Carriage road pursuit based on statistical and fractal analysis of the texture

Dan Popescu, and Radu Dobrescu

***Abstract*—** The images from a camera unit which is placed on an intelligent vehicle are divided in equivalent small squared regions by matrix partition. In order to pursue the carriage road, the reference region (an asphalt one) is compared with all others image regions. Then, the textured region similarity is measured and the regions with a different texture are detected and localized. Some types of statistical texture feature are analyzed: features extracted from co-occurrence matrix, edge density per unit of area, and grey level histogram of the difference image. On the other hand, we investigate how the fractal dimension is used for textured image classifications. In order to measure the similarity of a textured region pair, the fractal dimensions of both regions are computed by the box-counting algorithm utilizing. It is proposed two new features from the estimation of the fractal dimension: the mean fractal dimension and the effective fractal dimension.

For the proper region identification and classification, we introduced the notion of average co-occurrence matrix, which is quasi-invariant to image rotation and translation. Texture recognition is based on decision theoretic method. The algorithm is implemented in Visual C++ and MATLB and allows the simultaneously display of the investigated region, the Euclidian distance between them and a reference image region, and the segmentation map. The basic texture (reference) is considered an asphalt one and the different textures are considered the grass and the pebble. The result is the classification of the tested texture in road and non-road type. Based on the classification algorithm, a segmentation process of road images is accomplished. The segmentation finesse depends on the image resolution and the texture finesse.

Our experimental results indicate the fact that the selected features have a good discriminating power.

***Keywords*—**Average co-occurrence matrix, texture classification and segmentation, fractal dimension, box-counting algorithm.

I. Introduction

For the purpose of carriage road following, the images from the assistant camera unit are divided in equivalent small squared regions (subimages). The asphalt texture is considered as a reference texture. Our goal is to identify the asphalt regions and to produce an asphalt localization matrix by recognition techniques. One can observe easily that the road and its neighbourhoods are characterized by textures. Image texture, defined as a function of spatial variation in pixel intensity (grey values), is useful in a variety of applications and has been a subject of intense study by many researchers. The texture can be considered like a structure which is composed by many similar elements (patterns) named textons or texels, in some regular or continual relationship. Thus, texture can be considered as an attribute of a field having no components that appear enumerable. It can be useful to segment images into regions of interest and to classify those regions. Texture relates to the surface or structure of an object and depends on the relation of contiguous elements. Wilson [3] points out that textured regions are spatially extended patterns based on more or less accurate repetition of some unit cell; the origin of the term is related with the craft of weaving. Gonzalez [1] relates texture to other concepts like smoothness, fineness, coarseness, graininess and describes the three different approaches for texture analysis: statistical, structural and spectral.

There are two important kinds of problems that texture analysis research attempts to solve: texture segmentation and texture classification. The process called texture segmentation consists in identifying regions with similar texture and separating regions with different texture. Texture classification involves deciding what texture class an observed image belongs to. Thus, one needs to have a priori knowledge of the classes to be recognized.

Mathematical procedures to characterize texture fall into four general categories, statistical, geometrical, model-based methods and signal processing methods [8]. Because texture has many different dimensions and characteristics there is no single method of texture representation that is everywhere adequate. The different methods include grey level histograms, co-occurrence matrices, spatial autocorrelation functions,

Fractals and spatial second order moments are two spatial analytical techniques used to measure geometric complexity [6] and conveniently describe many irregular, fragmented patterns found in nature. Thus, the fractal based texture classification is another approach that correlates texture coarseness and fractal dimension. A fractal is defined [5] as a set for which Hausdorff-Besicovich dimension is strictly greater than the topological dimension.

There are two points of interest in this paper. The first point is the road analysis for moving objectives based on statistical features (especially derived from the average co-occurrence matrix). The case study consists in the asphalt region

Manuscript received April 3, 2008. This work was supported in part by the Romanian Ministry of Education, Research and Youth, under the Grant 11-040-1007. Revised received June 30, 2008.

The authors are with the POLITEHNICA University of Bucharest, Faculty of Control and Computers, Splaiul Independentei 313, Bucharest 7100, Romania (telephone: +40214029105, e-mail: dan_popescu_2002@yahoo.com).

separation (carriage road) in the acquired image from video camera. In fact, one does not need to know which characteristic textures exist in the image in order to do texture segmentation. The second point of interest is the analysis and classifications of textural images based on fractal dimension with box-counting algorithm. We provide a case study in more detail that illustrates how an estimation of the fractal dimension derived from box counting algorithm can be utilized in texture similarity measurement for asphalt road tracking.

The configuration of a system for mobile robot, navigation based on fractal and statistical type texture features is presented in Fig. 1.

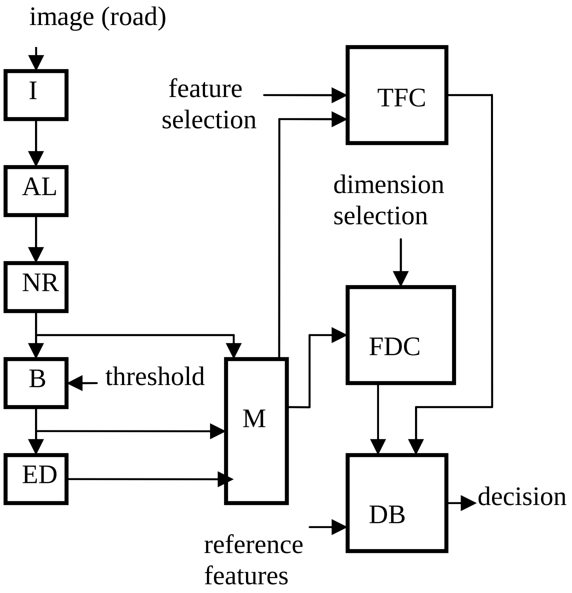


Fig.1. Configuration of system based on digital texture processing for mobile robot navigation.

The significances of notation in Fig.1 are the following:

I – interface; AL – auto-level, stretching of the minimum and maximum grey levels of the image to the maximum interval (0-255); NR – noise rejection; B – segmentation block; ED – edge detection block; M – memory for different pre-processed images (grey level image after noise rejection operation, segmented images with manifold thresholds, contour images); TFC – texture features computation; FDC – computation of fractal dimensions and some derived parameters; DB – decision block.

Texture segmentation implies the prior knowledge of the texture types and numbers which exist in the analyzed image. This is a more difficult problem than the texture classification.

II. Statistical feature based method for texture classification and segmentation

The most powerful statistical method for texture similarity evaluation is based on features extracted from the Grey-Level Co-occurrence Matrix (GLCM), proposed by Haralick in 1973 [8]. GLCM is a second order statistical measure of image variation and it gives the joint probability of occurrence of grey levels of two pixels separated spatially by a fixed vector distance $d = (\Delta x, \Delta y)$. Smooth texture gives co-occurrence matrix with high values along diagonals for small d. The range of grey level values within a given image determines the dimensions of a co-occurrence matrix. Thus, four bits grey level images give 16x16 co-occurrence matrices. The elements of a co-occurrence matrix C_d (1) depend upon the displacement $d=(\Delta x, \Delta y)$.

$$C_d (i,j) = Card\{((x,y),(t,v))/I(x,y) = i, I(t,v) = j, \\ (x,y), (t,v) \in N \, x \, N, (t,v) = (x+ \Delta x, y+ \Delta y)\} \quad (1)$$

From a co-occurrence matrix C_d one can draw out some important statistical features for texture classification. These features, which have a good discriminating power, were proposed by Haralick: contrast, energy, entropy, homogeneity, and variance.

For each pixel we can consider $(2d+1)$x$(2d+1)$ symmetric neighborhoods, d = 1, 2, 3,...,15. Inside each neighborhood there are 8 principal directions: 1, 2, 3, 4, 5, 6, 7, 8 (Fig. 2) and we evaluated the co-occurrence matrices $C_{d,k}$ corresponding to vector distances determined by the central point and the neighborhood edge point in the k direction (k = 1,2,...,8). With a view to obtain statistical feature insensitive relatively to texture rotate, we introduce the average co-occurrence matrix notion.

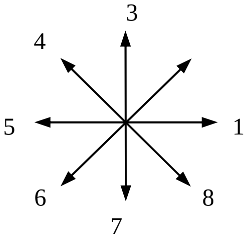


Fig.2. The principal directions for co-occurrence matrix calculus

For each neighborhood type, we define an *average co-occurrence matrix C_d* which is calculated by the average of the eight co-occurrence matrices (2).

$$C_d = 1/8(C_{d,1} + C_{d,2} + C_{d,3} + C_{d,4} + C_{d,5} + C_{d,6} + \\ + C_{d,7} + C_{d,8}) \ , d = 1,2,...,15 \quad (2)$$

Thus, in the case of 3x3 neighborhood, d = 1; for 5x5 neighborhood, d = 2, and so on.

In order to quantify the spatial dependence of gray level values, from average co-occurrence matrices C_d, we calculate various textural features like Contrast – Con_d – (3), Energy – Ene_d – (4), Entropy – Ent_d – (5), Homogeneity – Omo_d – (6) and Variance – Var_d – (7).

$$Con_d = \sum_{i=1}^{L} \sum_{j=1}^{L} (i - j)^2 C_d (i, j) \quad (3)$$

$$Ene_d = \sum_{i=1}^{L}\sum_{j=1}^{L} C_d(i,j)^2 \tag{4}$$

$$Ent_d = -\sum_{i=1}^{L}\sum_{j=1}^{L} C_d(i,j)\log(C_d(i,j)) \tag{5}$$

$$Omo_d = \sum_{i=1}^{L}\sum_{j=1}^{L} \frac{C_d(i,j)}{1+|i-j|} \tag{6}$$

$$Var_d = \frac{1}{L}\sum_{i=1}^{L}\sum_{j=1}^{L} [C_d(i,j) - \overline{C_d(i,j)}]^2 \tag{7}$$

In the preceding relations, *LxL* represents the dimension of co-occurrence matrices.

For the purpose of texture similarity evaluation it can be calculated the Euclidian distance between regions with similar texture like $D\{I_1(1),I_1(5)\}$, and the Euclidian distance between regions with different textures like: $D\{I_1(1),I_1(3)\}$, $D\{I_1(1),I_1(4)\}$, $D\{I_1(1),I_2(3)\}$, $D\{I_1(1),I_2(4)\}$. The Euclidian distance $D\{I_1, I_2\}$ between two images I_1 and I_2, which are characterized by the feature vectors $[C_1,E_1,Et_1,O_1,V_1]^T$ and $[C_2,E_2,Et_2,O_2,V_2]^T$, is expressed by the following relation:

$$D(I_1,I_2) = \sqrt{(C_1-C_2)^2+(E_1-E_2)^2+(Et_1-Et_2)^2+(O_1-O_2)^2+(V_1-V_2)^2}$$

where: $C = Con$, $E = Ene$, $Et = Ent$, $O = Omo$, $V = Var$.

Another simple statistic features is the edge density per unit of area, Den_e (8). The density of edges, detected by a local binary edge detector, can be used to distinguish between fine and coarse texture, like in Fig.2. Den_e can be evaluated by the ratio between the pixel number of extracted edges (which must be tinned – one pixel thickness) and image area:

$$Den_e = \frac{N_e}{A} \tag{8}$$

In (8), N_e represents the number of edge pixels (tinned edges, with one pixel thickness) and *A* is the region area (pixel number of image region).

The regions having different textures are considered to be segmented regions. There are two general approaches to performing texture segmentation: region-based approach and boundary-based approach. We utilized the first approach, because we tried to identify regions of the image which have a uniform texture. Small local regions are compared based on the similarity of some statistical texture property. The regions with mixed texture are considered like non-asphalt ones. Thus, the regions with different textures are always separated by the aid of a threshold which is fixed by error consideration. This method has the advantage that the boundaries of the asphalt regions are always well defined. Based on the efficiency analysis of the tested features, we selected the five features derived from the average co-occurrence matrices for similarity measurement of the image regions. The segmentation finesse depends on the partition degree of the initial images. On the other hand, if the partition degree is too high, then it is possible that the texture should disappear. In fact, for an application, the partition index is established tacking into account the given image resolution and the texture finesse. For example, if the asphalt image has 2048x2048 pixels, we can consider 1024 squared regions (32x32 image partition). One of the region, for instance the upper – left region, is considered like the reference one, Ir = I(1). After image partitioning, a block matrix (mxm) is obtained. Each block is a textured region with a well defined position. The reference region is compared with the other regions, successively, by means of Euclidian distances D(Ir, I(j)), j = 2,3,...,mxm. We can consider the texture features derived from the average co-occurrence matrix C_d. First, the comparison result is binarized by means of a corresponding threshold. The region I(j) is indexed 0, if $D(Ir, I(j)) \geq T$ (the region is similar with Ir), and is indexed 1, if $D(Ir, I(j)) < T$ (the region is similar with Ir). Then the regions are filled with 0 or 1 depending on the region index.

III. Fractal dimension as texture feature

Fractal analysis is a mathematical and computer technique that quantifies complex shapes. Fractal analysis can discriminate between the shapes of texture of roads. Thus, a method to relieve the irregularity of the road is to calculate and combine different forms of fractal dimension.

Fractals have high power in low frequencies, which enables them to model processes with long periodicities. Many natural surfaces have a statistical quality of roughness and self-similarity at different scales. Fractals are very useful and became popular in modeling these properties in image processing. We first define a deterministic fractal in order to introduce some of the fundamental concepts. Self-similarity across scales in fractal geometry is a crucial concept. A deterministic fractal is defined using this concept of self-similarity as follows. Given a bounded set *A* in a Euclidean n-space, the set *A* is said to be self-similar when *A* is the union of *N* distinct (non-overlapping) copies of itself, each of which has been scaled down by a ratio of *r*. The fractal dimension *D* is related to the number *N* and the ratio *r* as follows (9):

$$D = \frac{\log N(r)}{\log(1/r)} \tag{9}$$

The fractal dimension gives a measure of the roughness of a surface. Keller [19] uses fractal geometry to describe texture. Intuitively, the larger the fractal dimension, the rougher the texture is. Most natural surfaces and in particular textured surfaces are not deterministic as described above but have a statistical variation. This makes the computation of fractal dimension more difficult.

There are a number of methods proposed for estimating the fractal dimension *D*. One method is box-counting algorithm that assumes determination of fractal dimension in function of the evolution of the object size in connection with the scale factor. There are many programs for counting the fractal dimension in different forms, but the most familiar algorithm is the box counting. A lot of specialized papers describe this algorithm. For the box counting basic algorithm, the image must be binary type. The method consists in dividing the image, successively, in 4, 16, 64 etc. (1/*r*) same size squares and computing every time the number *N(r)* of squares covered by the object image, where r is the step size. The dividing process is limited by the image resolution. The fractal dimension can be obtained plotting *lnN(r)* for different values of *ln(1/r)*, where r is the side length of covering boxes and calculating the slope of the resulting curve, which is approximated by a line. A linear regression is performed using the logarithmic coordinates (10). The regression slope *a* is used to determine the box counting fractal dimension *FD* (11).

$$y = a\,x + b \tag{10}$$

The notation significances in equation (3) are the following: x_i = $\log_2$ (1/r_i), y_i = $\log_2$ ($N(r_i)$), n – number of partitions, i = 1,2,3,…,n – the function points in the graphical representation.

$$a = FD \;=\; \frac{n\displaystyle\sum_{i=1}^{n} x_i y_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{n\displaystyle\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2} \tag{11}$$

Towards evaluating the fractal dimension of a grey level image, we applied the box-counting algorithm to contours extracted from the binary images which are obtained by different thresholds. Because the binary image (and also the fractal dimension) depends on the threshold, we used in our algorithm all the significant grey levels contained in the image. The fractal dimensions computed for every grey level will be represented into a graphic named fractal dimension spectrum. Thus, the algorithm calculates a mean fractal dimension *MFD* from individual values *FD_j* of some image contours detected from initial grey-level image (12).

$$MFD = \frac{1}{k}\sum_{j=1}^{k} FD_j \tag{12}$$

The algorithm, which was implemented in MATLAB, consists of the following steps:

1) Reading and converting of the color image in 256 grey levels;

2) Converting of the 256 grey levels image to a binary level image using a fixed threshold T_j;

3) Extraction of the image contour using 3x3 neighborhoods;

4) Computing of the fractal dimension FD_j, from the contour image, applying the box-counting algorithm;

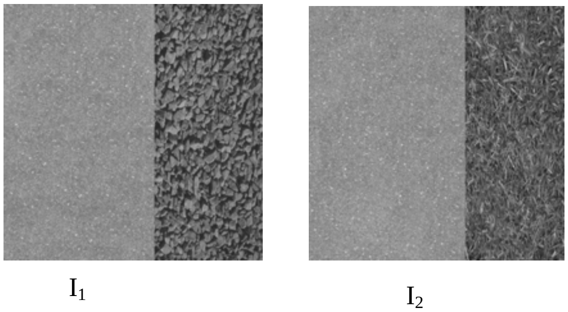
5) Iteration of the steps 1- 4, for j = 1,…,*k*;

6) Determination of *MDF* from equation (11).

IV. THE FIRST CASE STUDY: CARRIAGE ROAD IDENTIFICATION AND SEGMENTATION BASED ON REGION SIMILARITY

With this end in view, for study, the whole image is partitioned in sixteen equivalent regions. Different textured regions (Fig.3) are compared based on minimum distance between measured features which are derived from medium co-occurrence matrices (contrast, energy, entropy, homogeneity, and variance). Image region I_1(1), which contains only asphalt texture is considered the reference texture template. If a region contains another texture or mixed textures, then it is considered a defect region.

For algorithm testing and program validation we have used two textured images I_1 and I_2 (Fig. 3), each partitioned in sixteen regions I_i(1), I_i(2),..., I_i(16), i = 1,2 (Fig.4).



I_1

I_2

Fig. 3. Analyzed images I_1 and $\mathrm{I}_{2.}$

In fact, it was considered regions with 128 x 128 pixels, and 16 grey levels. From these images we choosed five regions for I_1 image: I_1(1) – reference texture, asphalt; I_1(5) – tested region, asphalt; I_1(4) – tested region, pebble; I_1(8) – tested region, pebble I_1(3) – tested region, asphalt and pebble, and also three regions for I_2 image: I_2(4) – tested region, grass; I_2(8) – tested region, grass; I_2(3) – tested region, asphalt and grass (Fig. 5).

I(1)	I(2)	I(3)	I(4)
I(5)	I(6)	I(7)	I(8)
I(9)	I(10)	I(11)	I(12)
I(13)	I(14)	I(15)	I(16)

Fig.4. Sixteen regions image partition.

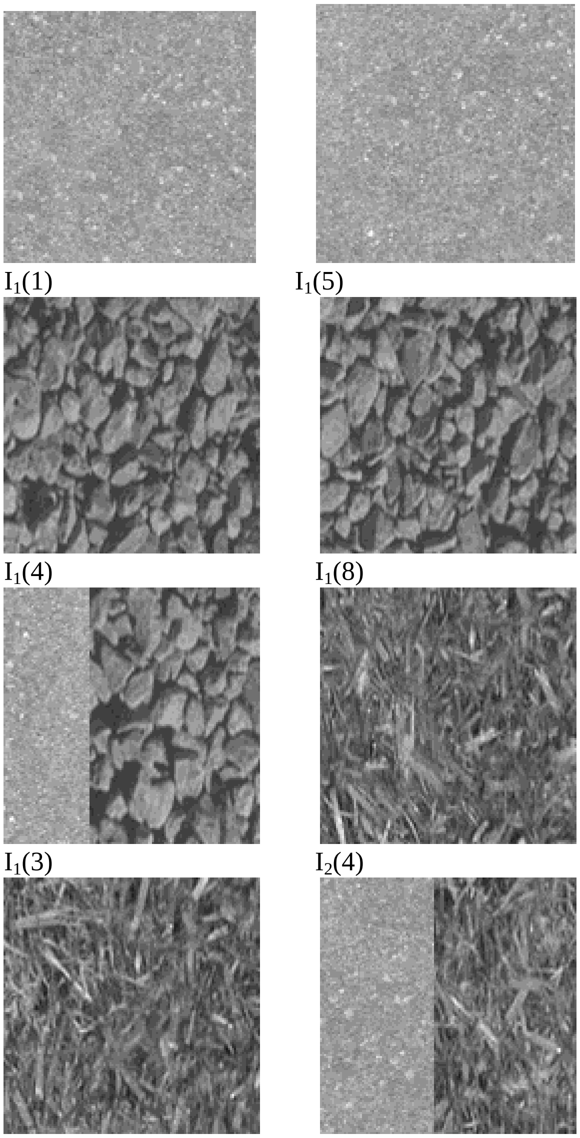


Fig. 5. Selected regions from I_1 and I_2.

Textural features like Con_d, Ene_d, Ent_d, Omo_d, and Var_d are calculated from average co-occurrence matrices, for different distances d. The normalized results are presented in Table 1 (I_1) and Table 2 (I_2), for d = 10.

Table 1. Normalized statistical texture features for I_1 and d = 10.

Region Index	Ent	Ene	Con	Omo	Var
1	1.00	1.00	1.00	1.00	1.00
3	0.85	0.20	4.84	0.73	0.48
4	0.85	0.17	5.31	0.65	0.48
5	1.00	0.98	1.00	1.00	1.01
8	0.85	0.17	5.25	0.66	0.48

The normalized characteristics are necessary for efficient calculation of the Euclidian distance, because the ranges of initial characteristics can differ too much. The normalization is refered to the refference texture (region $I_1(1)$).

Table 2. Normalized statistical features for I_2 and d = 10

Region Index	Ent	Ene	Con	Omo	Var
3	0.93	0.31	2.81	0.90	0.71
4	0.94	0.26	5.72	0.81	0.46
8	0.93	0.27	4.00	0.78	0.46

The results of the Euclidian distance calculus between template $I_1(1)$ and mentioned different regions distances are the following:

$D\{I_1(1),I_1(5)\} = 0.02$;
$D\{I_1(1),I_1(3)\} = 3.96$; $D\{I_1(1),I_1(4)\} = 4.43$;
$D\{I_1(1),I_1(8)\} = 4.37$; $D\{I_1(1),I_2(3)\} = 1.96$;
$D\{I_1(1),I_2(4)\} = 4.81$; $D\{I_1(1),I_2(8)\} = 4.11$

One can observe that the distances between two different regions, like $D\{I_1(1),I_1(4)\}$, $D\{I_1(1),I_1(8)\}$, $D\{I_1(1),I_1(3)\}$, $D\{I_1(1),I_2(4)\}$, $D\{I_1(1),I_2(8)\}$, *and* $D\{I_1(1),I_2(3)\}$, are greater than distances between two similar regions, like $D\{I_1(1),I_1(5)\}$. In order to appreciate the efficiency of the presented algorithm, we analyzed the most unfavorable cases, namely the minimum distance between two regions coming from different textures, and the maximum distance between two regions coming from the same texture. Thus, minimum value for dissimilar textures,

$$min\{D\{I_1(1),I_1(3),D\{I_1(1),I_1(4)\},D\{I_1(1),I_2(4)\}, D\{I_1(1),I_2(3)\},....\} = 1.82,$$

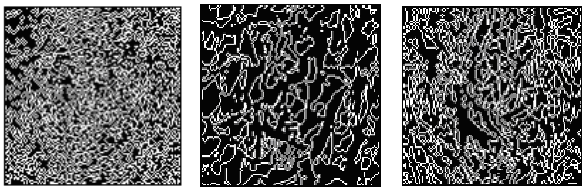
is grater than maximum value for similar textures,

$$max\{D\{I_1(1),I_1(2)\}, D\{I_1(1),I_1(5)\},...\} = 0.10,$$

in large neighborhood case (d = 5,10,15).

Also, we can observe that, in this case study, the most important features, with greater discriminating power, both in texture similarity evaluation and in defect region detection and identification, are the contrast and the energy.

Towards ameliorate the classification accuracy, a development of the recognition algorithm, consisting in the attachment of new textural features like edge point density per unit of area is analyzed. Thus, we considered an edge extraction algorithm, based on binary image and logical function [11], which gives tinned edges (Fig. 6).



$I_1(5)$ $I_1(4)$ $I_2(4)$

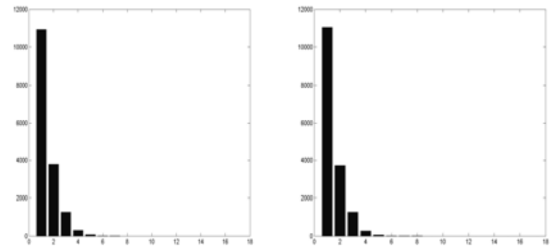
Fig. 6. Contour image for some image regions

Table 3. Normalized edge densities for some regions

Region	*Den$_e$*
I_1 (1)	1.000
I_1 (5)	0.995
I_1 (4)	0.782
I_1 (3)	0.789
I_2 (4)	0.845
I_2 (3)	0.793

Unfortunately, for the analyzed regions *$I_1(5)$, $I_1(3)$, $I_1(4)$, $I_2(3)$,* and *$I_2(4)$*, the edge densities show that this feature has not a good discriminating power (Table 3) and the combination with the previously second order type statistical features would give better results in texture classification. Another disadvantage of this algorithm is the dependence of the results by the threshold level for edge extraction.

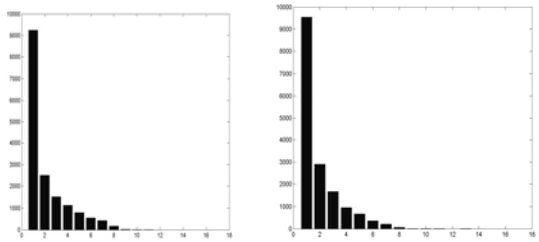
Also, we tested the efficiency of the grey level image difference histogram in texture classification and segmentation. With that end in view we have considered the same images $I_1(1)$, $I_1(5)$, $I_1(3)$, $I_2(3)$. The image difference histograms in the displacement (x = 10, y = 10) for $I_1(1)$, $I_1(5)$ are presented in Fig. 7, and the image difference histograms for $I_1(3)$, $I_2(3)$ are presented in Fig. 8.



$I_1(1)$ $I_1(5)$

Fig.7. Grey level difference histogram for $I_1(1)$ and $I_1(5)$.

The basic aspect of the histogram is similar in the presence of a texture defect (mixed texture), like $I_1(3)$ and $I_2(3)$, but is dissimilar for different texture, like $I_1(1)$ and $I_1(3)$. For this reason, the image difference histogram can be utilized efficiently in texture classification by the minimum distance criterion between the corresponding level vectors.



$I_1(3)$ $I_2(3)$

Fig.8. Grey level difference histogram for $I_1(3)$ and $I_2(3)$.

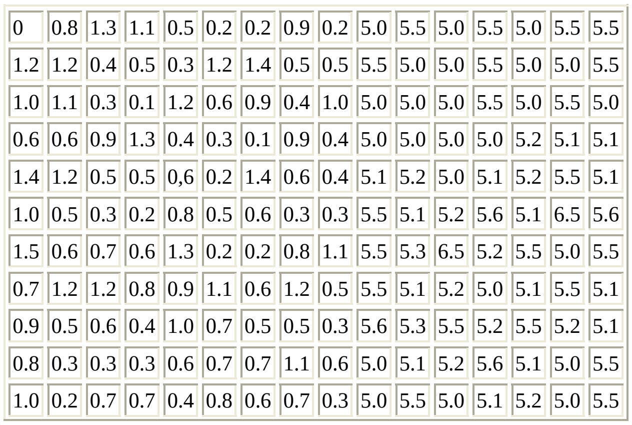
In graphical histogram representation, the value for gray level 0 is too high and irrelevant comparing with the others. Therefore it is neglected. One can observe that the difference image histogram has a good behavior referring to texture classification in the low level domain of the histogram.

The second case study is related to a segmentation method of a road image, based on multiple comparisons of the textured regions. With this end in view, the whole image is partitioned in 256 equivalent regions (16x16 block matrix).

The blocks are 32x32 matrices. The rounded values of the region similarity measurement are also placed in a 16x16 matrix (Fig.9). After the binarization process, the 16x16 block matrix becomes the segmentation matrix (Fig.10), with 1 for carriage road (asphalt) and 0 for the rest. In Fig 10, **1** represents a 32x32 matrix filled with 1 logic, and **0** represents a 32x32 matrix filled with 0 logic.

V. The second case study: carriadge road identification based on fractal dimension estimation

For our case study, we used three 512x512 grey level type images with different texture: asphalt, broken stones, and grass. The images were divided in four equivalent parts (regions). Towards algorithm testing, we selected two different regions from each single texture image (like asphalt, $I_1(1)$, $I_1(5)$, broken stones, $I_1(4)$, $I_1(8)$, and grass, $I_2(4)$, $I_2(8)$) and one region from each mixed image (like asphalt-broken stones, $I_1(3)$, and asphalt-grass, $I_2(3)$ – Fig. 5. For each region, with 256x256 pixels, one can compute the fractal dimension FD_j, by the box-counting algorithm, up to the pixel level.



0	0.8	1.3	1.1	0.5	0.2	0.2	0.9	0.2	5.0	5.5	5.0	5.5	5.0	5.5	5.5
1.2	1.2	0.4	0.5	0.3	1.2	1.4	0.5	0.5	5.5	5.0	5.0	5.5	5.0	5.0	5.5
1.0	1.1	0.3	0.1	1.2	0.6	0.9	0.4	1.0	5.0	5.0	5.0	5.5	5.0	5.5	5.0
0.6	0.6	0.9	1.3	0.4	0.3	0.1	0.9	0.4	5.0	5.0	5.0	5.0	5.2	5.1	5.1
1.4	1.2	0.5	0.5	0,6	0.2	1.4	0.6	0.4	5.1	5.2	5.0	5.1	5.2	5.5	5.1
1.0	0.5	0.3	0.2	0.8	0.5	0.6	0.3	0.3	5.5	5.1	5.2	5.6	5.1	6.5	5.6
1.5	0.6	0.7	0.6	1.3	0.2	0.2	0.8	1.1	5.5	5.3	6.5	5.2	5.5	5.0	5.5
0.7	1.2	1.2	0.8	0.9	1.1	0.6	1.2	0.5	5.5	5.1	5.2	5.0	5.1	5.5	5.1
0.9	0.5	0.6	0.4	1.0	0.7	0.5	0.5	0.3	5.6	5.3	5.5	5.2	5.5	5.2	5.1
0.8	0.3	0.3	0.3	0.6	0.7	0.7	1.1	0.6	5.0	5.1	5.2	5.6	5.1	5.0	5.5
1.0	0.2	0.7	0.7	0.4	0.8	0.6	0.7	0.3	5.0	5.5	5.0	5.1	5.2	5.0	5.5

0.3	0.5	0.8	0.2	0.3	0.7	0.3	0.8	1.1	4.9	5.1	5.2	5.2	5.1	5.2	5.5
1.4	0.4	0.9	0.3	0.1	1.0	1.4	0.5	0.4	5.0	5.1	5.3	5.1	5.6	4.9	5.2
0.6	0.9	0.8	0.9	0.7	0.3	0.8	0.4	0.8	5.2	6.6	5.2	5.5	5.1	5.5	5.6
0.6	0.1	0.8	0.9	0.1	0.6	0.5	0.9	0.7	5.0	5.1	5.5	5.2	5.1	5.5	6.5
0.2	1.0	0.8	0.6	0.2	0.5	0.8	0.8	0.1	4.9	5.2	5.1	5.5	5.6	5.4	5.6

Fig. 9. Similarity index matrix.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
3	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
4	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
5	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
6	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
7	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
8	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
9	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
10	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
11	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
12	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
13	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
14	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
15	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
16	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0

Fig. 10. Segmentation block matrix.

For every binary threshold, we extract the image edges and we calculate the fractal dimension. For the binary thresholds, an interval was chosen, based on the request that a definite texture exist in the contour image (Fig.11). The texture edges are extracted by a local logical operator. We represent the fractal dimension spectrum for contour image and we compute the mean fractal dimension *MDF*. The slope of the log-log curve is evaluated by the linear regression method. The existence, inside an image, of textured regions with different fractal dimension requires alternative methods for *FD* estimation. Because the analysed regions are textured, their contour images (edges) are full of edge pixels, and the first points in the log-log representations give a partial *FD* equal with 2. Therefore, we proposed another fractal dimension which we named *effective fractal dimension* (*EFD*). *EFD* is calculated by the omission of the first points in the log-log representation (the points of the form (x_i , x_i^2), i = 1,2,…,k) (13).

$$a_E = EFD = \frac{n\sum_{i=k+1}^{n} x_i y_i - \left(\sum_{i=k+1}^{n} x_i\right)\left(\sum_{i=k+1}^{n} y_i\right)}{n\sum_{i=k+1}^{n} x_i^2 - \left(\sum_{i=k+1}^{n} x_i\right)^2} \qquad (13)$$

The threshold assessment which is used for edge extraction constitutes a problem of the fractal dimension evaluation in the grey level image case. Thus, we proposed some methods for threshold establishment:

1. The values for which the contur conserve the texture of image;
2. The value for which there are the most of points to contur;
3. The values for which the contour pixel set is a nonempty one;

In Fig.11 it is presented the contour images for $I_1(1)$,, $I_1(4)$, and $I_2(4)$, when the most points in the contour images appear.

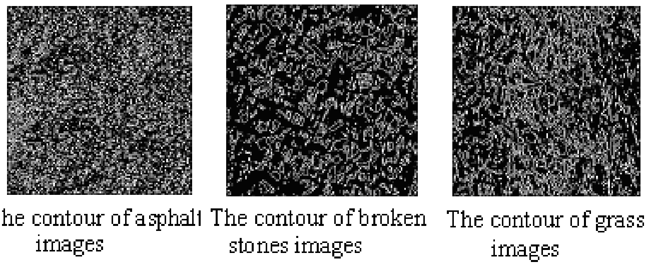
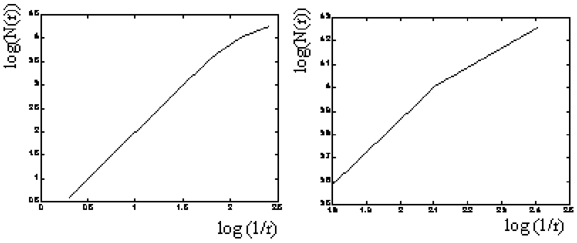

The contour of asphalt images
The contour of broken stones images
The contour of grass images

Fig. 11. The contour images for some textured images (most of points).

In Fig. 12 we presented the log-log curve in the basic fractal dimension case and the modified log-log curve for *EFD* calculus of the region *I*11.



a) b)

Fig. 12. The curve log-log for the box-counting fractal dimension evaluation. a) Entire grey level range; b) Selected grey level range.

If different thresholds for edge extraction are used, one can plot the fractal dimension dependence from threshold value (Fig. 4). From this one can compute MFD in different cases: 1.All the thresholds for nonempty contour points, and 2.The selected ones for texture preservation. In the second case, for each region, we considered a minimum threshold (T_{min}), when the edge texture becomes observable, and a maximum threshold (T_{max}), when the edge texture disappears. Thus, we can compute the fractal dimensions FD_j for each threshold between T_{min} and T_{max}, and the resulting mean fractal dimension *MFD* (12). The results are presented in Table 4.

Table 4. The fractal dimension

Image region	Threshold Min-Threshold Max	MFD	EFD
I11	150-190	1.325	0.937
I13	150-190	1.324	0.921
I21	50-140	1.674	1.242
I23	50-140	1.672	1.230
I31	80-120	1.763	1.372
I33	80-120	1.759	1.353
C11	80-160	1.618	1.229
C21	80-160	1.631	1.259

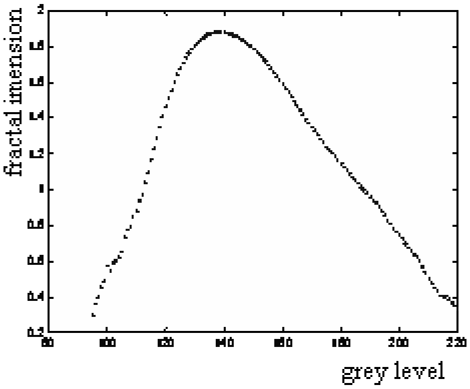


Fig. 13. The dependencies of the box-counting fractal dimension on the threshold, in the asphalt image (*I*11) case.

One can observe that MFDs and EFDs for asphalt regions are appropriated (I11, I15). Also, MFDs and EFDs for other textures, like pebble, grass, mixed, are not appropriate from asphalt.

V. Conclusion

Because it is considered an average co-occurrence matrix, the presented algorithm is relatively insensible to image translation and rotation. The results confirm that the statistic second order features, extracted from medium co-occurrence matrices, offer a good discriminating power both in texture similarity evaluation and in defect region detection and identification. The application of the algorithm consists in road (asphalt) identification and segmentation based on image partition and textured regions defect region detection (pebble or grass). The additional features like difference image histograms and edge pixel density per unit of area can increase the power of discriminating for texture identification and classification. The efficiency of the road following and defect region detection and localization depends on the range of image partition. The most important features, with greater discriminating power, both in texture similarity evaluation and in defect region detection and identification, are the contrast and the energy.

The fractal dimensions offer a good discriminating power both in texture similarity evaluation and in defect region detection. The advantage of using the fractal dimension consists in a significant decrease of the number of calculations.

References

[1] Haralick, R.M. et al. - Textural Features for Image Classification, *IEEE Trans. on Systems, Man. And Cybernetics*, vol.SMC-3, no.6, nov.1973, pp 610-621;

[2] Haralick, R.M., Shapiro, L.G. - *Computer and Robot Vision*, Add.-Wesley, Pub. Co., 1992;

[3] Tamura, H.; Mori, S.; Yamawaki, T. - Texture features corresponding to visual perception, *IEEE Trans.* On *Systems, Man and Cybernetics*. 6(4):460-473, 1976;

[4] Niblack, W. et al. - The QBIC Project: Querying Images by Content Using Color, Texture and Shape, *Proc. of the Conference Storage and Retrieval for Image and Video Databases*, SPIE vol.1908, pp.173-187, 1993;

[5] Liu, F. and Picard, R.W. - Periodicity, directionality and randomness: World features for image modeling and retrieval, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18(7):722-733, 1996;

[6] Kaplan, L.M. et al. - Fast texture database retrieval using extended fractal features, in Storage and Retrieval for Image and Video Databases VI (Sethi, I K and Jain, R C, eds.), *Proc SPIE* 3312, 162-173, 1998;

[7] Smith, J. - *Integrated Spatial and Feature Image System: Retrieval, Analysis and Compression*, Ph. D. Thesis, Columbia University, 1997;

[8] Deng,Y. - *A Region Representation for Image and Video Retrieval*, Ph. D. thesis, University of California, Santa Barbara, 1999;

[9] Popescu, D., Dobrescu, R., Avram, V., Mocanu, St. – Dedicated Primary Image Processors For Mobile Robots, *WSEAS Trans. on Systems*, Issue 8, Vol.5, August 2006, pp. 1932-1939.

[10] Shahrokni, A., Drummond, T., and Zisserman: Texture boundary detection for real-time tracking. In *Proc. European Conference on Computer Vision*, volume 1, pp. 566-577, 2004;

[11] Dobrescu R., Ionescu, F. *Fractal dimension based technique for database image retrieval*, Proceedings of the IAFA Symposium, Bucharest, Romania, 2003, pp. 107-112;

[12] Barnsley, M., S.G. Demko, *Iterated Function Systems and the Global Construction of Fractals*, Proc.Roy.Soc.London, Ser.A 399, 1985, pp. 243-275;

[13] Petrou M. and P. Garc a-Sevilla, *Image Processing: Dealing with Texture*, 2006 John Wiley & Sons, Ltd. ISBN: 0-470-02628-6;

[14] Fauqueur, J.; Kingsbury, N.; Anderson, R, *Semantic discriminant mapping for classification and browsing of remote sensing textures and objects*, Image Processing, ICIP 2005, Volume 2, 11-14 Sept. 2005, pp. II-846-9;

[15] Maenpaa T and Pietikainen M, *Texture analysis with local binary patterns*, in *Handbook of Pattern Recognition and Computer Vision*, C H Cahen and P S P Wang (eds), World Scientific 2005, pp 197–216;

[16] Pietikainen M, Ojala T and Xu Z, Rotation invariant texture classification using
feature distributions, *Pattern Recognition*, Vol 33, 2000, pp 43–52;

[17] Pranam, J., Yu, Z., *Invariant Features of Local Textures a rotation invariant local texture descriptor"*, IEEE Conference on Computer Vision and Pattern Recognition, CVPR '07. 17-22 June 2007, pp 1-7

[18] Gonzalez, R. C. and R. E. Woods: *Digital Image Processing*. Addison Wesley, Reading Mass, second edition, 1992;

[19] Keller, J. M. and S. Chen: Texture Description and Segmentation through Fractal Geometry, *Computer Vision, Graphics and Image Processing* 45, 1989, pp. 150–166;

[20] Wilson, R**.** and M. Spann: *Image Segmentation and Uncertainty*, John Wiley and Sons Inc., New York, 1988;

[21] B.B. Mandelbrot, *Fractal Geometry of Nature, Freeman*, New York, 1982;

[22] Lam, N.S-N., Qiu, H. L., Quattrochi, D. A., and Emerson, C. W.: An Evaluation of Fractal Methods for Characterizing Image Complexity. *Cartography and Geographic Information Science*, 29(1): 25-35, 2002;

[23] Jaggi, S., Quattrochi, D., and Lam, N. S.: Implementation of operation of three fractal measurement algorithms for analysis of remote sensing data, *Computers and Geosciences*, 19, 745–767, 1993;

[24] Emerson, C. W., Lam, N. S.-N., and Quattrochi, D. A.: Multiscale Fractal Analysis of Image Texture and Pattern, *Photogrammetric Engineering and Remote Sensing*, 65: 51-61, 1999;

[25] Haralick, R.M. et al. - Textural Features for Image Classification, *IEEE Trans. on Systems, Man. And Cybernetics*, vol.SMC-3, no.6, nov.1973, pp 610-621;

[26] D.Popescu, N.Angelescu, I.Caciula, I.Udroiu: Statistical Features Based Method For Textured Image Classification, *Proc.of 16-th Int.Conf.on Control Systems and Computer Science*, vol.3- Int. Symp. on Interdisciplinary Applications of Fractal Analysis, ISBN 978-973-718-741-3, Bucuresti, May 25-27, 2007, Editura POLITEHNICA Press, pp 28-32;

[27] D.Popescu, M. Nicolae, D. A. Crisan, N. Angelescu: Fractal And Texture Features In Tumour Detection, *WSEAS Trans. on Signal Processing, Issue 10,* Vol.2, October 2006, p. 1387-1395, ISSN 1790-5022.

[28] D.Popescu, R.Dobrescu, M. Nicolae: Texture Classification and Defect Detection by Statistical Features, *NAUN International Journal Of Circuits, Systems And Signal Processing, vol.1/2007*, pp79-84.