reduce the dimensionality of the data. We expect that the first components of the PCA (which are the components with the highest variance in the image) will contain the information about the spatial frequencies of the patch, and therefore, will differentiate urban zones from non-urban zones.

The main step in the dictionary building procedure is clustering the patches to form a small-size dictionary of visual words. A common clustering algorithm, such as iterative self organizing data analysis (ISODATA) [13] or K-means can be used for this purpose. This yields data vectors in the projected space that are clustered into M groups. Finally, the mean vector of every group is computed to create a dictionary with M visual words. Note that this dictionary development step is done in an unsupervised mode without any reference to the urban/non-urban label of each patch.

B. *Urban words learning phase*

Based on labeled images, urban and non-urban areas are statistically modeled as frequency occurrence histograms of the dictionary words, and the relevant words from the dictionary that best differentiate urban areas from non-urban areas are found. First, urban and non urban areas are defined on the training image. Each area is then divided into patches, a mean normalization is carried out on every patch following by the linear PCA transformation (that was computed in the previous step) and assignment of the patch to the nearest dictionary word (using the Euclidian distance). We obtain two word frequency histograms, one for the urban zone and one for the non-urban zone. Normalizing the histograms we can view them as discrete distributions $P_{\text{urban}}(\cdot)$ and $P_{\text{non-urban}}(\cdot)$ of the visual words in urban and non-urban areas respectively.

Our goal is to find the words in the dictionary whose use in urban areas is significantly higher than in non-urban areas. Therefore, given an arbitrary patch, the probability of this

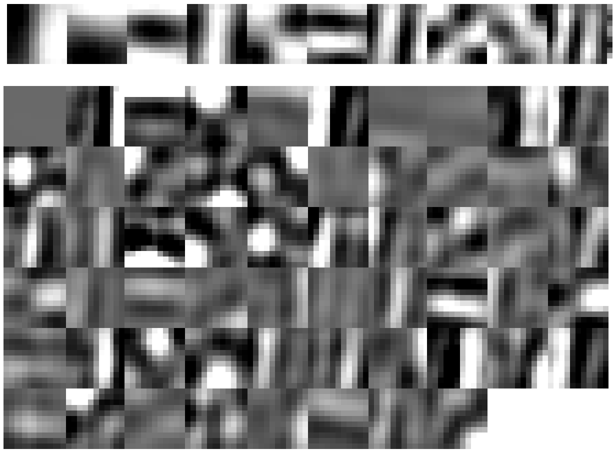

Fig. 1. (top) The 10 PCA eigenvectors that were used to reduce the data dimensionality. (bottom) The dictionary of 58 words, words are ordered from left to right, one row after another.

patch being taken from an urban region can be computed using Bayes' rule:

$$P(\text{urban}|u) = \frac{\alpha P_{\text{urban}}(u)}{\alpha P_{\text{urban}}(u) + (1-\alpha)P_{\text{non-urban}}(u)} \quad (1)$$

where α is the prior probability of a patch to be in an urban area region. The words from the dictionary that best differentiate urban areas from non-urban area are the words that $P(\text{urban}|u) \geq threshold$ whereas the $threshold$ is a tunable parameter. Thus we obtain a group of "urban words" that characterize urban patches. Detection of such words is a strong indication of an urban area.

C. Urban detection in a new image

Given a new image, we want to detect and segment the urban regions. Each one of the image patches on a regular grid is translated into one of the visual words from the dictionary. This is done by first normalizing the patch vector, applying the PCA transformation that was learned in the training step. Then, every transformed vector word is assigned to its nearest word from the dictionary (based on the Euclidian distance). Utilizing equation (1), we can compute the posterior probability for each patch to be in an urban region. The result is a local urban/non-urban decision for each separate patch. One of weaknesses of the visual words concept is that it ignores the spatial relationships among the patches, which is crucial in image representation. A standard way to incorporate spatial consistency is the Markov Random Field (MRF) model. We can view the urban/non-urban labels of each patch as a grid of hidden binary random variables and the urban/non-urban histograms can be seen as distributions of the observed patches conditioned on the binary hidden label. The global image urban labeling can be then obtained using standard MRF optimization algorithms.

We took a simpler approach that avoids the need for MRF optimization algorithms with high computational complexity. As is explained above, patches that correspond to words which are above the urban threshold are detected as patches in urban areas. It was empirically found that this decision is very reliable and therefore these patches can be used as anchor points for a global decision. To remove outliers and to obtain a global smooth decision on urban areas, a post-processing morphological operator is applied on the local urban-decision map. Using a majority voting analysis, to replace "holes" in the urban detection areas with their surrounding values, is sufficient to achieve reliable global smooth results.

TABLE I
DATABASE SUMMARY

Sensor	Resolution	#scenes	# train images	# test images
LANDSAT 7	30m	2	2	49
SPOT 5	10m	7	4	74
IKONOS	4m	5	6	49

IV. EXPERIMENTAL RESULTS

This section presents the results of the proposed method when applied to real satellite images. A total number of 14 different scenes from three different sensors were used in our experiments. The scenes characteristic were as follows: The IKONOS scenes mostly contain a dense urban areas and agricultural fields. The SPOT scenes mostly contain plane agricultural fields and small villages. The Landsat scenes consist of mountainous areas and small villages. These scenes were divided into 184 sub-images with spatial dimensions of 640×640 pixels each. The scenes were divided into train scenes and test scenes. The training sub-images were used for compiling the dictionary and for the urban-words learning processes, and the reminder of the images were used to evaluate algorithm performance. This separation was done in order to check the algorithm robustness to changes in scene. We used the training images from all the three sensors to build a single visual dictionary. The images have different resolutions which contributes multi-resolution abilities to the produced image dictionary. A detailed description of out dataset is given in Table I.

In our implementation, the training images were divided into patches of size 11×11 each. We then normalized every patch by subtracting the patch mean. The process was followed by a dimensionality reduction step to reduce the data to a dimension of 10. The 10 eigenvectors (which can be also viewed as patches) that were used to reduce the dimensionality of the data are presented in Fig. 1. The next step in the dictionary compilation process was to cluster the reduced data into M groups. It is important to select a number of words that provides proper quantization of the data and yet doesn't overfit it. We have found that for our task a dictionary size of 60 provides a good balance. Fig. 1 shows the words in the dictionary.

In this phase, the relevant words that best differentiated urban areas from non urban areas were found. First, urban and non-urban areas were defined on the training images. Then, the frequencies of every word in the dictionary in the urban and non-urban areas were computed. The gray level difference among non-urban patches is eliminated during the patch mean

TABLE II
DETECTING URBAN ELEMENTS, VWRD VS. GLCM RESULTS

	Sensor	# urban	#detected elements		PD pixels (%)		FAR (%)	
		elements	VWRD	GLCM	VWRD	GLCM	VWRD	GLCM
FINAL RESULTS	LANDSAT 7	29	28	29	71	71	0.8	38.1
	SPOT 5	82	82	81	75	80	4.7	13.4
	IKONOS	431	416	401	79	68	0.04	16.5
	TOTAL	542	526	511	75	74	2.2	21
BEFORE POSTPROCESSING	LANDSAT 7	29	27	29	58	72	0.8	44.3
	SPOT 5	82	82	81	61	80	3	20.6
	IKONOS	431	412	424	58	69	0.04	9.6
	TOTAL	542	519	534	60	75	1.1	24

normalization step. On the other hand, urban areas, that are characterized by high variability, are modeled by the majority of the words in the dictionary. The next step in the learning process is to find the words that have the highest posterior probability to be part of an urban scene, according to the probabilistic model defined in the previous section. We defined the prior probabilities as 0.5 each. The posterior probability of every word in the dictionary to be part of an urban scene, was computed according to Eq. (1).

The final step in the learning process is defining the "urban words" set, the words whose posterior probability to be part of an urban scene is above a certain threshold. We set the threshold to be $threshold = 0.95$. (we chose the higher threshold as possible in order to decrease the FAR). The outcome was 36 words in the "urban words" set. The indices of these words in the dictionary presented in Fig. 1 are: $4, 11-15, 17, 20-22, 24-28, 30-32, 34-40, 42, 43, 45-47, 52$ and $54-58$. It can clearly be seen that most of the words included in the "urban words" set exhibit morphological features that mostly characterize urban scenes (e.g. edges, corners), whereas most of the words that are not included in the "urban words" (e.g. 1, 7 and 8) do not include these features.

In order to obtain a decision map of urban pixels in a new image, we used a moving window of 11×11 pixels. The operations that were carried out on each window were as follows. First, the same pre-processing as on the train images was applied to assign a visual word from the dictionary to the window. Finally, an "urban" decision was made for the central pixel of a window if the word that was assigned to the window was included in the "urban words" set. By dragging the window pixel by pixel over the image, a full decision map of urban pixels was obtained. After construction of the entire decision map for the image, a morphological operator was then applied to the classified image to remove outliers and to impose smoothness. We used a majority vote analysis with a kernel size that was 5 times larger than the patch spatial size, to fill the "holes" in the urban detection results.

To quantify the results of urban areas detection in the test images, urban areas were defined in the test images to create a ground truth images. These ground truth images were obtained manually by an experienced image analyst. The smaller urban area that was defined included 20 pixels. A total number of 542 urban areas were defined on the ground truth images. We considered an urban area as detected by the algorithm if at least 50% of its pixels were labeled as urban pixels by the algorithm. Two quality measures were used. The probability of detection (PD) is the number of urban elements that were detected in the test image divided by the total number of urban elements in the images (542). The false alarm rate (FAR) is the number of pixels the were falsely detected as urban areas, divided by the total number of pixels in the images. Using our method, 526 out of 542 urban zones were detected, (PD=97%) while the FAR was 2.2%. An example of the urban detection results for one of the IKONOS test images is given in Fig. 2. Several pre-processing steps for patch normalization can be used. We found that normalization by dividing each patch by its standard deviation decreases the amount of information in the patches to a level where urban and non-urban zones cannot be differentiated. None of the words from the dictionary passed the selected threshold. As a result, the "urban words" set was empty, leading to $PD = 0$ in these cases. In addition, we found that totally disabling the normalization process, lessens the robustness of the method and decreases the results. The performance results with patch mean subtraction were 96.5% and without were 89.6%. To summarize, we used mean subtracting and did not use standard-deviation normalization. Regarding the optimal selection of the number of words in the dictionary.

To test the performances of our algorithm, we compared our method to the method based on a gray-level co-occurrence matrix (GLCM) to detect urban areas [1]. We used exactly the same set of training and testing images as appears in Table I. We used the Bayes classification rule of equation (1) to find urban pixels using the extracted GLCM features. Then, we applied the same post-processing operator that was applied to the classified VWRD results. A detailed summary of the results of the GLCM algorithm vs. our algorithm is given in Table II. It can be seen that the PD of our algorithm is slightly higher than the PD of the GLCM method (97% vs. 94 %), but our method also provides a much lower FA ratio (2.2% vs. 21%). We also show a pixel-level comparative results with and without a morphological post-processing step. It can be seen from Table II that the post-processing step actually performed a "completion" operation to the initial VWRD results: it led to an increase in both PD and FAR measures in the final VWRD result. However, the same post-processing operation

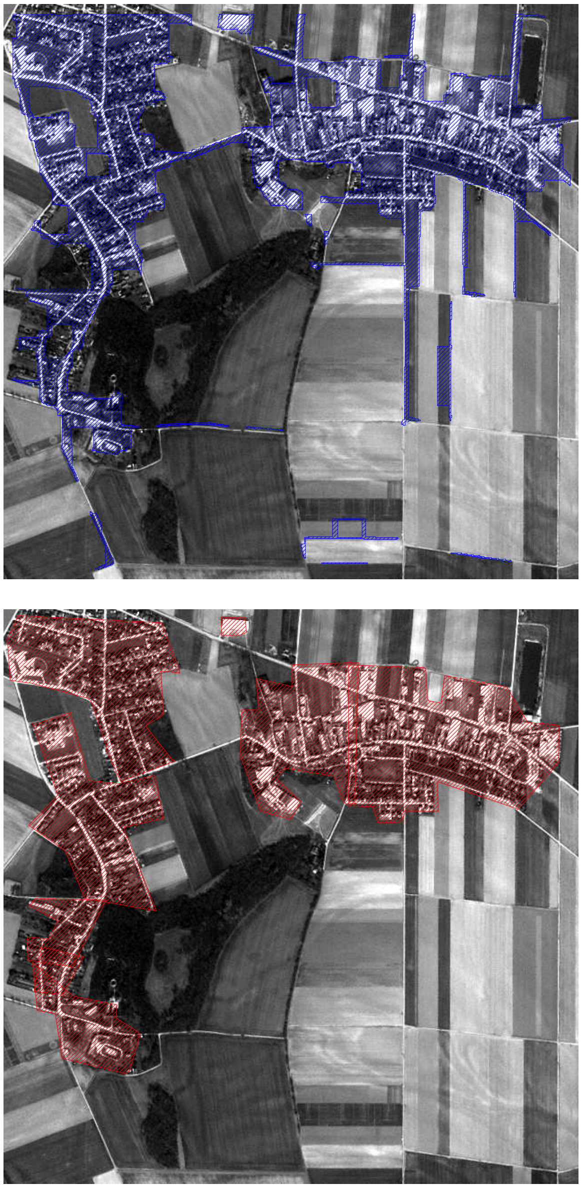

Fig. 2. Detection results (top) vs. ground truth (bottom) of urban areas in one of the test images (IKONOS).

actually removed outliers in the initial GLCM result: it led to a decrease in both PD the FAR values in the final GLCM results. To demonstrate the major improvement of our method as compared over the GLCM method in terms of FA ratio, Fig. 3 presents the false detected pixels for both methods when applied to one of the test images. This figure shows that most of the false detected pixels in the GLCM algorithm belong to borders areas (i.e. roads, borders between agricultural sites, etc.) whereas our method mostly overcome this type of false detection.

To conclude, we proposed a method to learn and recognize urban areas in satellite images based on a new pixel-based variant of the visual word concept. Our method has an advantage over other methods for urban area detection, since it is not constrained to extract a predefined set of features, and it can be robust to changes in scene and to illumination effects. In addition, we believe the VWRD method can also be successfully used for other detection or classifications purposes of remotely sensed data.

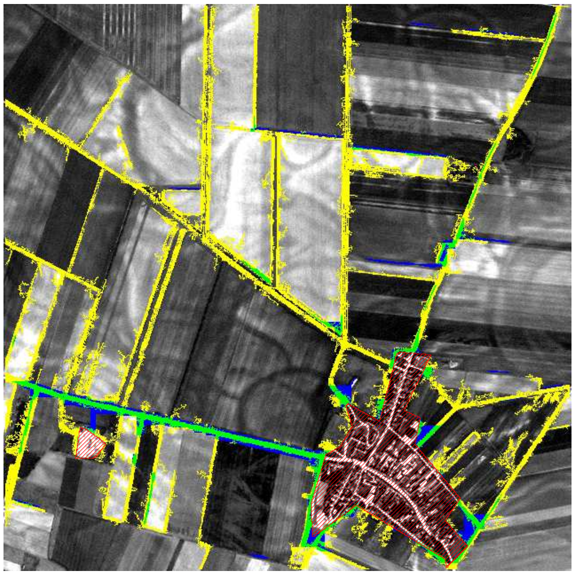

Fig. 3. False detected pixels for the GLCM (yellow), for the VWRD (blue). Common false detected pixels are marked in green and the ground truth areas are marked in red.

References

[1] P. C. Smits and A. Annoni,“Updating land-cover maps by using texture information from very high-resolution space-borne imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 37, pp. 1244–54, May 1999.

[2] A. K. Shackelford and C. H. Davis,“A hierarchical fuzzy classification approach for high-resolution multispectral data over urban areas,” *IEEE Trans. Geosci. Remote Sens.*, vol. 41, pp. 1920–32, Sep. 2003.

[3] J. Li and R. M. Narayanan, “Integrated spectral and spatial information mining in remote sensing imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 42, pp. 673–685, Mar. 2004.

[4] S. Yu, M. Berthod, and G. Giraudon, “Toward robust analysis of satellite images using map information-Application to urban area detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 37, pp. 1925–39, Jul. 1999.

[5] P. Zhong and R. Wang, “Using combination of statistical models and multilevel structural information for detecting urban areas from a single gray-level image,” *IEEE Trans. Geosci. Remote Sens.*, vol. 45, pp. 1469–82, May 2007.

[6] G. Rellier, X. Descombes, F. Falzon, and J. Zerubia, “Texture feature analysis using a Gauss-Markov model in hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 42, pp. 1543–51, Jul. 2004.

[7] F. Lafarge, X. Descombes, and J. Zerubia, “Textural kernel for SVM classification in remote sensing: application to forest fire detection and urban area extraction,” *IEEE Int. Conf. on Image Processing*, vol. 3, pp. 1096–99, 2005.

[8] S. Berberoglu, C. D. Lloyd, P. M. Atkinson, and P. J. Curran, “The integration of spectral and textural information using neural networks for land cover mapping in the Mediterranean,” *Computers & Geosciences*, vol. 26, pp. 385–396, 2000.

[9] R. Fergus, P. Perona, and A. Zisserman, “Object class recognition by unsupervised scale-invariant learning,” *IEEE Computer Vision and Pattern Recognition*, vol. 2, pp. 264–271, 2003.

[10] L. Fei-Fei and P. Perona, “A Bayesian hierarchical model for learning natural scene categories,” *IEEE Computer Vision and Pattern Recognition*, vol. 2, pp. 524–531, 2005.

[11] C. D. Manning, P. Raghavan, and H. Schutze, Introduction to information retrieval, Cambridge University Press, 2008.

[12] D. Lowe. “Distinctive image features from scale-invariant keypoints,” *Int. Journal of Computer Vision*, pp. 91–110, 2004.

[13] J. T. Tou and R. C. Gonzalez, Pattern recognition principles, Addison-Wesley, 1977.