Texture analysis for images with forested areas

Andreea Concea
Faculty of Automatic Control and Computer Science
University Politehnica of Bucharest, Bucharest, Romania
concea.andreea@yahoo.com

Tudor Nitescu
Faculty of Automatic Control and Computer Science
University Politehnica of Bucharest, Bucharest, Romania
tudor.alin97@yahoo.com

Dan Popescu
Faculty of Automatic Control and Computer Science
University Politehnica of Bucharest, Bucharest, Romania
dan_popescu_2002@yahoo.com

***Abstract*— The information on the structure of the forest has been of high interest for a long time, being related to the possibilities of maximizing the production of the forest in the conditions of ensuring ecological stability, but also of detecting and preventing illegal deforestation. The main objective of this paper is the texture analysis of forested areas from geographical images. To achieve this goal, texture features will be used such as: fractal dimension, lacunarity, LBP histogram, mean intensity and contrast.**

Keywords— Forested areas, Texture features, Texture analysis

I. INTRODUCTION

In order to protect the environment and to prevent and combat illegal deforestation, as well as to detect affected areas of forests, techniques such as texture and fractal analysis are used. In this regard, there are a lot of studies, which apply the methods mentioned above on geographical images with forests.

The main reason why the texture analysis is suitable for forested areas is because a forest can contain areas with different textures (deforested areas, road areas, richly forested areas, sparsely forested areas).

Texture is a rich source of visual information and it can be a key element in the image analysis domain. It is known to provide clues about the depth of the scene, the orientation of the surface, describing the content of the images. Texture analysis is a technique for extracting information related to it, being considered a type of processing for digital images used to obtain quantitative measures of information that can be helpful for classification or segmentation.

Texture is a feature that can describe all surfaces, such as: wood, fabrics, the pattern of crops in a field, forests, rivers, vegetation, areas with roads, etc. It contains relevant information about the structure of the surfaces and their placement in the environment. Although there are a multitude of studies on texture analysis, it is quite difficult to precisely define and analyze texture by digital means [1].

Requirements for the protection of forests and nature necessarily involve the acquisition of extensive knowledge of the structure of the forest, through qualitative and quantitative parameters. To this end, the texture analysis folds very well on this subject, as it provides the necessary information related to the structure of forested areas.

II. RELATED WORK

Texture analysis is suitable for our forested images because these type of images are very rich in the texture field and not in colour as others. Our images contain forested areas and land areas which are very similar in colour, but are different in texture.

The same topic was approached by a team of researchers from Australia, in which they analyzed the geographical images from UAVs using two methods: pixel-based image classification and a convolutional neural network (CNN) machine learning algorithm. In the first method, they defined seven classes and used a maximum likelihood parametric rule on the red, green and blue channels of the images. They wanted to recognize elements from the following classes: rock, water, Kikuyu Grass, Coastal Morning Glory, Mirror Plant, Lomandra and other native/dead vegetation. The accuracy that they obtained on this approach ranged from 74% to 85%, for different island area images. For the second method, where a CNN was used, they defined 4 steps: input image segmentation, train detector, detector implementation and region classification. For the training dataset, they used 1194 positive and 174.423 negative examples. Each region of interest was represented by a 64×64 pixel window. The detection exercise took 600s to run on an Intel Core i7-3820 CPU, and the performance indicators for this method were: 82% TP rate, 18% FN rate, 0.05 FP rate and 99.95% TN rate. In order to get to this point, they applied non-maximum suppression (NMS) to merge the multiple detected windows containing Lomandra which had significant overlapping [2].

Another study, made in China, used texture features to classify coastal wetland vegetation from high resolution images, using completed local binary patterns (CLBP) and Haralick features. They wanted to separate the following types of vegetation into different classes: water, ground, Phragmites communis (PC), Spartina alterniflora (SA), Suaeda glauca community (SG), mixed vegetation of Phragmites communis and Suaeda glauca (PC & SG), mixed vegetation of Spartina alterniflora and Suaeda glauca (SA & SG). In order to separate the classes, they used the Jeffries-Matusita distance, which they found to be more suitable for expressing the separability characteristics of categories than other separation indicators. The value of this distance being between 0 and 2, the larger the value, the better the separability obtained. They obtained accuracies between 83% and 86%, with better performances obtained for CLBP texture features [3].

III. METHODS AND MATERIALS

In this section we describe the method used for segmentation of forested areas from UAV images and also detail the texture and fractal analysis techniques which were used in our classification process.

A. *Materials*

The images used for training and testing were taken from a fixed-wing UAV MUROS [4]. The main characteristics, flight requirements and performances can be found below.

TABLE I. UAV – CHARACTERISTICS AND TECHNICAL SPECIFICATIONS

Characteristics	**Technical Specifications**
UAV	Electric propulsion, fixed wing, automatic
Payload	Gyro-stabilized mechanism, retractable
Camera photo	Sony Nex7, objective 50mm, 24.3 megapixels, 10 fps
Flight high	200 m – 300 m
Speed	70 km/h – 100 km/h
Autonomy	120 min, 15 km
Type of flight	Photogrammetric
Image acquisition	Memory card

Figure 1. UAV characteristics

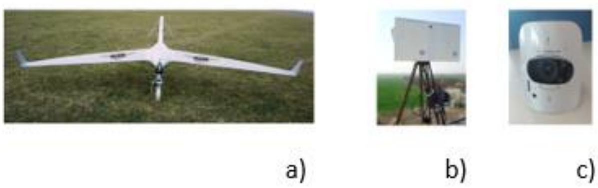


Figure 2. System components a) UAV fixed wings b) Ground terminal c) Payload photo

We used 100 images, each image having the resolution of 4000 x 4000 pixels. Each image has been splitted into 400 patches, each having 200 x 200 pixels, and each patch was separated into the three color channels, RGB. In total, there were 40000 patches used in our process.

The images were processed on a computer with the following specifications: Intel Core i5-8300H, 16 GB RAM and 512 GB SSD.

B. *Methods*

B.1. Fractal features

Fractal analysis evaluates the fractal characteristics of images. It consists of several methods of assigning a fractal dimension and other fractal characteristics to an image

Among the techniques used in fractal analysis, the most popular are the following ones: box counting for fractal dimension, lacunarity analysis, mass analysis, multifractal analysis, etc.

i. Fractal dimension

For the analysis of the fractal dimension, a 3D grid is formed with boxes of size:

$$r \times r \times s \tag{1}$$

where r is the division factor (the side length of the square in the grid), and s is the height of each stack of squares (or box).

s represents a value calculated according to the formula:

$$s = \frac{r \times Imax}{M} \tag{2}$$

where I_{max} represents the maximum value of the intensity from the image, and M × M represents the image resolution.

Each square, $S_r(u, v)$ of size $r \times r$ at a point (u, v) in the grid corresponds to a stack of squares with the height s. Each pixel (i, j) in the square has an intensity value (a gray level) that belongs to a square in the stack. The maximum value of the intensity in a square is denoted by p (u, v), and the minimum value of the intensity by q (u, v).

The algorithm calculates this value s in the current iteration, for each value of r. The logic chosen in the algorithm was to initialize r with M/2 and then divide this r by 2, as much as the image resolution allows. At each iteration, having the calculated value s and the current value r, the grid of squares (after u and v) was traversed, up to the value of M / r (maximum number of squares in the grid by rows and columns). The value of p (u, v) was initialized with 0 and q (u, v) with 255, to later determine their values.

Furthermore, the algorithm always compares the intensity values of the current pixel I (i, j) with the minimum and maximum value. If this value is greater than the current maximum value, then the maximum value becomes I (i, j). Similarly, if this value is less than the current minimum value, then the minimum value will become I (i, j).

After finally going through all the pixels of a square, the variables max and min will have the maximum intensity, respectively the minimum intensity, from the square Sr (u, v). Having these two values, we can calculate further:

$$n_r = \max - \min + 1 \tag{3}$$

In the end, after crossing through the grid for the current value r, we calculate the value of N_r as the sum of all the numbers calculated for each square Sr (u, v) of the grid.

The algorithm will return, after an iteration, the value Nr, which we later associated with the corresponding r value for which it was calculated, representing a point (log 1/r, log N_r) on the graph for which it will be calculated, by linear regression, slope (which represents the differential fractal size for images with gray levels) [5].

Also, within this described algorithm, the lacunarity will be calculated, for each value of r, and the method used will be described in the next section.

ii. Lacunarity

The lacunarity property was introduced in the fractal analysis of images to improve the description for the content of fractal objects.

Intuitively, if the structure of a fractal has gaps or holes, then it is considered to have a high degree of lacunarity. In other words, a high value of this feature means that the pixels are grouped, generating a wide variety of islands or gaps. A high value of the lacunarity indicates a strong heterogeneous character of the texture, while a low value of this property is generally associated with homogeneous images.

The lacunarity property can be used to compare structures that have the same fractal size, but different distributions of the gaps. The greater the gap, the greater the variation in pixel distribution in an image. In other words, a high lacunarity means that the pixels are grouped in a wide variety of island sizes, surrounded by an extremely varied void, which indicates the heterogeneity of the spatial pattern or texture.

The calculation of the lacunarity values was done on each differential box-counting iteration, using the following formula:

$$L(r) = \frac{\sum_N N^2 \times P(N,r)}{\left[\sum_N N \times P(N,r)\right]^2} \tag{4}$$

where P(N,r) represents the aparition probability of the $n_r(u,v)$ value over the image and N represents the N_r value computed during box-counting iterations [6].

B.2. Other texture features

In texture analysis, techniques such as the LBP histogram (low binary pattern), the HOG histogram (histogram of oriented gradients), or other texture classifiers (mean intensity, contrast, image gradient, entropy etc.) are used to classify the images elements. Some of these methods will be described below.

i. LBP histogram

LBP is a simple but efficient operator that assigns to each pixel of the image a binary coded value, by analyzing the properties of the pixels in its neighborhood. The operator can thus be seen as an approach that combines two different traditional methods, namely the statistical method and the structural method.

The first variant of the algorithm, proposed by Ojala and collaborators, uses 3 × 3 pixel neighborhoods in the image with gray levels. The values of the pixels in this neighborhood are binarized according to the value of the central pixel, multiplied by the powers of 2 according to the position in the neighborhood and added together to obtain a value assigned to the central pixel.

7	5	2
6	6	1
8	6	7

a)

1	0	0
1		0
1	1	1

b)

1	2	4
128		8
64	32	16

c)

Figure 3. LBP Computation Example a) Neighborhood example b) After binarization c) Weight

$$Pattern = 11110001 \tag{5}$$

$$LBP = 1 + 16 + 32 + 56 + 128 = 241 \tag{6}$$

By replacing the central pixels of the neighborhoods with the LBP values, a new image is obtained, which we call the LBP image. After computing the values in the LBP image, the LBP histogram is obtained (often group methods are used), which can be used in the texture classification process [7].

In our implementation, we grouped the 256 values obtained for the LBP histogram in separate intervals, each having 25 elements, with the last interval containing 31 elements.

ii. Mean intensity

A pixel has an intensity value associated between 0 and 255, represented on 8 bits. The value associated with black is 0, while the value associated with white is 255. Thus, the mean intensity value is computed by calculating the arithmetic average of all the pixels within an image.

iii. Contrast

Each image has a maximum(max_i) and a minimum(min_i) value for intensity. In our project, we used the Michelson contrast [8], which is defined as in the following formula:

$$\frac{max_i - min_i}{max_i + min_i} \tag{7}$$

C. Implementation

We implemented, using the Python language, an algorithm which computes the features described above and classifies the images given as input. We chose Python because it is suitable for image processing, offering a lot of libraries useful in this regard. The diagram of the algorithm implemented can be found below:

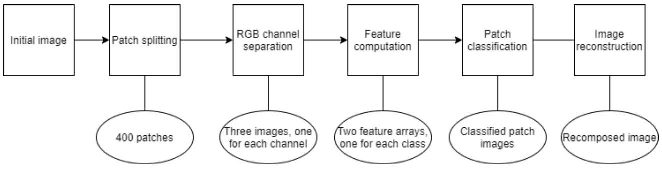


Figure 4. Algorithm diagram

D. *Training*

In our training process we used 10 images. For each image, we chose 10 patches from forest class, and 10 patches from field class with a total of 200 patches. For these, we calculated the features described in sections B.1 and B.2 for each channel, and we noticed better performances for lacunarity on blue channel, mean intensity and contrast (the later two on all RGB channels). For fractal dimension and LBP histogram, we observed that the values were overlapping in most of the cases, so the classes could not be distinguished using these two features. So, we decided to use the average values for mean intensity(RGB), contrast(RGB) and lacunarity(B), after analysing the values calculated and comparing the results obtained with this approach to the results that we got using all the features described above. The reference values for the two classes, obtained for one of the training images can be found in the table below:

TABLE II. TRAINING REFERENCE VALUES

Feature	**Class**	**Channel**		
		R	**G**	**B**
Fractal Dimension	Forest	2.50	2.49	2.49
	Land	2.49	2.49	2.49
Lacunarity	Forest	1.08	1.08	1.08
	Land	1.07	1.07	1.05
Mean Intensity	Forest	83.72	107.19	118.19
	Land	195.03	199.48	159.88
Contrast	Forest	0.92	0.78	0.72
	Land	0.52	0.44	0.53

Figure 5. Training reference values for an image

We can observe visible differences between the values obtained for some features of the two classes, while some of them are almost identic (fractal dimension, for example). We decided to ignore the values which were overlapping, so we went further with mean intensity(RGB), contrast(RGB) and lacunarity on blue channel, for a total of 7 classifiers.

The LBP histograms obtained on the RGB channels, for the same image given as an example above, can be found below:

TABLE III. LBP HISTOGRAM VALUES EXAMPLE

Channel	**Class**	
	Forest	**Land**
Red	[0.195, 0.085, 0.102, 0.019, 0.102, 0.090, 0.017, 0.093, 0.036, 0.260]	[0.196, 0.091, 0.112, 0.017, 0.104, 0.100, 0.017, 0.098, 0.033, 0.234]
Green	[0.194, 0.085, 0.103, 0.019, 0.102, 0.090, 0.016, 0.092, 0.035, 0.263]	[0.196, 0.089, 0.109, 0.016, 0.104, 0.102, 0.017, 0.098, 0.033, 0.237]
Blue	[0.198, 0.085, 0.101, 0.019, 0.097, 0.090, 0.018, 0.095, 0.036, 0.262]	[0.194, 0.082, 0.104, 0.017, 0.102, 0.102, 0.017, 0.101, 0.034, 0.244]

Figure 6. LBP histogram values example

A notable observation we made was the fact that the features values for the two classes (forest and land) must not overlap in order to consider that feature eligible in our classification process. Notable differences should be observed between the values for a classifier on the two classes, in order for that to be eligible for being used. Thus, the classifier must distinguish the classes as much as possible with its values on at least one channel,

E. *Testing*

For the testing part, we computed the set of the 7 values we chose, for 200 patches, different than the ones used in the training phase. Using the Minkowski I minimum distance (16) between the patch feature array and the two classes feature arrays (land and forest), each feature component had a value which was closer to the value from one of the two classes corresponding feature, so each feature classified the patch in one of the two classes. Having a set of 7 classifiers, each specifying one of the two classes, we chose the majority of the features classifying votes as the patch class. The patches that were classified as land were colored in a red tone, while the others were left as they were, and after the algorithm finished the execution for all the patches, it also reconstructed the initial images, using the new patches.

$$D(X,Y) = |X - Y| \tag{8}$$

For performance evaluation, we compared the expected results with the actual results of over 200 randomly chosen patches. We computed the number of true positives, true negatives, false positives and false negatives in order to determine the accuracy and precision.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{11}$$

$$Specificity = \frac{TN}{TN + FP} \tag{12}$$

Where:
TP (true positive) – correctly classified forested areas;
TN (true negative) – correctly classified land areas;
FP (false positive) –wrongly classified forested areas;
FN (false negative) – wrongly classified land areas.

TABLE IV. PERFORMANCE EVALUATION

Indicator	Result
TP	123
TN	58
FP	17
FN	2
Total number of patches	200
Accuracy	90.50%
Precision	87.85%
Sensitivity	95.00%
Specifity	77.33%
Execution time for one image	170 seconds

Figure 7. Performance indicators

IV. EXPERIMENTAL RESULTS

In this section we will present the experimental results obtained for a few images involved in our processing, as well as some of the patches used in our approach.

TABLE V. TEST LAND PATCHES RESULTS

Patch	Forest votes	Land votes	Expected result	Actual result
	0	7	land	land
	0	7	land	land
	4	3	land	forest
	1	6	land	land
	0	7	land	land
	0	7	land	land
	0	7	land	land
	3	4	land	land
	0	7	land	land
	1	6	land	land

Figure 8. Testing land patches

TABLE VI. TEST FOREST PATCHES RESULTS

Patch	Forest votes	Land votes	Expected result	Actual result
	6	1	forest	forest
	6	1	forest	forest
	5	2	forest	forest

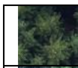
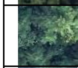
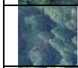
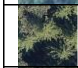
	7	0	forest	forest
	6	1	forest	forest
	7	0	forest	forest
	6	1	forest	forest
	7	0	forest	forest
	7	0	forest	forest
	7	0	forest	forest

Figure 9. Testing forest patches

In the tables from above, we presented a few patches from different images that were classified by the vote of the seven features that we used. We can observe that in some cases, not all the classifiers provided the correct result. One of the patches from the first table, which we expected to be classified as land was actually "voted" as forest, but we can observe that the number of votes was balanced between the two classes. Most of the errors we had in our classification were related with land patches which were misclassified.

In the following tables we attached some of the images that we used for the processing, as well as the results for the reconstructed images. The land classified areas were colored with red and the forested areas were left untouched.

TABLE VII. INITIAL IMAGES

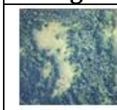
Image 1	Image 2	Image 3	Image 4	Image 5
				

Figure 10. Image samples that were used for texture analysis

TABLE VIII. IMAGES RESULTED FOR ALL FEATURES

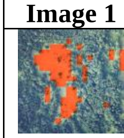
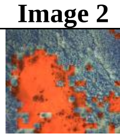
Image 1	Image 2	Image 3	Image 4	Image 5
				

Figure 11. The recomposed images using all the features described in this paper: fractal dimension, lacunarity, LBP histogram, mean intensity, contrast (red, green and blue channels)

TABLE IX. IMAGES RESULTED FOR THE CHOSEN FEATURES

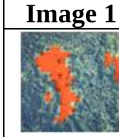
Image 1	Image 2	Image 3	Image 4	Image 5
				

Figure 12. The recomposed sample images using the chosen features: r,g,b mean intensity, b lacunarity and r,g,b contrast

Comparing the tables VIII And IX, we can observe a notable improvement in the classification process for the images in the second one mentioned, where we didn't include fractal dimension, LBP histogram and lacunarity on red and green channels.

In the first paper presented in section II, where a group of scientists wanted to recognize the elements from seven classes, the accuracy that they obtained on the pixel-based image classification ranged from 74% to 85% for different images. In our method, we obtained an accuracy of 90.5%, a better result compared with the one mentioned earlier, but a percentage obtained for the classification of only two classes. The authors of the second paper described in the section II obtained an average accuracy of approx. 85%, but they tried to classify more classes as well. So, we obtained better performances, but for a lower amount of classes.

After recomposing the initial images with the classified patches, we computed the forest and land area percentages, by counting the number of patches with land and computing its percentage from the total number of patches (400). The percentage of forest patches was computed by subtracting from 100 the calculated land percentage. The results can be found in the following table:

TABLE X. FOREST AND LAND AREA

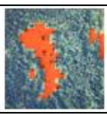
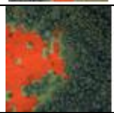
No.	Image	Field Patches Number	Forest Patches Number	Forest Area	Land Area
1		68	332	83%	17%
2		117	283	70.75%	29.25%
3		120	280	70%	30%
4		106	294	73.5%	26.5%
5		22	378	94.5%	5.5%

Figure 13. The results obtained for the classified forested and land areas

Based on the results obtained above, considering we had good performance indicators, we can say that these percentages can offer a good hint related to deforested areas from processing the UAV images of different geographical areas.

CONCLUSION

We observed that mean intensity and contrast were good classifiers for texture analysis of UAV forest images. The values obtained for these two features were good for clearly distinguishing between classes, although in some cases there were some errors.

The execution time of the algorithm that we implemented was good, considering that it processed an image with 400 patches in the average time of 170 seconds, meaning that it processed more than two patches per second.

In our implementation, we obtained good performance indicators, pointing out the fact that our method is efficient, but also that there is room for further improvement. For example, the field of parallel processing can be approached in order to improve execution times.

In conclusion, the chosen features were suitable in our needs for classifying the forested areas from UAV images, obtaining good accuracy and precision indicators, but the addition of neural networks can be favorable in regard of obtaining even better performance indicators.

ACKNOWLEDGMENT

REFERENCES

[1] L. Armi, S. Fekri-Ershad, „Texture image analysis and texture classification methods", International Online Journal of Image Processing and Pattern Recognition, Vol. 2, No.1, pp. 1-29, 2019.

[2] S.M. Hamylton, R.H. Morris, R.C. Carvalho, "Evaluating techniques for mapping island vegetation from unmanned aerial vehicle (UAV) images: Pixel classification, visual interpretation and machine learning approaches", International Journal of Applied Earth Observation and Geoinformation, Volume 89, July 2020.

[3] M. Wang, X. Fei, Y. Zhang, Z. Chen, X. Wang, "Assessing Texture Features to Classify Coastal Wetland Vegetation from High Spatial Resolution Imagery Using Completed Local Binary Patterns (CLBP)", Remote Sensing, 10, 778, 2018.

[4] EUROPA, Multisensory robotic system for aerial monitoring of critical infrastructure systems. Available online: https://trimis.ec.europa.eu/ (accessed on 19 January 2020).

[5] Sarker, N., Chaudhuri, "An efficient differential box-counting approach to compute fractal dimension of image", IEEE Transactions on Systems, Man, and Cybernetics, vol. 24, pp. 115-120, 1994.

[6] Karperien, A., Jelinek, H., Milosevic, N. T. (2011). Reviewing lacunarity analysis and classification of microglia in neuroscience. In European Congress of Mathematics and Theoretical Biology: Interdisciplinary Approaches to Fractal Analysis (IAFA) (pp. 888-893), 2011.

[7] A. Porebski, N. Vandenbroucke, D. Hamad, "LBP histogram selection for supervised color texture classification", published in IEEE International Conference on Image Processing, 2013.

[8] H. Kukkonen, J. Rovamo, K. Tiippana, R. Näsänen, "Michelson contrast, RMS contrast and energy of various spatial stimuli at threshold", Vision Res. Vol. l-1. No. 10. pp. l43l 1436, 1993.