Diyala Journal of Engineering Sciences

ISSN 1999-8716
Printed in Iraq

Second Engineering Scientific Conference
College of Engineering –University of Diyala
16-17 December. 2015, pp. 453-461

CONTOUR PROCESSING OF TEXTURE IMAGES

H.M. ALZAKKI, V.YU. TSVIATKOU
Dept.Telecommunication networks and devices, Belarusian State University of Informatics and Radioelectronics
Brovky St. 6, Minsk, 220013, Belarus
haidermakki300@yahoo.com, vtsvet@bsuir.by

ABSTRACT: - A new approach to video coding is presented, where video scenes are classified into textures with subjectively relevant and irrelevant details. We apply this idea to improve video coding by using a texture analyzer and a texture synthesizer.

Keywords: cantor processing, image segmentation, texture analyzer.

1. INTRODUCTION

Due to higher resolution video cameras increases the likelihood of texture regions in images. Therefore, to process such images need to use methods of texture analysis and segmentation. With the development of high-definition television texture image are becoming more common. Video coding standard MPEG-4 also allows the texture of the image. Known methods of texture segmentation are based on [1] .The main disadvantages of these methods are a great time segmentation and a high probability of a segmentation fault. In this regard, it is urgent to develop new faster and more accurate methods of texture segmentation.

1 .Texture analysis, segmentation and synthesis in the MPEG-4

Many video scenes contain textures like grass, trees, sand, water, clouds, etc. These textures are difficult to code because of the large amount of visible detail. However, the exact reproduction of these textures can be considered as not important if they are shown with a limited spatial accuracy – the details are often irrelevant. Moreover, since the viewer typically does not know the original video, it is very unlikely that the exact reproduction of details is important. The viewer should just be able to recognize the textures, which is often not the case when for instance a pre-filter is utilized or these are blurred due to strong quantization. We exploit this idea for video coding using a texture analyzer at the encoder side and a texture synthesizer at the decoder side as shown in Figure(1).

The texture analyzer identifies detail-irrelevant texture regions, creates coarse masks corresponding to these regions and signals these masks as side information to the decoder to run the texture synthesizer. The texture synthesizer replaces the textures identified by the masks via inserting synthetic textures. The most important underlying assumption of the presented approach is that for the identified detail-irrelevant textures, known distortion criteria like mean squared error (MSE) are not suitable for efficient coding, since irrelevant detail may be reproduced[1].

2. Methods of texture analysis and segmentation

There are four major issues in texture analysis [2]:

1. Feature extraction: to compute a characteristic of a digital image able to numerically describe its texture properties;
2. Texture discrimination: to partition a textured image into regions, each corresponding to a perceptually homogeneous texture (leads to image segmentation);
3. Texture classification: to determine to which of a finite number of physically defined

classes (such as normal and abnormal tissue) a homogeneous texture region belongs;
4. Shape from texture: to reconstruct 3D surface geometry from texture information.

A. Gabor Filter

In image processing, a Gabor filter, named after Dennis Gabor, is a linear filter used for edge detection. Frequency and orientation representations of Gabor filters are similar to those of the human visual system, and they have been found to be particularly appropriate for texture representation and discrimination.

In the spatial domain, a 2D Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave [3].

B. Markov Random Fields

A Markov random field (MRF) is a probabilistic process in which all interactions is local the probability that a cell is in a given state is entirely determined by probabilities for states of neighbouring cells (Blake 1987). Direct interaction occurs only between immediate neighbours. However, global effects can still occur as a result of propagation. The link between the image energy and probability is that P* a *exp(E /T) .Where T is a constant. The lower the energy of a particular image (that was generated by a particular MRF), the more likely it is to occur [4].

C. Thresholding

The simplest method of image segmentation is called the thresholding method. This method is based on a clip-level (or a threshold value) to turn a gray-scale image into a binary image. The key of this method is to select the threshold value (or values when multiple-levels are selected). Several popular methods are used in industry including the maximum entropy method, Otsu's method (maximum variance), and k-means clustering. Recently, methods have been developed for thresholding computed tomography (CT) images [5].

D. Histogram-Based Methods

Histogram-based methods are very efficient when compared to other image segmentation methods because they typically require only one pass through the pixels. In this technique, a histogram is computed from all of the pixels in the image, and the peaks and valleys in the histogram are used to locate the clusters in the image. A refinement of this technique is to recursively apply the histogram-seeking method to clusters in the image in order to divide them into smaller clusters. This is repeated with smaller and smaller clusters until no more clusters are formed. One disadvantage of the histogram-seeking method is that it may be difficult to identify significant peaks and valleys in the image [6].

E. Others methods

Edge detection, content-based image retrieval, supervised segmentation, unsupervised segmentation, clustering methods [2].

3 .texture segmentation method based on the energy map

Another approach to generating texture features is to use local masks to detect various types of texture. Laws developed a texture-energy approach that measures the amount of variation within a fixed-size window. A set of nine 5 x 5 convolution masks is used to compute texture energy, which is then represented by a vector of nine numbers for each pixel of the image being analyzed. The masks are computed from the following vectors.

L5 (Level) = [1 4 6 4 1]
E5 (Edge) = [-1 -2 0 2 1]
S5 (Spot) = [-1 0 2 0 -1]
R5 (Ripple) = [1 -4 6 -4 1]

The names of the vectors describe their purposes. The L5 vector gives a center-weighted local average. The E5 vector detects edges, the S5 vector detects spots, and the R5 vector detects ripples. The 2D convolution masks are obtained by computing outer products of pairs of vectors. For example, the mask E5L5 is computed as the product of E5 and L5 as follows.

$$\begin{bmatrix}-1\\-2\\0\\2\\1\end{bmatrix} \times \begin{bmatrix}1 & 4 & 6 & 4 & 1\end{bmatrix} = \begin{bmatrix}-1 & -4 & -6 & -4 & -1\\-2 & -8 & -12 & -8 & -1\\0 & 0 & 0 & 0 & 0\\2 & 8 & 12 & 8 & 2\\1 & 4 & 6 & 4 & 1\end{bmatrix}$$

First step in Laws' procedure is to remove effects of illumination by moving a small window around the image, and subtracting the local average from each pixel, to produce a preprocessed image, in which the average intensity of each neighbourhood is near to zero. The size of the window depends on the class of imagery; a 15 X 15 window was used for natural scenes. After the preprocessing, each of the sixteen 5 X 5 masks are applied to the preprocessed image, producing sixteen filtered images [7].

And energy map cannot separate the all different part on the image. It not working with all images and separates its good. We can see this for example in the Fig(2) . The energy map working good with some parts but with others it cannot separate the edge and treat with its like one region. In the fig (3) we can see it working good and separate all parts on the image and find all edge

4 .Texture analysis based on binary masks

In this table(1) can see the proccessing on the image at first doing cantor for image and then use region_growing [8] mothed and then image zone which it the result of our descriptor table(2) for theses pixels which repeak on image more than 64 times .

5 .Assessment of the effectiveness of methods of texture analysis and segmentation

In this table(2) above are build descriptor the first Colum is for image which are choice for classify and the second Colum for the pixel value in image and third Colum for how many this pixel in an image and from 1 to 8 is describer . We are taken 3x3 mask fig (4) and doing scan for neighbors pixels and determine how many neighbors have same value for each pixel is chosen by this way we are build our describer.

We are taken this pixel which is repeat in image more than 5 and less than 64 times. By this descriptor we can search different lines and cell and whole. In this descriptor from 1 to 8 Colum the largest energy is found in 1, 2 and 3 Colum and less than actually to 8 Colum. For more make sure we are search by this mask 3x3 the end point for each line fig (5). we are take this pixel and doing scan for neighbors and check if this pixel have two same neighbors just only this two neighbors must be neighbors too .

By this way we can make sure this line have end points and it take form line maybe short or long line but it have end points. If we are taken for example the image Barb128 and build128 fig (7).

We can see different types of lines short and long lines but each of this have end points and have different types because its different in energy therefore take different form .But at same time our method cannot find some pixels which have all conditions and form line .If we take image barb128 and see to it describer it must should be line in table (3).

Now we can see for example pixel which have value (119) it have two end points and max energy in 3th Colum but our method cannot find it , and same with others images .For pixels which is repeat in the image 4 times to classify its we are suggestion this else point or line and for each case his condition . to search and classify its we are use mask 2x2 and it move around the pixel and search the same neighbors it meaning this pixel take four places in our mask how it explain in this fig(8).

For case point we are suggestion it will be when this pixel have three neighbors have same value fig (9).It will be show in image new_7 we can see these points fig (10).In other case are classify it like line (short line) as show in fig (11)

If pixel which is repeat 3 times we are too suggestion to classify it’s like point and line (short line).And we are too use mask 2x2 and it move around the pixel and search the same neighbors it meaning this pixel take four places in our mask .

For the point we are take condition if this pixel have two same pixel it meaning point how show in fig(12).It will be show in image new_7 and build128 we can see these points fig(13).In other case are classify it like line (short line) as show in fig(14).

If pixel it repeat in the image one or two times we will not using any mask it will be direct point. We can see this for example in the image Sea 64 (point1 and 2) fig (15).
For pixels which are repeat more than 64 times we are classify its direct like big zone and we can see it for example in image Sea 64 and Build 128 (zone) fig(16).

Conclusion

In our work we are use different images with different sizes first process for image is to do image cantor with different filters direction and then the image go to region _grown method the out image after region grown go to analysis and classification to classify the texture region on the image. In this work it classified like (points, lines (different), and zone). For this there is descriptor which can search and classify texture region. In our work we are take only theses pixels which is repeat less than 64 times if it’s more than 64 times it classified like zone on the image.

References

1) S.-Y. Yoon and E. H. Adelson, “Subband texture synthesis for image coding”, Proceedings of SPIE on Human Vision and Electronic Imaging III, Vol. 3299, pp. 489-497, San Jose, CA, USA, January 1998
2) Vaijinath V. Bhosle on Texture Segmentation: Different Methods / Vaijinath V. Bhosle, Vrushsen P. Pawar// International Journal of Soft Computing and Engineering (IJSCE), India. November 2013 ISSN: 2231-2307, Volume-3, Issue-5, P.69-74.
3) Amanpreet Kaur (2012) “Texture Based Image Segmentation using Gabor filters” [IJESAT] International Journal of Engineering Science & Advanced Technology Vol-2, Issue-3, 687 – 689.
4) Erdogan Çesmeli and DeLiang Wang “Texture Segmentation Using Gaussian–Markov Random Fields and Neural Oscillator Networks” IEEE Transactions on Neural Networks, Vol. 12, No. 2, March 2001.
5) Magdolna Apro1, Szabolcs Pal2, Sandra Dedijer1 “Evaluation of single and multi-threshold entropy-based algorithms for folded substrate analysis” Journal of Graphic Engineering and Design, Volume 2 (2), 2011
6) Xiuwen Liu, Member, IEEE, and DeLiang Wang, Fellow, IEEE “Image and Texture Segmentation Using Local Spectral Histograms” IEEE Transactions On Image Processing, Vol. 15, No. 10, October 2006.
7) K. Laws, “Textured image segmentation,” Ph.D. dissertation, University of Southern California, 1980.
8) Shilpa Kamdi on Image Segmentation and Region Growing Algorithm/ Shilpa Kamdi, R. K. Krishna// International Journal of Computer Technology and Electronics Engineering (IJCTEE). India. ISSN 2249-6343, Volume 2, Issue 1

Table(1): proccessing on the image

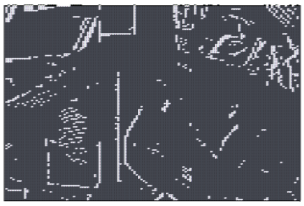
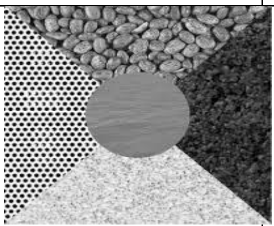
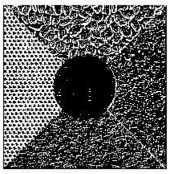
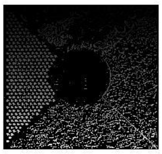
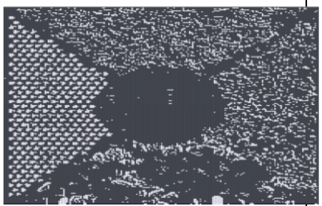
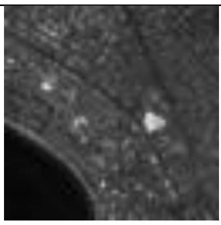
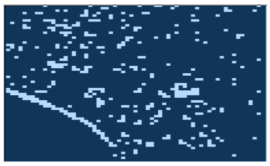
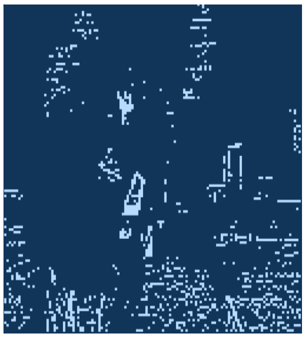
Image	Image cantor	Region _growing	Image zone
 Barb 128			
 New_7			
 Sea 64			
			

Table (2): Assessment of the effectiveness of methods of texture analysis and segmentation

Picture	**Pixel**	**Sum**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
	24	8	2	2	4	0	0	0	0	0
	26	5	1	3	1	0	0	0	0	0
Barb128	101	26	2	16	8	0	0	0	0	0
	138	5	1	3	1	0	0	0	0	0
Sea 64	102	5	0	2	2	1	0	0	0	0
	98	57	0	3	13	23	13	5	0	0
	228	16	1	4	6	3	1	1	1	1
New_7	263	36	1	10	18	5	1	1	0	0
	148	5	1	3	1	0	0	0	0	0
	129	9	3	5	1	0	0	0	0	0
Build128	218	26	6	11	8	1	0	0	0	0
	237	20	1	5	10	3	1	0	0	0

Table (3): different types of lines short and long lines

Picture	Pixel	Sum	1	2	3	4	5	6	7	8
Barb128	119	10	0	2	4	4	0	0	0	0
	231	9	1	3	1	4	0	0	0	0

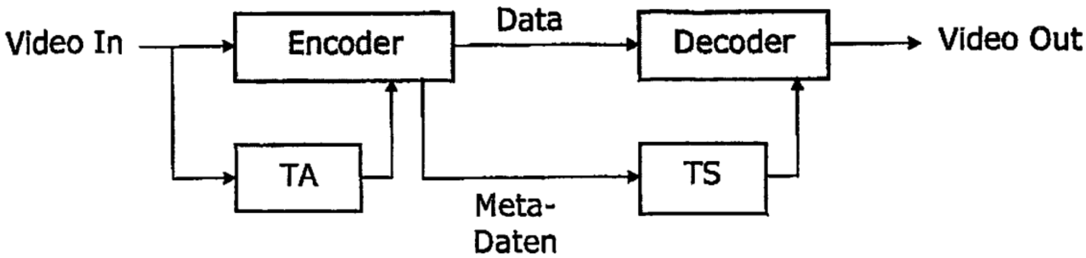
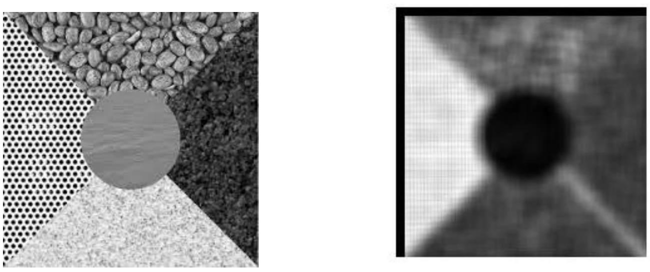
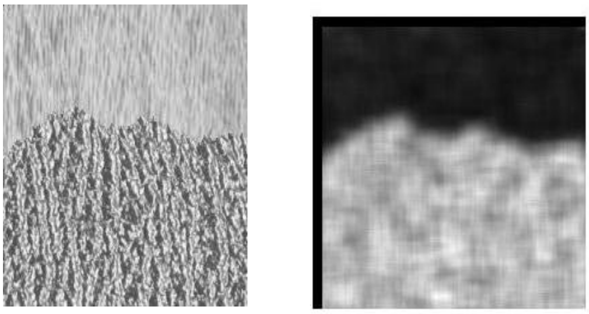


Figure (1): segmentation and synthesis in the MPEG-4



a b

Figuer(2): a.new_7 (energy map),b. new_7 (energy map)



a b

Figuer(3): a.new_8 (energy map) , b.new_8 (energy map)

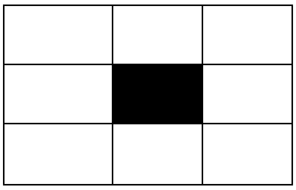


Figure (4):3x3 mask

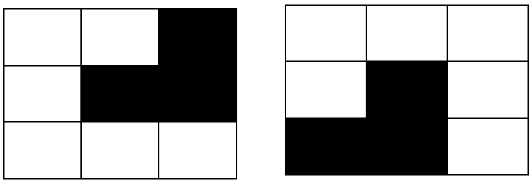
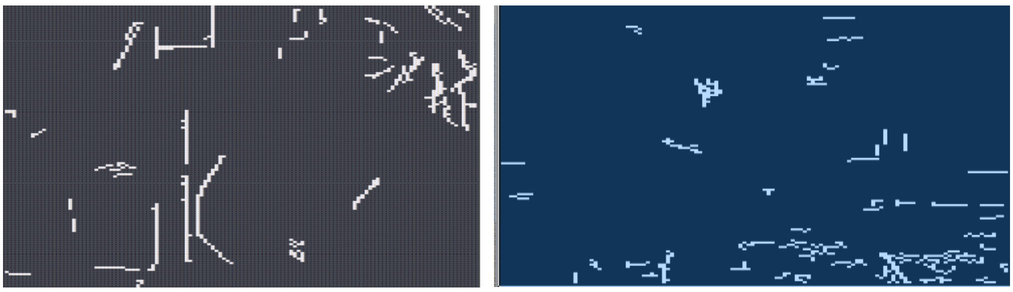
a  b

Figure (5): mask 3x3 the end point for each line



a b

Figuer(6): a.barb128 ,b.build128

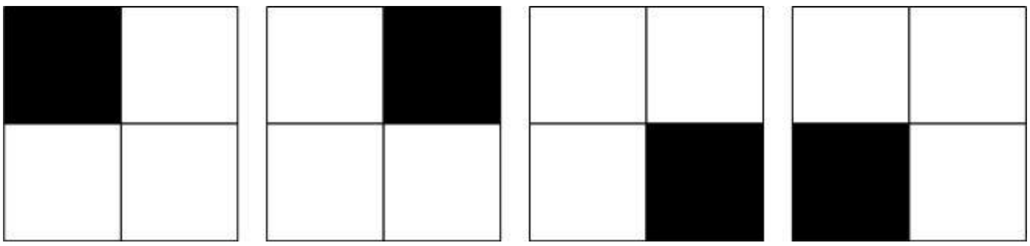


Figure (8): Mask 2x2

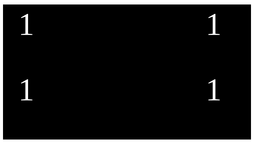


Figure (9): Three neighbors

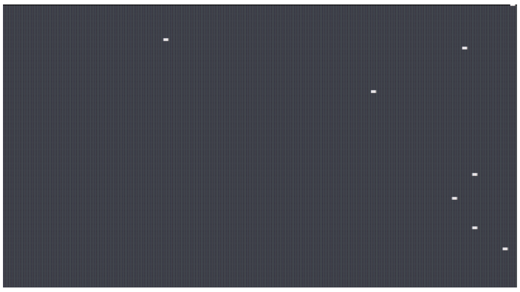
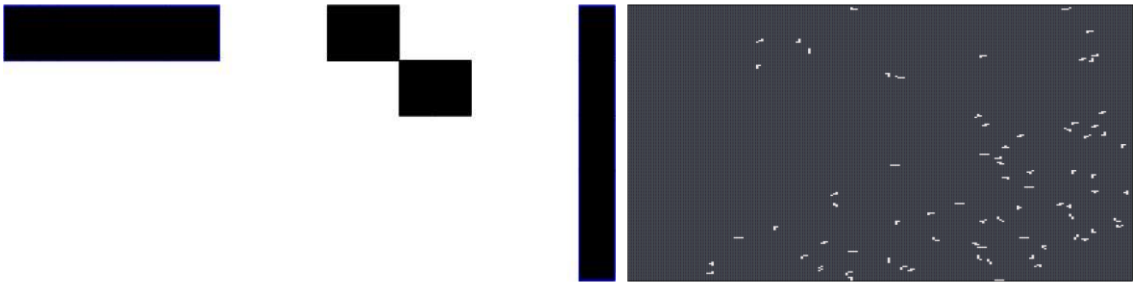


Figure (10): short line



a b

Figure (11): a. shape of line ,b. new_7

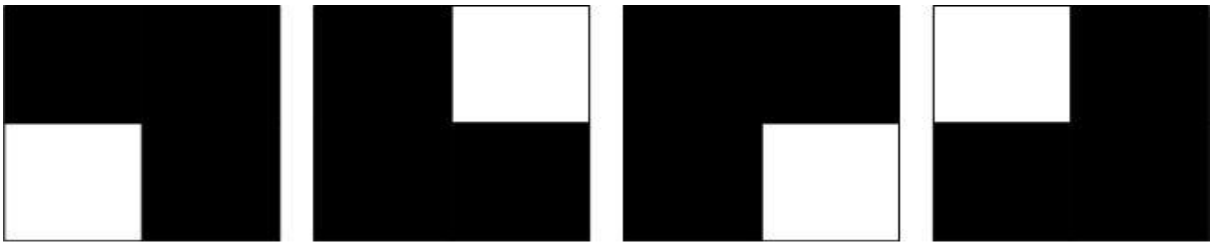
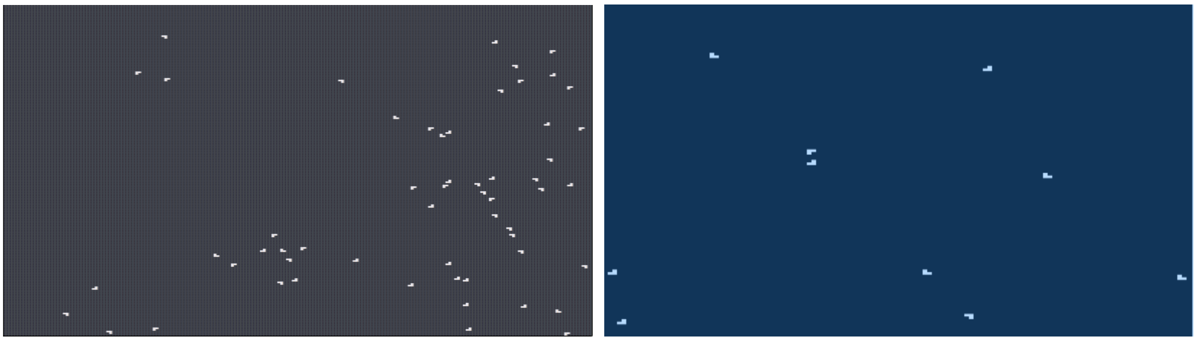
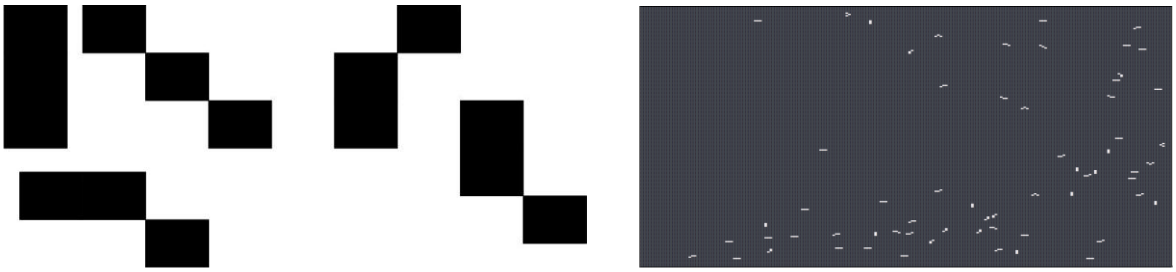


Figure (12): image new_7 and build128



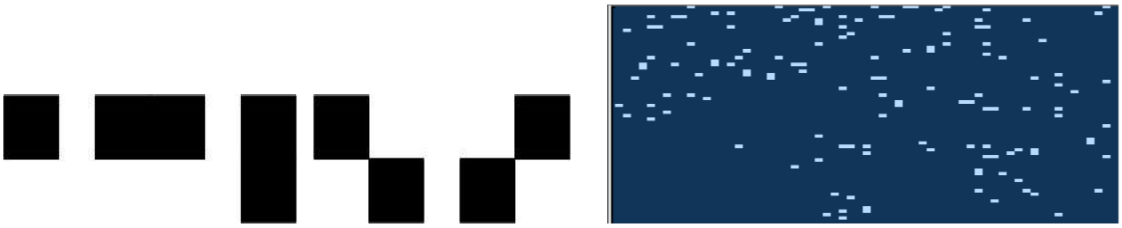
a b

Figure (13): a.new_7, b.build128



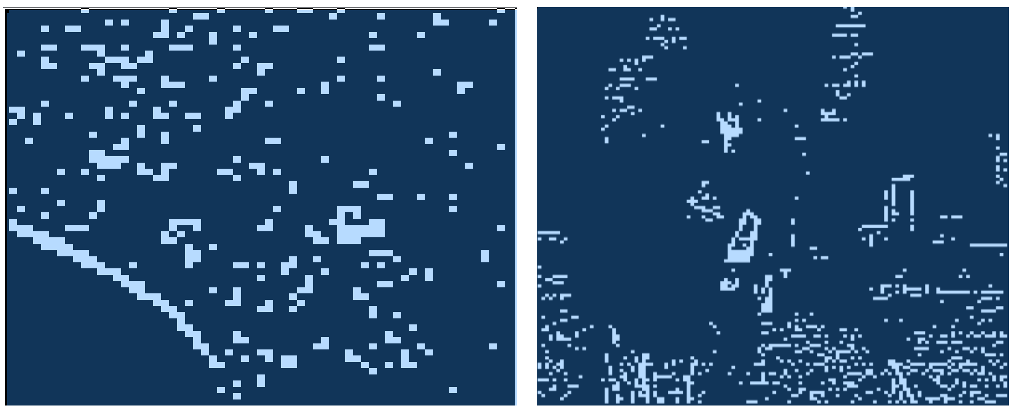
a b

Figure (14): a. shape of line, b.build128



a b

Figure (15): a. shape of point, b.sea64



a b

Figure (16): a.sea, b.build128