A Comparative Study of Texture Features for Image Segmentation

G. Madasamy Raja[1] and V.Sadasivam[2]

ABSTRACT

Image segmentation is one of the most significant tasks in image processing. The outcome of image segmentation is a group of regions that collectively cover the entire image, each of the pixels in a region are homogeneous with respect to some characteristic or computed property, such as color, intensity, or texture. Already there are many approaches proposed for texture feature extraction which can be useful for image segmentation. One of the important issues here is how well these methods work on differentiating various textures that are available in a single image. This paper considers two texture measures namely Texture Spectrum and Uniform Local Binary Pattern for texture segmentation and evaluates their performance based on the segmentation accuracy. Two different synthetic images are used in experiments. One image contains four different textures and another one contains two different textures. MATLAB has been used for the implementation purpose.

Keywords : Texture, Texture Spectrum, Local Binary Pattern, Uniform Local Binary Pattern, Texture Segmentation, Supervised Segmentation.

[1]Department of Computer Science and Engineering, Anna University of Technology, Coimbatore. Email: anushpriya2004@yahoo.com.

[2]Professor and Head, Department of Computer Science and Engineering, Manonmaniam Sundaranar University, Tirunelveli.

I. INTRODUCTION

MAGE segmentation refers to the process of partitioning a digital image into multiple regions (sets of pixels) [1]. The goal of segmentation is to simplify and/or change the representation of an image into something that is more

meaningful and easier to analyze. The result of image

segmentation is a set of regions that collectively cover the entire image, or a set of contours extracted from the image.

I Each of the pixels in a segmented region is similar with respect to some characteristic or computed property, such as color, intensity, or texture. Some of the practical

applications of image segmentation are :

- Medical Imaging
    - o Locate tumors and other pathologies
    - o Measure tissue volumes
    - o Computer-guided surgery
    - o Diagnosis
    - o Study of anatomical structure
- Locate objects in satellite images (roads, forests, etc.)
- Face recognition.

Segmentation merely based on the grey value alone is not efficient and features like color, texture, gradient magnitude or orientation, measure of a template match etc., can also be considered for the better output. Texture is an important characteristic for the analysis of many types of images [2].

Image texture is believed to be a rich source of visual information about the nature and three-dimensional shape of physical objects. Nowadays texture based finger print matching is an active research area because textures are complex visual patterns composed of entities or sub-patterns that have characteristic brightness, color, slope, size, etc., [3].

Texture measurements can also be used to segment an image and classify its segments [4]. Texture segmentation is to segment an image into regions according to the textures of the regions. Texture classification or segmentation is not an easy problem because there is not any precise definition of what a texture is.

This paper evaluates two important texture measures that are widely used in the present research for image segmentation in variety of applications. In this paper rest of the portion is organized as follows. Section II describes the texture measures that are used in this study and section III discusses the algorithm which is used for the segmentation. Experimental results are provided in section IV and finally the results with concluding remarks are discussed in section V.

X_4	X_3	X_2
X_5	X_0	X_1
X_6	X_7	X_8

3^3	3^2	3^1
3^4	X_0	3^0
3^5	3^6	3^7

Figure 1. Texture Unit.

II. Texture Measures Used In This Study

A. Texture Spectrum Method

The Texture Spectrum, one of a statistical method of texture analysis, focuses on texture characterization and discrimination [5]-[7]. The texture spectrum is based on the computation of the relative intensity relations between the pixels in a small neighborhood and not on their absolute intensity values. The importance of the texture spectrum method is determined by the extraction of local texture information for each pixel and of the characterization of textural aspect of a digital image in the form of a spectrum. The texture spectrum method results in a vector which characterizes the original image and the output spectrum maintains the texture characteristics of the input image.

Texture Spectrum method uses a basic concept called Texture Unit (TU). A Texture unit is characterized by eight pixels each of which has one of three possible values (0,1,2), obtained from a neighborhood of 3*3 pixels. Fig. 1. shows the method of forming the Texture Unit. If the intensity value of the central pixel is considered as X0 and the intensity value of each neighboring pixel as Xi, the set that is considered as the smallest complete unit of the under consideration image is: X = {X0, X1, X2, ... , X8}.This technique compares the greylevel of the central pixel (the one which is currently being processed), X0, with those of its neighbors, Xi (1<i<8), and records three logical relationships: smaller, equal and greater; noted by E, and coded as "0" "1" or "2", respectively(1). By this way, each image pixel generates a Texture Feature Vector, called as Texture Unit which is defined as: TU = {E1... E8}. Equation (1) is used for deriving TU.

$$E_i = \begin{cases} 0 & \text{if } X_i < X_0 \\ 1 & \text{if } X_i = X_0 \quad \text{for } i = 1,2,\dots 8. \\ 2 & \text{if } X_i > X_0 \end{cases}$$

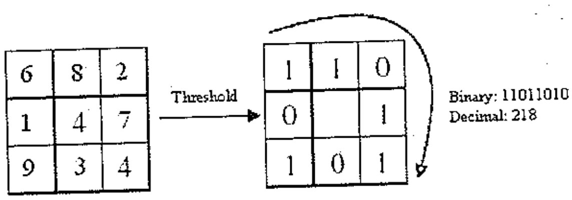

Figure 2 : A basic LBP operator.

According to equation (1), each element can be assigned one of three possible values so that the total number of possible texture units for the eight elements can be estimated as 38 = 6561. As there is no unique method for labeling the texture units, equation (2) is followed for numbering the texture units.

$$N_{TU} = \sum_{i=1}^{8} E_i * 3^{i-1} \qquad (2)$$

In the equation (2) NTU varies from 0 to 6560. Texture spectrum method is still a dominating texture measure in the research to reveal texture information in digital images and it has a promising discriminating performance for different textures [8].

B. Uniform Local Binary Patterns Method (ULBP)

Local Binary Patterns (LBP) operator introduced by Ojala et al [9] is a simple filter that labels an image by thresholding the neighborhood of each pixel with the value of the center pixel and gives the obtained result in binary values. Histogram of the labeled image is then used as a mean of texture description. The basic LBP operator is illustrated in Fig. 2. Many researches have been going on this texture descriptor method to enhance its uses for the various applications mainly for the reason that it is having very low computational complexity [10]-[13].

Equation (3) describes how each pattern in the image should be assigned a unique label.

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \qquad (3)$$

$$where\ s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

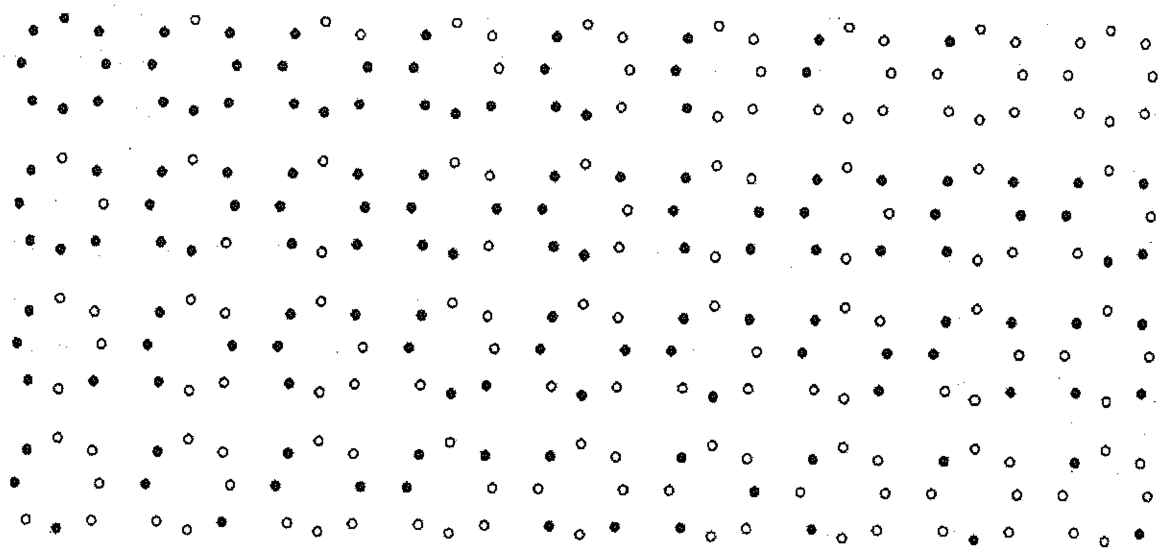

Figure 3 : The 36 unique rotation invariant binary patterns that can occur in the circularly symmetric neighborhood. Black and white circles correspond to bit values of 0 and 1 in the 8-bit output of the operator. The first row contains nine "uniform" patterns.

In equation (3) g_c is the intensity of the center pixel, g_p is the intensity of the neighbor p, $s(x)$ is the step function, P represents angular resolution and R represents spatial resolution. The LBP(P,R) operator produces 2P different patterns. It is clear that LBP defined in the equation (3) is not rotation-invariant as the intensity value of gp changes when the neighborhood circle is rotated by a specific angle. When the image is rotated, the pixels values will correspondingly move along the perimeter of the circle around. In order to remove the effect of rotation, Rotation Invariant Local Binary Pattern method was introduced and was defined as :

$$LBP^{ri}_{P,R}\ min\ \{ROR\ (LBP_{P,R}\ i)\ |\ i = 0, 1, \ldots, P-1\} \quad (4)$$

where *ROR(x,i)* performs a circular bit-wise right shift on Pbit number x , i times.

Two patterns should be treated as "Uniform", if one can be obtained from the other through rotating by a certain angle. The extension of Local binary pattern method in which uniformity measure, U, is defined as the number of spatial transitions between 1s and 0s in the pattern is called Uniform Local Binary Pattern method. Patterns that have uniformity values of at most 2 are designated as uniform patterns. The extended LBP which deals with uniform pattern is defined as per the equation (5).

$$LBP^{riu2}_{P,R} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c) & \text{if } U(LBP_{P,R}) \le 2 \\ P+1 & \text{otherwise} \end{cases} \quad (5)$$

$LBP^{ri}_{P,R}$ can have 36 unique rotation invariant binary patterns in the circularly symmetric neighbor set, which are shown in the Fig. 3. It was observed that more than 90 percent of all 3*3 neighborhood pixels present in any texture images are falling within this uniform patterns.

As Uniform Local Binary Pattern was not capable of effectively retrieving the textural information by merely considering the histogram of the uniform patterns, a new concept called DLBP was introduced. The dominant Local binary patterns (DLBP) concept, by S.liao et al. [14], considers the most frequently occurred patterns to capture descriptive textural information. DLBP does not contain any information about the dominant pattern types but the occurrence frequencies only. Recently the DLBP method has been successfully used for detecting the bleeding regions in human digestive tract [15] also.

In the conventional Uniform Local Binary Pattern method, sometimes the availability of noise converts useful patterns into non- uniform patterns so that they are not considered as uniform patterns. This problem is avoided in DLBP method [14]. Though DLBP was very good in encoding the pixel-wise information in the texture images, it does not consider the pixel interaction that takes place outside the coverage of its circular neighborhood system, which plays an important role in feature extraction in texture models.

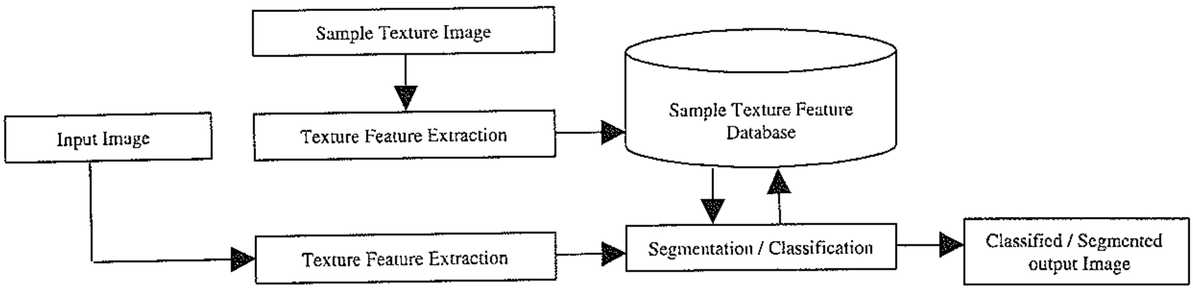

Figure 4 : Block diagram for segmentation

III ALGORITHM FOR SEGMENTATION

By following supervised segmentation, texture sample distributions are obtained by scanning the texture samples with the corresponding texture descriptors and corresponding histograms are calculated. The Fig. 4. shows the Block diagram for the following algorithm which is used for segmentation.

The segmentation algorithm is described as follows:

1. A random sample sub-image with the size of 30*30 pixels from each texture image (one sample per texture) is retrieved.
2. Texture Model is calculated for all sample texture images.
3. The input image is scanned by a window of 30*30 pixels and again the Texture Model is calculated for each window.
4. Texture model for every window of size 30*30 of the input image, is compared with the texture model of the each sample and the absolute difference (D) between them is calculated.
5. The central pixel of the window considered will be assigned to class K such that D(K) is minimum among all the D(i), for i= 1,2,3,4,......,where i represents the sample texture image.

IV. EXPERIMENTAL RESULTS

For this supervised study, five different texture images are used and samples are taken from the image with 25*25 pixels in size. Two synthetic images, one with four different textures and another with two different textures, are used as input images.

Experiment Setup #1

The above segmentation algorithm is applied to the four texture synthetic image by using Texture Spectrum as the

feature descriptor with varying scanning window size. The segmented results are shown in the Fig. 5., where four different textures are represented by four different grey levels.

In the second approach, the same four texture synthetic image is given as the input for the same segmentation algorithm by using Uniform Local Binary Pattern method as the texture feature descriptor with varying scanning window size and the results are shown in Fig. 6.

Experiment Setup #2

In these experiments, a two texture synthetic image is given as the input image for the segmentation algorithm. First Texture Spectrum method is used as the texture descriptor and the results are shown in the Fig. 7., where two different textures are represented by two different grey levels. Fig. 8. shows the segmentation results, when the Uniform Local Binary Pattern is the feature extraction method.

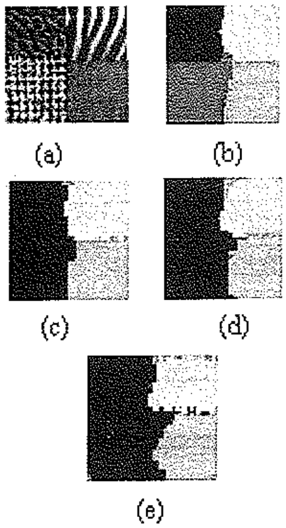

Fig. 5. (Texture Spectrum).
(a) Original Input Image
(b) Window Size is 25*25
(c) Window Size is 20*20
(d) Window Size is 15*15
(e) Window Size is 10*10

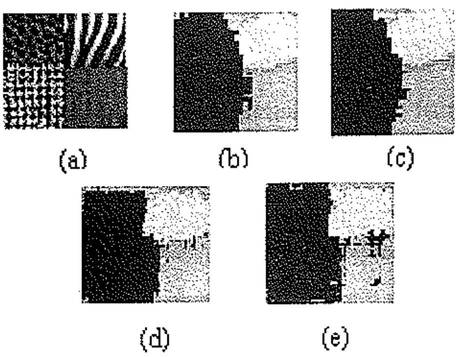

Fig. 6. (Uniform Local Binary Pattern).
(a) Original Input Image
(b) Window Size is 25*25
(c) Window Size is 20*20
(d) Window Size is 15*15
(e) Window Size is 10*10

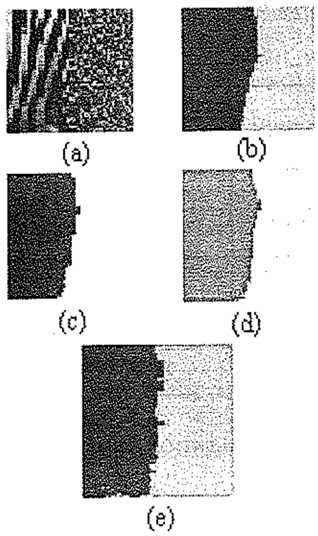

Fig. 7. (Texture Spectrum).
(a) Original Input Image
(b) Window Size is 25*25
(c) Window Size is 20*20
(d) Window Size is 15*15
(e) Window Size is 10*10

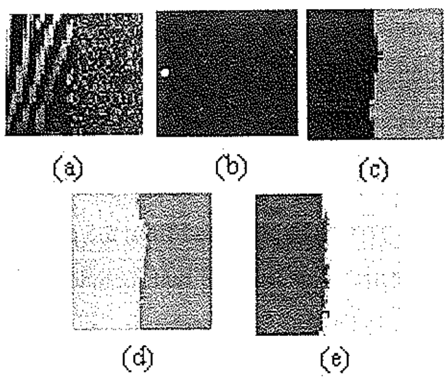

Fig. 8. (Uniform Local Binary Pattern).
(a) Original Input Image
(b) Window Size is 25*25
(c) Window Size is 20*20
(d) Window Size is 15*15
(e) Window Size is 10*10

Table I. Segmentation Accuracy
(for four – texture image)

S.No	Window Size	Segmentation Accuracy - Texture spectrum	Segmentation Accuracy - ULBP
1	25*25	94%	80%
2	20*20	92%	78.3%
3	15*15	91%	76.67%
4	10*10	85%	68.4%

Table II. Segmentation Accuracy
(for two – texture image)

S.No	Window Size	Segmentation Accuracy - Texture spectrum	Segmentation Accuracy - ULBP
1	25*25	94%	98%
2	20*20	94%	97%
3	15*15	95%	97%
4	10*10	98%	97%

V. Experimental Evaluation And Conclusion

By this study, it is noted that both Texture Spectrum and Uniform Local Binary Pattern methods were simple for implementation because only few mathematical operations were needed. These methods facilitate a very straightforward and efficient implementation, which may be mandatory in time critical applications. Table I shows the segmentation accuracy for the varying window size with the four-texture image and Table II shows the segmentation accuracy for the varying window size with the two-texture image. Here, segmentation accuracy rates are calculated over all the pixels including the region near the boundaries of textures. If we remove these pixels from the counter, the segmentation accuracy will be higher.

Both Texture Spectrum and uniform Local Binary Pattern methods have been evaluated from the point of view of discriminating performance that includes the influence of the boundaries of different textures. When Texture Spectrum method was used, promising segmentation results have been obtained with the average segmentation rate of 92%, whereas Uniform Local Binary Pattern gives an average segmentation rate of 86%. The bar chart (Fig. 9.) shows the comparative result analysis between Texture Spectrum and Uniform Local Binary Pattern methods on the basis of segmentation error rate for the given four-texture image.

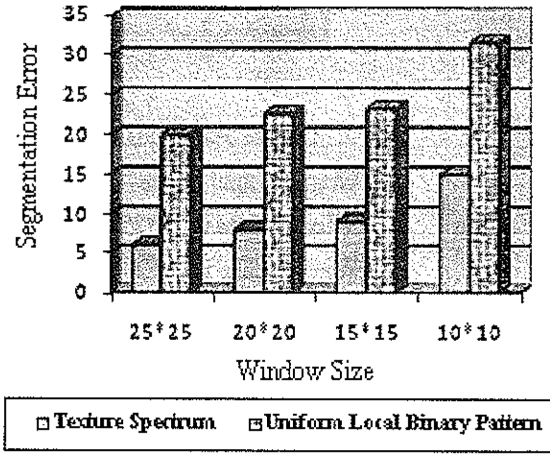

Figure 9. Comparative result on the basis of segmentation error rate with varying window size for the four- texture image.

Further evaluation shows that Texture Spectrum is sensitive to the directional aspect of texture whereas uniform Local Binary Pattern is sensitive to uniformity of texture in nature. In the case of Uniform Local Binary Pattern, it can be noted in the result that the top left texture (in the four-texture input image) was not segmented properly because of its non uniformity in the textural aspect. Greater the uniformity in the texture, more quality we can get in the output, which was proved in the case of the bottom left texture. As far as Uniform Local Binary Pattern method is concerned, it considers only 9 patterns as "Uniform" and it considers only these 9 patterns for segmentation. This is the reason for not producing smooth boundaries in the process of texture segmentation. The comparative result analysis between

Texture Spectrum and Uniform Local Binary Pattern method, in segmenting the given two-texture image on the basis of segmentation error rate is shown in the Fig. 10.

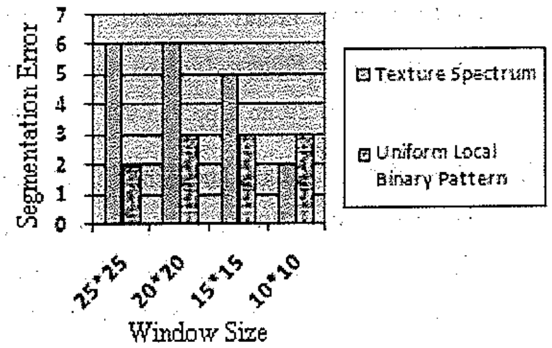

Figure 10 : Comparative result on the basis of segmentation error rate with varying window size for the two-texture image.

According to our results, it is noted that, Uniform Local Binary Pattern gives higher segmentation accuracy when the window size becomes high, irrelevant of number of textures available in a single image. Uniform Local Binary Pattern method is very robust in segmenting the images which are affected even with of grey scale variation due to poor lighting.

Though Texture Spectrum method has given better performance, it is still possible to change the Uniform Local Binary Pattern method even powerful, by joining with some filters or contrast measures. Since both methods use the texture features from the neighborhood window, they cannot produce the smooth boundaries. With smaller window size, the texture features are not extracted completely from the neighborhood and at the same time, the larger value of window size will result in inaccurate segmentation near the boundaries, especially when the input image contains more number of textures. Moreover, when the window size is increased it increases the computation time also. So, the window size must be chosen in such a way that it covers the whole smallest unit of a texture unit or pattern. As a future enhancement, this study can be improved by applying various similarity measures, various segmentation algorithms and of course various texture features can also be included.

REFERENCES

[1] R.C.Gonzalez, R.E.Woods, and Steven L.Eddins, "*Digital Image Processing* using MATLAB", Pearson Education, 2004.

[2] T.Ojala, M. Pietikainen, and D.Harwood, "A comparative study of texture measures with classification based on feature distributions", *Pattern Recognition.*, vol. 29, no.1,pp.51-59,1996.

[3] Anil K.Jain, Davide Maltoni, Dario Maio "Handbok of Fingerprint Recognition", *Springer Publishers*, 2009.

[4] William K.Pratt, "*Digital Image Processing*", A Wiley Interscience Publication, John Wiley & sons Inc, U.S.A, 2007.

[5] Dong-Chen He and Li Wang, 1990, "Texture unit, texture spectrum and texture analysis", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 28, no.4, pp.509-512.

[6] Dong-Chen He and Li Wang, "Texture Segmentation Using Texture Spectrum", *Pattern Recognition*, vol. 23, pp.905-910, 1990.

[7] Dong-Chen He and Li Wang, "Texture Features Based on Texture Spectrum", *Pattern Recognition*, vol. 24.pp. 391-399, 1991.

[8] S.S.Sreejamole and L. Ganesan, "Supervised Classification of Image using Textural features", *International journal of soft computing*, vol. 4,

No.3, pp.122-126,2009.

[9] T.Ojala, M.Piettikainen, and T.Maenpaa, "Multiresolution grayscale and rotation invariant texture segmentation with local binary patterns", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no.7, pp. 971-987, Jul. 2002.

[10] G.Zhao and M.Pietikainen, "Local binary pattern descriptors for dynamic texture recognition", in Proc.18th *Int. Conf. Pattern Recognition*, 2006.pp. 211-214.

[11] T.Ahonen, A.Hadid, and M.Pietikainen, "Face description with local binary patterns: Application to face recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no.12, pp.2037-2041, Dec. 2006.

[12] D.Connah and G.D. Finlayson, " Using local binary pattern operators for color constant image indexing", in *Proc. CGTV 2006 Conf. Proc. 2006.*

[13] G.Zhao and M.Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.29, no.6, pp.915-928, Jun. 2007.

[14] S.Liao, Max W.K.Law, and Albert C.S.Chung, "Dominant Local Binary Patterns for Texture Classification", *IEEE Transactions on Image Processing*, vol. 18, no.5,pp.1107-1118, May 2009.

[15] Baopu Li and Max Q.H.Meng "Computer-Aided Detection of Bleeding Regions for Capsule Endoscopy Images", *IEEE Transactions on Biomedical Engineering*, vol.56, no.4,pp.1032-1039.