Global Journal of Computer Science and Technology
Graphics & Vision
Volume 12 Issue 15 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Fuzzy Based Texton Binary Shape Matrix (FTBSM) for Texture Classification

By P.Chandra Sekhar Reddy & B.Eswara Reddy
Jntua College of Engineering, Anantapur, A.P, India

Abstract - Texton is a extensively applied approach for texture analysis. This technique shows a strong dependence on certain number of parameters. Unfortunately, each variation of values of any parameter may affect the texture characterization performance. Moreover, micro structure texton is unable to extract texture features which also have a negative effect on the classification task. This paper, deals with a new descriptor which avoids the drawbacks mentioned above. To address the above, the present paper derives a new descriptor called Fuzzy Based Texton Binary Shape Matrix (FTBSM) for clear variation of any feature/parameter. The proposed FTBSM are defined based on similarity of neighboring edges on a 3×3 neighborhood. With micro-structures serving as a bridge for extracting shape features and it effectively integrates color, texture and shape component information as a whole for texture classification. The proposed FTBSM algorithm exhibits low dimensionality. The proposed FTBSM method is tested on Vistex and Akarmarble texture datasets of natural images. The results demonstrate that it is much more efficient and effective than representative feature descriptors, such as logical operators and GLCM and LBP, for texture classification.

Keywords : *texton, micro structure, fuzzy, shape component.*

GJCST-F Classification: *I.3.5*

FUZZY BASED TEXTON BINARY SHAPE MATRIX FTBSM FOR TEXTURE CLASSIFICATION

Strictly as per the compliance and regulations of:

Fuzzy Based Texton Binary Shape Matrix (FTBSM) for Texture Classification

P.Chandra Sekhar Reddy [α] & B.Eswara Reddy [σ]

Abstract - Texton is a extensively applied approach for texture analysis. This technique shows a strong dependence on certain number of parameters. Unfortunately, each variation of values of any parameter may affect the texture characterization performance. Moreover, micro structure texton is unable to extract texture features which also have a negative effect on the classification task. This paper, deals with a new descriptor which avoids the drawbacks mentioned above. To address the above, the present paper derives a new descriptor called Fuzzy Based Texton Binary Shape Matrix (FTBSM) for clear variation of any feature/parameter. The proposed FTBSM are defined based on similarity of neighboring edges on a 3×3 neighborhood. With micro-structures serving as a bridge for extracting shape features and it effectively integrates color, texture and shape component information as a whole for texture classification. The proposed FTBSM algorithm exhibits low dimensionality. The proposed FTBSM method is tested on Vistex and Akarmarble texture datasets of natural images. The results demonstrate that it is much more efficient and effective than representative feature descriptors, such as logical operators and GLCM and LBP, for texture classification.

Keywords : texton, micro structure, fuzzy, shape component.

I. Introduction

Texture classification is a fundamental issue in computer vision and image processing, playing a significant role in a wide range of applications that include medical image analysis, remote sensing, object recognition, document analysis, environment modeling, content-based image retrieval etc. [1]. For four decades, texture analysis has been an area of intense research, however analyzing real world textures has proven to be surprisingly difficult, in many cases caused by natural texture in-homogeneity of varying illumination, scale changes and variability in surface shape.

Many researchers have put forward various algorithms to extract color, texture and shape features for texture classification. Color is the most dominant and distinguishing visual feature. Texture features provide an important information of the smoothness, coarseness and regularity of many real-world objects such as fruit, skin, clouds, trees, bricks and fabric, etc. [10], and texture based algorithms are also widely used in CBIR systems, including the gray co-occurrence matrixes [2], Markov random field (MRF) model [3], simultaneous auto-regressive (SAR) model [4], Wold decomposition model [5], Gabor filtering [6,7] and wavelet decomposition [8,9] and so on. Tang [11] demonstrated that textural features extracted from a new run-length matrix can produce great classification results over traditional run-length techniques. Chen etal. Proposed a set of statistical geometrical features based on the statistics of geometrical properties of connected regions in a sequence of binary images.

Textures are classified recently by edge direction movements [12], classification and recognition of handwritten digits using mathematical wavelet transforms using first and second order statistics [13], skeleton extraction [14] and avoiding complex patterns [15]. Fuzzy based methods also proposed in the analysis of textures [16, 17], age classification problems are also proposed [18, 19, 20] in the literature based on texture features. The above methods captured different topological configurations and texture properties of the image. As a consequence, their performance is best suited for the analysis of textures.

The term ''texton'' is conceptually proposed by Julesz [21] and it is a very useful concept in texture analysis and has been utilized to develop efficient models in the context of texture recognition or object recognition [22, 23]. The texton [21] has been used in several classification problems [24, 25], age classification problem, face recognition, image retrieval [26]. These methods need high classification rate, which is however still an open problem. The present paper put forward a new method of Fuzzy Texton Binary Matrix to describe texture features for texture classification. This method can express the spatial correlation of micro structure textons.

The rest of this paper is organized as follows. In Section 2, the proposed methodology is introduced. In Section 3, the texture classification performance resulted from logical operators, GLCM, LBP and our proposed method is compared by conducting two experiments over the Vistex texture database of MIT, Akarmarble images and those images which come from web. Section 4 concludes the paper.

Author α : P.Chandra Sekhar Reddy Associate Professor, CSE Dept, Giet, Rajahmundry, A.P, India. E-mail : pchandureddy@yahoo.com
Author σ : B.Eswara Reddy Associate Professor, CSE Dept, Jntua College of Engineering, Anantapur, A.P, India.
E-mail : eswarcsejntu@gmail.com

II. Methodology Construction of Fuzzy Based Text on Binary Shape Matrix (FTBSM) of Textures

Various algorithms are proposed by many researchers to extract color, texture and other features. Color is the most distinguishing important and dominant visual feature. That's why color histogram techniques remain popular in the literature. The main drawback of this is, it lacks spatial information. Texture patterns can provide significant and abundance of texture and shape information. The proposed method consists of three steps which are listed below. In the first step the color image is converted in to grey level image by using any HSV color model. The following section describes the RGB to HSV conversion procedure.

a) RGB to HSV Color Model Conversion

In color image processing, there are various color models in use today. The RGB model is mostly used in hardware oriented application such as color monitor. In the RGB model, images are represented by three components, one for each primary color – red, green and blue. However, RGB color space is not sensitive to human visual perception or statistical analysis. Moreover, a color is not simply formed by these three primary colors. HSV color space is a non-linear transform from RGB color space that can describe perceptual color relationship more accurately than RGB color space. In this paper, HSV color space is adopted.

HSV color space is formed by hue (H), saturation (S) and value (V). Hue denotes the property of color such as blue, green, red, and so on. Saturation denotes the perceived intensity of a specific color. Value denotes brightness perception of a specific color. Thus it can be seen that HSV color space is different from RGB color space in color variations. When a color pixel-value in RGB color space is adjusted, intensities of red channel, green channel, and blue channel of this color pixel are modified. That means color, intensity, and saturation of a pixel is involved in color variations. It is difficult to observe the color variation in complex color environment or content. However, HSV color space separates the color into hue, saturation, and value which means observation of color variation can be individually discriminated. Based on the above the proposed method adopted HSV descriptor for color space because it describes colour intensity and brightness's in a significant manner. In order to transform RGB color space to HSV color space, the transformation is described as follows:

The transformation equations from RGB to HSV color model conversion is given below

$$V = \max(R, G, B) \tag{1}$$

$$S = \frac{V - \min(R,G,B)}{V} \tag{2}$$

$$\mathrm{H} = \frac{\mathrm{G-B}}{6S} \quad if \quad \mathrm{V = R} \tag{3}$$

$$\mathrm{H} = \frac{1}{3} + \frac{\mathrm{B-R}}{6S} \quad if \quad \mathrm{V = G} \tag{4}$$

$$\mathrm{H} = \frac{1}{3} + \frac{\mathrm{R-G}}{6S} \quad if \quad \mathrm{V = B} \tag{5}$$

Where R, G, B are Red, Green and Blue normalized in value [0, 1]. In order to quantize the range of the H plane is normalized with value [0, 255] for extracting features specifically.

b) Fuzzy Texton Matrix Detection

In natural images, due to the presence of noise, different illumination levels and various conversion factors, between neighboring pixels of a window represent as equal, though they rarely have exactly the same intensity value. To avoid this imprecision and be able to represent the vagueness within the processes, the present paper made use of fuzzy logic and fuzzy techniques in deriving fuzzy texton binary matrix for classification of textures. To deal classification effect by different shape components, with regions of natural images perceived as homogeneous by human beings, the present paper proposes a Fuzzy Based Texton Binary Shape Window (FTBSM) encoding.

The present paper labels eight neighbors of a 3×3 neighborhood using five possible fuzzy patterns or values {0, 1, 2, 3 and 4} derived from the fuzzy code as depicted in Equation 6 and the fuzzy membership function is represented as shown in Fig.1. From Fig.1, the element V_i represent the intensity values of the eight neighboring pixels on a 3×3 neighborhood, V_0 represents the intensity value of central pixel, x and y are the user-specified lag values.

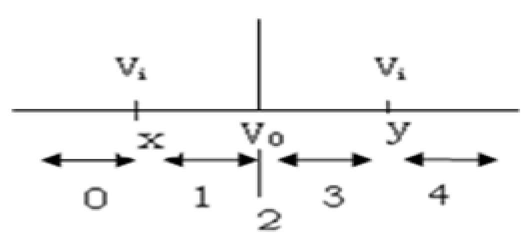

Fig. 1 : Fuzzy texture number (Base-5) representation

$$\mathrm{E_i} = \begin{cases} 0 & \text{if} \quad \mathrm{V_i < V_0} \quad \text{and} \quad \mathrm{V_i} < x \\ 1 & \text{if} \quad \mathrm{V_i < V_0} \quad \text{and} \quad \mathrm{V_i} > x \\ 2 & \text{if} \quad \mathrm{V_i = V_0} \\ 3 & \text{if} \quad \mathrm{V_i > V_0} \quad \text{and} \quad \mathrm{V_i > y} \\ 4 & \text{if} \quad \mathrm{V_i > V_0} \quad \text{and} \quad \mathrm{V_i < y} \end{cases} \quad \text{for } \mathrm{i = 1,2,3,\dots,8} \tag{6}$$

For example, the process of evaluating fuzzy values on a 3×3 neighborhood is shown in Fig.2.

(a)

90	130	145
160	140	200
100	140	250

(b)

0	1	3
4	2	4
0	2	4

(c) FV= {0, 1, 3, 4, 2, 4, 2, 0, 4}

Fig. 2 : Representation of (a) a 3×3 neighborhood (b) fuzzy values (c) set of fuzzy values

c) *Micro Structure Texton Detection*

Textons are fundamental micro structures in texture images and are considered as the atoms of pre-attentive human visual perception [21]. Textons have more powerful description ability than the pixels themselves. Texton are defined as a set of blobs or emergent patterns sharing a common property all over the image [21]. The different textons may form various image features. If the textons in the image are small and the tonal difference between neighbouring textons is large, a fine texture may result. If the textons are larger and concise of several pixels, a coarse texture may result. If the textons in image are large and consists of few texton categories, an obvious shape may result. If the textons are greatly expanded in one orientation, pre-attentive discrimination is somewhat reduced. If elongated elements are not jittered in orientation, the texton gradients at the texture boundaries are increased. To address this, the present study considered fuzzy based texton approach is used for classification of textures. The proposed Fuzzy texton approach utilized to detect micro-structures blocks from left-to-right and top-to-bottom through- out the image.

A fuzzy code is applied for overlapped window of the texton micro-structure for the construction of Fuzzy Texton Binary Shape Matrix (FTBSM). The FTBSM is used for detection of shapes for classification of textures. In a 3×3 block, if one of the eight nearest neighbors has the same value as the center pixel, then it is kept unchanged and marked with green color as shown in Fig.3(c); otherwise set it to '0'. Incase if the centre pixel is zero and one of the eight nearest neighbors has the same value as the center pixel, then these pixel values are also set to '1'. If all the eight nearest neighboring pixels are '0', then the 3×3 block is not considered as a micro structure. The marked pixels are treated as micro-structure and this structure is set to '1'. The working mechanism of proposed fuzzy texton binary matrix method is illustrated in Fig.3.

(a)

51	143	143	143	152	153
151	143	143	143	152	153
155	142	142	138	147	153
157	143	143	135	142	151
157	143	143	135	142	151
154	146	146	140	143	148

(b)

3	2	2	1	2	3
3	2	2	1	2	3
3	0	0	0	1	3
3	1	1	0	2	3
3	1	1	0	2	3
3	2	2	1	4	4

(c)

0	2	2	1	2	0
0	2	2	1	2	0
0	0	0	0	1	0
0	1	1	0	2	0
0	1	1	0	2	0
0	0	0	1	0	0

(d)

0	1	1	1	1	0
0	1	1	1	1	0
0	1	1	1	1	0
0	1	1	1	1	0
0	1	1	1	1	0
0	0	0	1	0	0

Fig. 3 : Illustration of the Fuzzy Texton Binary Matrix (a) Original texture image (b) Detection of fuzzy values (c) Fuzzy texton mapping process on a 3×3 neighborhood d) Fuzzy texton binary image

d) *Fuzzy Texture Features on FTBM*

The present paper evaluated fuzzy texture features for classification of textures based on proposed FTBSM. It consists of a 3×3 neighborhood for evaluating fuzzy shape components. It has derived five different fuzzy shape components named as Diamond, Diagonal, Vertical Line, Horizontal Line and Blob on a 3×3 neighborhood. Each of the fuzzy shape components is represented as shown in Fig.4.

(a)

	0	
0		0
	0	

(b)

0		0
0		0

(c)

0		
0		
0		

(d)

0	0	0

(e)

0	0	0
0		0
0	0	0

Fig. 4 : Representation of fuzzy shape components (a) Diamond(b) Diagonal (c) Vertical Line(d) horizontal Line (e) Blob

For the classification of textures the frequency occurrences of each of the fuzzy shape component with different texture patterns is counted using the Algorithm 1. The novelty of the present work is it uses only five different types of fuzzy shape components using the proposed FTBSM.

Algorithm 1: Classification of textures based on different fuzzy shape components on a proposed FTBSM.

1. Read the original Textures Tk, where k=1:n with dimension N×M.
2. Convert color texture image to gray image by using HSV as explained in Section II.
3. Convert each 3×3 neighborhood of the gray level texture image into a Fuzzy values (0, 1, 2, 3 or 4) by using fuzzy code as explained in Section II.

4. Evaluate fuzzy texton binary matrix using fuzzy code and texton as explained in Section II.
5. Represent the given shape components on a 3×3 neighborhood, where i=1 to 5 as shown in Fig.4.
6. Compute frequency occurrence (FPi) of each shape component by convolving the entire texture image (Tk). Repeat the procedure for all shape components of all texture images.
7. Compute the percentage of occurrence of each shape component (SPi, i=1 to 5) for each of the texture Tk, k=1:24.

$$\forall_{i=1}^{5} sp_i = \frac{FP_i}{(N-2)\times(M-2)} \times 100$$

8. Classify textures using distance function of Step 7.
9. Calculate the average percentage of occurrence (APOi) of each shape component of all textures.

$$APO_i = \frac{\forall_{i=1}^{5} \sum_{k=1}^{n} T_k sp_i}{max(k)}$$

10. A texture Tk will be placed in one of the two classes C1 or C2 in the following way.

for T_k, k=1:n
begin
if ($\forall_{i=1}^{5} T_k sp_i == 1$) T_k is assigned to C_1
else T_k is assigned to C_2
end

III. Results and Discussions

To evaluate a good classification based on the fuzzy shape components, the present study initially computed the frequency occurrences of each shape component. The proposed methodology is tested with a set of different groups of textures as shown in Fig.6. The frequency occurrences of the derived fuzzy shape components are counted for all the original textures and the results are furnished in Table 1.

Table 1 : Frequency occurrences of fuzzy shape components on a 3×3 neighborhood of different groups of textures

Texture Name	Diamond	Diagonal	Horizontal Line	Vertical Line	Blob
Brick1	103	56	196	153	165
Brick2	298	187	498	270	338
Brick3	405	279	520	344	440
Brick4	41	20	130	61	85
Brick5	74	114	163	178	150
Brick6	476	259	627	312	569
Brick7	141	238	228	349	197
Brick8	165	465	260	556	225
Granite1	50	59	143	154	117
Granite2	99	31	185	104	168
Granite3	32	75	111	177	85
Granite4	43	7	157	40	147
Granite5	52	25	158	71	144
Granite6	2	2	26	7	16
Granite7	28	41	107	118	89
Granite8	137	91	224	195	219
Marble1	19	5	56	21	54
Marble2	12	19	79	54	80
Marble3	133	113	280	185	237
Marble4	62	57	185	153	155
Marble5	89	663	109	849	100
Marble6	99	400	142	552	140
Marble7	320	199	492	258	472
Marble8	300	148	465	214	422
Mosaic1	182	372	250	448	268
Mosaic2	508	412	592	433	575
Mosaic3	401	409	507	423	501
Mosaic4	441	366	542	393	533
Mosaic5	352	25	561	81	485
Mosaic6	322	19	546	70	471
Mosaic7	331	297	426	444	391
Mosaic8	344	337	475	408	422

From the results of Table 1, texture classification can be done by distance function. By using distance function, two textures are similar count the number of textures and the result are stored in the training database. The present study, classified textures based on the proposed method using distance function with a lag value. The distance among all groups of textures based on number of frequency occurrences of different shape components are calculated and are furnished in Table 2. The distance measure of different groups of textures is tabulated in Table 2, Table 3, Table 4 and Table 5 respectively. The classification group of textures with lag value for all textures is shown in Table 6, Table 7 and Table 8 and Table 9 respectively.

Table 2 : Distance measure of five fuzzy shape components of Brick group of textures

Diamond	Brick1	Brick2	Brick3	Brick4	Brick5	Brick6	Brick7	Brick8
Brick1	0	195	302	62	29	373	38	62
Brick2		0	107	257	224	178	157	133
Brick3			0	364	331	71	264	240
Brick4				0	33	435	100	124
Brick5					0	402	67	91
Brick6						0	335	311
Brick7							0	24
Brick8								0
Diagonal	Brick1	Brick2	Brick3	Brick4	Brick5	Brick6	Brick7	Brick8
Brick1	0	131	223	36	58	203	182	409
Brick2		0	92	167	73	72	51	278
Brick3			0	259	165	20	41	186
Brick4				0	94	239	218	445
Brick5					0	145	124	351
Brick6						0	21	206
Brick7							0	217
Brick8								0
Horizontal Line	Brick1	Brick2	Brick3	Brick4	Brick5	Brick6	Brick7	Brick8
Brick1	0	302	324	66	33	431	32	64
Brick2		0	22	368	335	129	270	238
Brick3			0	390	357	107	292	260
Brick4				0	33	497	98	130
Brick5					0	464	65	97
Brick6						0	399	367
Brick7							0	32
Brick8								0
Vertical Line	Brick1	Brick2	Brick3	Brick4	Brick5	Brick6	Brick7	Brick8
Brick1	0	117	191	92	25	159	196	403
Brick2		0	74	209	92	42	79	286
Brick3			0	283	166	32	5	212
Brick4				0	117	251	288	495
Brick5					0	134	171	378
Brick6						0	37	244
Brick7							0	207
Brick8								0
Blob	Brick1	Brick2	Brick3	Brick4	Brick5	Brick6	Brick7	Brick8
Brick1	0	173	275	80	15	404	32	60
Brick2		0	102	253	188	231	141	113
Brick3			0	355	290	129	243	215
Brick4				0	65	484	112	140
Brick5					0	419	47	75
Brick6						0	372	344
Brick7							0	28
Brick8								0

Table 3 : Distance measure of five fuzzy shape components of Granite group of textures

Diamond	Granite1	Granite2	Granite3	Granite4	Granite5	Granite6	Granite7	Granite8
Granite1	0	49	18	7	2	48	22	87
Granite2		0	67	56	47	97	71	38
Granite3			0	11	20	30	4	105
Granite4				0	9	41	15	94
Granite5					0	50	24	85
Granite6						0	26	135
Granite7							0	109
Granite8								0
Diagonal	Granite1	Granite2	Granite3	Granite4	Granite5	Granite6	Granite7	Granite8
Granite1	0	28	16	52	34	57	18	32
Granite2		0	44	24	6	29	10	60
Granite3			0	68	50	73	34	16
Granite4				0	18	5	34	84
Granite5					0	23	16	66
Granite6						0	39	89
Granite7							0	50
Granite8								0
Horizontal Line	Granite1	Granite2	Granite3	Granite4	Granite5	Granite6	Granite7	Granite8
Granite1	0	42	32	14	15	117	36	81
Granite2		0	74	28	27	159	78	39
Granite3			0	46	47	85	4	113
Granite4				0	1	131	50	67
Granite5					0	132	51	66
Granite6						0	81	198
Granite7							0	117
Granite8								0
Vertical Line	Granite1	Granite2	Granite3	Granite4	Granite5	Granite6	Granite7	Granite8
Granite1	0	50	23	114	83	147	36	41
Granite2		0	73	64	33	97	14	91
Granite3			0	137	106	170	59	18
Granite4				0	31	33	78	155
Granite5					0	64	47	124
Granite6						0	111	188
Granite7							0	77
Granite8								0
Blob	Granite1	Granite2	Granite3	Granite4	Granite5	Granite6	Granite7	Granite8
Granite1	0	51	32	30	27	101	28	102
Granite2		0	83	21	24	152	79	51
Granite3			0	62	59	69	4	134
Granite4				0	3	131	58	72
Granite5					0	128	55	75
Granite6						0	73	203
Granite7							0	130
Granite8								0

Table 4 : Distance measure of five fuzzy shape components of Marble group of textures

Diamond	Marble1	Marble2	Marble3	Marble4	Marble5	Marble6	Marble7	Marble8
Marble1	0	7	114	43	70	80	301	281
Marble2		0	121	50	77	87	308	288
Marble3			0	71	44	34	187	167
Marble4				0	27	37	258	238
Marble5					0	10	231	211
Marble6						0	221	201
Marble7							0	20
Marble8								0
Diagonal	Marble1	Marble2	Marble3	Marble4	Marble5	Marble6	Marble7	Marble8
Marble1	0	14	108	52	658	395	194	143
Marble2		0	94	88	644	381	180	129
Marble3			0	56	550	287	86	35
Marble4				0	606	343	142	91
Marble5					0	263	464	515
Marble6						0	201	252
Marble7							0	51
Marble8								0
Horizontal line	Marble1	Marble2	Marble3	Marble4	Marble5	Marble6	Marble7	Marble8
Marble1	0	23	224	129	53	86	436	409
Marble2		0	201	106	30	63	413	386
Marble3			0	95	171	138	212	185
Marble4				0	76	43	307	280
Marble5					0	33	383	356
Marble6						0	350	323
Marble7							0	27
Marble8								0
Vertical Line	Marble1	Marble2	Marble3	Marble4	Marble5	Marble6	Marble7	Marble8
Marble1	0	33	164	132	828	531	237	193
Marble2		0	131	99	795	498	204	160
Marble3			0	32	664	367	73	29
Marble4				0	696	399	105	61
Marble5					0	297	591	635
Marble6						0	294	338
Marble7							0	44
Marble8								0
Blob	Marble1	Marble2	Marble3	Marble4	Marble5	Marble6	Marble7	Marble8
Marble1	0	26	183	101	46	86	418	368
Marble2		0	157	75	20	60	392	342
Marble3			0	82	137	97	235	185
Marble4				0	55	15	317	267
Marble5					0	40	372	322
Marble6						0	332	282
Marble7							0	50
Marble8								0

Table 5 : Distance measure of five fuzzy shape components of Mosaic group of textures

Diamond	Mosaic1	Mosaic2	Mosaic3	Mosaic4	Mosaic5	Mosaic6	Mosaic7	Mosaic8
Mosaic1	0	326	219	259	170	140	149	162
Mosaic2		0	107	67	156	186	177	164
Mosaic3			0	40	49	79	70	57
Mosaic4				0	89	119	110	97
Mosaic5					0	30	21	8
Mosaic6						0	9	22
Mosaic7							0	13
Mosaic8								0
Diagonal	Mosaic1	Mosaic2	Mosaic3	Mosaic4	Mosaic5	Mosaic6	Mosaic7	Mosaic8
Mosaic1	0	40	37	6	347	353	75	35
Mosaic2		0	3	46	387	393	115	75
Mosaic3			0	43	384	390	112	72
Mosaic4				0	341	347	69	29
Mosaic5					0	6	272	312
Mosaic6						0	278	337
Mosaic7							0	46
Mosaic8								0
Horizontal Line	Mosaic1	Mosaic2	Mosaic3	Mosaic4	Mosaic5	Mosaic6	Mosaic7	Mosaic8
Mosaic1	0	342	257	292	311	296	176	225
Mosaic2		0	85	50	31	46	166	117
Mosaic3			0	35	54	39	81	32
Mosaic4				0	19	4	116	67
Mosaic5					0	15	135	86
Mosaic6						0	120	71
Mosaic7							0	49
Mosaic8								0
Vertical Line	Mosaic1	Mosaic2	Mosaic3	Mosaic4	Mosaic5	Mosaic6	Mosaic7	Mosaic8
Mosaic1	0	15	25	55	367	378	4	40
Mosaic2		0	10	40	352	363	11	25
Mosaic3			0	30	342	353	21	15
Mosaic4				0	312	323	51	15
Mosaic5					0	11	363	327
Mosaic6						0	374	338
Mosaic7							0	36
Mosaic8								0
Blob	Mosaic1	Mosaic2	Mosaic3	Mosaic4	Mosaic5	Mosaic6	Mosaic7	Mosaic8
Mosaic1	0	307	233	265	217	203	123	154
Mosaic2		0	74	42	90	104	184	153
Mosaic3			0	32	16	30	110	79
Mosaic4				0	48	62	142	111
Mosaic5					0	14	94	63
Mosaic6						0	80	49
Mosaic7							0	31
Mosaic8								0

Table 6 : Classes of textures for the proposed method using lag value of Diamond shape component

Texture Group	Class	Classified Textures
Brick	C_1	{Brick1, Brick2, Brick3, Brick5, Brick7, Brick8}
	C_2	{Brick4, Brick6}
Granite	C_1	{Granite1, Granite2, Granite3, Granite4, Granite5, Granite7, Granite8}
	C_2	{Granite6}
Marble	C_1	{Marble3, Marble4, Marble6, Marble7, Marble8}
	C_2	{Marble1, Marble2, Marble5}
Mosaic	C_1	{Mosaic1, Mosaic2, Mosaic3, Mosaic4, Mosaic7, Mosaic8}
	C_2	{Mosaic5, Mosaic6}

Table 7 : Classes of textures for the proposed method using lag value of Diagonal shape component

Texture Group	Class	Classified Textures
Brick	C_1	{Brick1, Brick2, Brick3, Brick4, Brick5, Brick6, Brick7, Brick8}
	C_2	----------
Granite	C_1	{Granite1, Granite2, Granite3, Granite4, Granite5, Granite6, Granite7, Granite8}
	C_2	----------
Marble	C_1	{ Marble1, Marble2, Marble3, Marble4, Marble5, Marble6, Marble7, Marble8}
	C_2	----------
Mosaic	C_1	{Mosaic3, Mosaic4, Mosaic5, Mosaic6, Mosaic7, Mosaic8}
	C_2	{Mosaic1, Mosaic2}

Table 8 : Classes of textures for the proposed method using lag value of Horizontal Line shape component

S	Class	Classified Textures
Brick	C_1	{Brick1, Brick2, Brick3, Brick5, Brick7, Brick8}
	C_2	{Brick4, Brick6}
Granite	C_1	{Granite1, Granite2, Granite4, Granite5, Granite6, Granite7, Granite8}
	C_2	{Granite3}
Marble	C_1	{ Marble4, Marble6, Marble8}
	C_2	{Marble1, Marble2, Marble3, Marble5, Marble7}
Mosaic	C_1	{ Mosaic1, Mosaic2, Mosaic3, Mosaic4, Mosaic7, Mosaic8}
	C_2	{Mosaic5, Mosaic6}

Table 9 : Classes of textures for the proposed method using lag value of Vertical Line shape component

Texture Group	Class	Classified Textures
Brick	C_1	{Brick5, Brick7, Brick8}
	C_2	{Brick1, Brick2, Brick3, Brick4, Brick6}
Granite	C_1	{Granite4, Granite5, Granite7, Granite8}
	C_2	{Granite1, Granite2, Granite3, Granite6}
Marble	C_1	{Marble1, Marble2, Marble3, Marble4, Marble6, Marble8}
	C_2	{Marble5, Marble7}
Mosaic	C_1	{Mosaic2, Mosaic3, Mosaic4, Mosaic7, Mosaic8}
	C_2	{Mosaic1, Mosaic5, Mosaic6}

Table 10 : Classes of textures for the proposed method using lag value of Blob shape component

Texture Group	Class	Classified Textures
Brick	C_1	{Brick2, Brick3, Brick4, Brick5, Brick7, Brick8}
	C_2	{Brick1, Brick6}
Granite	C_1	{Granite1, Granite2, Granite4, Granite5, Granite6, Granite7, Granite8}
	C_2	{Granite3}
Marble	C_1	{ Marble2,Marble3, Marble4, Marble6, Marble8}
	C_2	{Marble1, Marble5, Marble7}
Mosaic	C_1	{Mosaic3, Mosaic7, Mosaic8}
	C_2	{Mosaic1, Mosaic2, Mosaic4, Mosaic5, Mosaic6}

By observing the results of Tables 6 to 10 the following facts are noted down. Table 7 clearly indicates that, it shows a uniform distance between each of them. The following facts are noted down from the classification tables of Table 6 to Table 10.

- The extracted diamond shape component on the FTBSM of Table 6 classified each of the Brick, Granite, Marble and Mosaic textures into two classes.
- The extracted diagonal shape component on the FTBSM of Table 7 classified each of the Brick, Granite and Marble textures into separate class only, and it classified the mosaic textures into two classes
- The extracted horizontal line shape component on the FTBSM of Table 8 classified each of the Brick, Granite, Marble and Mosaic textures into two classes.
- The extracted vertical line shape component on the FTBSM of Table 9 classified each of the Brick, Granite, Marble and Mosaic textures into two classes.
- The extracted blob shape component on the FTBSM of Table 10 classified each of the Brick, Granite, Marble and Mosaic textures into two classes.

The facts indicate that a good, precise and accurate stone classification is observed by the proposed FTBSM using diagonal shape components. The proposed method FTBSM also analyzed the percentage occurrence of each shape component represented in the Table 11. The Table 11 evaluated on FTBSM reveals that diagonal shape component classifies brick, granite and marble texture images accurately.

Table 11 : Percentage occurrences of each shape component with every group of textures

Shape Component	Brick	Granite	Marble	Mosaic	Average
Diamond	75	87.5	65	75	75.63
Diagonal	100	100	100	75	93.75
Horizontal Line	75	88	65	75	75.75
Vertical Line	65	50	75	65	63.75
Blob	75	88	65	65	73.25

IV. Conclusions

The present study created a new direction for classification of textures based on texture features derived from shape components on a 3×3 neighborhood. By investigating texture classification using different shape components with fuzzy logic the present study concludes that diagonal shape component contains more classification information than other shape components. Based on the experimental results the proposed FTBSM method concludes that one need not necessarily count the other shape components except the diagonal shape. Therefore the present study reduced a lot of complexity in the selection of shape components for classification purpose.

V. Acknowledgment

The authors would like to express their gratitude to Sri K.V.V. Satyanarayana Raju, Chairman, and Sri K. Sasi Kiran Varma, Managing Director, Chaitanya group of Institutions for encouraging to work at SRRF-GIET Advanced labs.

References Références Referencias

1. Dash M., H. Liu, Feature Selection for Classification. Intelligent Data Analysis, Elsevier, 1997, pp. 131–156.
2. Koller D., M. Sahami, Toward optimal feature selection, in: Proceedings of the 13th International Conference on Machine Learning, Bari, Italy, 1996, pp. 284–292.
3. Zhang P., J. Peng, B. Buckles, Learning optimal filter representation for texture classification, in: International Conference on Pattern Recognition, vol. 2, Hong Kong, 2006, pp. 1138–1141.
4. Reed T.R., J.M.H. du Buf, A review of recent texture segmentation and feature extraction techniques, CVGIP: Image Understanding 57 (1993) 359–372.
5. Tuceryan M., A.K. Jain, Texture analysis, in: C.H. Chen, L.F. Pau, P.S.P. Wang (Eds.), The Handbook of Pattern Recognition and Computer Vision, second ed., World Scientific, Singapore, 1998, pp. 207–248. \
6. Samal A., J.R. Brandle, D.S. Zhang, Texture as the basis for individual tree identification, Information Sciences 176 (2006) 565–576.
7. Raju U S N, B Eswar Reddy, V Vijaya Kumar and B Sujatha "Texture Classification Based On Extraction Of Skeleton Primitives Using Wavelets", Information Technology Journal (7) 6, 2008, pages: 883-889.
8. Vijaya Kumar. V, U S N Raju, K Chandra Sekaran, V V Krishna, "Employing Long Linear Patterns for Texture Classification relying on Wavelets", ICGST-GVIP Journal, ISSN 1687-398X, Volume (8), Issue (V), January 2009, pages: 13-21
9. Jafari-Khouzani K., H. Soltanian-Zadeh, Radon transform orientation estimation for rotation invariant texture analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (6) (2005) 1004–1008.
10. Lee W.-L., Yung-C. Chen, Ying-C. Chen, K.-S. Hsieh, Unsupervised segmentation of ultrasonic liver images by multiresolution fractal feature vector, Information Sciences 175 (2005) 177–199.
11. Ojala T., M. Pietikäinen, D. Harwood, A comparative study of texture measures with classification based

on feature distribution, Pattern Recognition 29 (1996) 51–59.

12. Ojala T., M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (7) (2002) 971–987.
13. Pietikäinen M., T. Ojala, Z. Xu, Rotation-invariant texture classification using feature distribution, Pattern Recognition 33 (2000) 43–52.
14. Ahonen T., A. Hadid, M. Pietikäinen, Face description with local binary patterns: application to face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (12) (2006) 2037–2041.
15. He L.H., C.R. Zou, L. Zhao, D. Hu, An enhanced LBP feature based on facial expression recognition, in: Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai, 2005, pp. 3300–3303.
16. Lahdenoja. O, M. Laiho, A. Paasio, Reducing the feature vector length local binary pattern based face recognition, in: ICIP 2005, IEEE International Conference on Publication, vol. 2, 2005, pp. II-914–II-917.
17. Garcia M.A., D. Balu, Supervised texture classification by integration of multiple texture methods and evaluation windows, Image and Vision Computing 25 (2007) 1091–1106.
18. Marzabal A., C. Torrens, A. Grau, Textured-based characterization of defects in automobile engine valves, in: Proceedings of the Ninth Symposium on Pattern Recognition and Image Processing, Ed. Univ. Jaume I, 2001, pp. 267–272.
19. Paclik P., R. Duin, G. van Kempen, R. Kohlus, Supervised segmentation of textures in backscatter images, in: Proceedings of the 16th International Conference on Pattern Recognition, Quebec City, 2002, pp. 490–493.
20. Lucieer A., A. Stein, P. Fisher, Multivariate texture-based segmentation of remotely sensed imagery for extraction of objects and their uncertainty, International Journal of Remote Sensing 26 (14) (2005) 2917–2936.
21. Eswara Reddy B., A. Nagaraja Rao, A. Suresh, V. Vijaya Kumar “Texture Classification by Simple Patterns on Edge Direction Movements”, IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.11, November 2007, pages: 221-225.
22. Ahonen T., Hadid A. and Pietikainen M., “Face Recognition with Local Binary Patterns,” Computer Vision, ECCV Proceedings, pp. 469-481, 2004.
23. Ahonen T., Pietikainen M., Hadid A. and Maenpaa T., “Face Recognition Based on the Appearance of Local Regions,” 17th International Conference on Pattern Recognition III: pp. 153-156, 2004.
24. Feng X., Hadid A. and Pietikainen M., “A Coarse-to-Fine Classification Scheme for Facial Expression Recognition,” Image Analysis and Recognition, ICIAR 2004 Proceedings, Lecture Notes in Computer Science 3212 II: pp. 668-675, 2004.
25. Feng X., Pietikainen M. and Hadid A., “Facial Expression Recognition with Local Binary Patterns and Linear Programming,” Pattern Recognition and Image Analysis 15 pp. 550-552, 2005.
26. Hadid A., Pietikainen M. and Ahonen T., “A Discriminative Feature Space for Detecting and Recognizing Faces,” IEEE Conference on Computer Vision and Pattern Recognition II: pp. 797-804, 2004.
27. Galloway M. M., “Texture analysis using gray level run lengths,” Comput. Graphics Image Process., vol. 4, pp. 172–179, June 1975.
28. Wiselin Jiji G., Ganesan L., “A new approach for unsupervised segmentation,” Applied Soft Computing, vol.10, pp: 689-693, 2010.
29. Wiselin Jiji G., Ganesan L., “Comparative analysis of colour models for colour textures based on feature extraction,” Int. Jour. of Soft computing, vol.2(3), pp:361-366, 2007.
30. Chu A., Sehgal C.M.A. and Greenleaf J.F., (1990), Use of gray value distribution of run lengths for texture analysis, Pattern Recognition Letters, Volume 11(6), Pages 415-419
31. Belur V. Dasarathy and Edwin B. Holder, Image characterizations based on joint gray level—run length distributions. (1981), Pattern Recognition Letters, Volume 12 (, Issue 8, August 1991, Pages 497-502.
32. Faisal Ahmed, A.S.M. Hossain Bari, Emam Hossain, “Compound Local Binary Pattern (CLBP) for Facial Expression Recognition”, ivcnz2011
33. Yung-Kaun Chan, Chin-Chen Chang, “Image matching using run-length feature”, Pattern Recognition Letters, Volume 22(2001), Pages 447-455.