

# 九州工業大学学術機関リポジトリ



Title	Study on Co-occurrence-based Image Feature Analysis and Texture Recognition Employing Diagonal-Crisscross Local Binary Pattern
Author(s)	Hossain, Shahera
Issue Date	2013-06
URL	<a href="http://hdl.handle.net/10228/5282">http://hdl.handle.net/10228/5282</a>
Rights	

**Study on Co~occurrence~based Image Feature  
Analysis and Texture Recognition Employing  
Diagonal~Crisscross Local Binary Pattern**

**Shahera Hossain**

**Faculty of Engineering  
Department of Electrical Engineering and Electronics  
Kyushu Institute of Technology  
JAPAN**



# **THESIS**

**Study on Co-occurrence-based Image Feature Analysis and  
Texture Recognition Employing Diagonal-Crisscross Local  
Binary Pattern**

*Submitted in partial fulfilment of the requirements for the degree of*  
**Doctor of Philosophy**

*by*

**Shahera Hossain**

Student ID: 08349450

*Supervised by*

**Professor Seiichi Serikawa**

Faculty of Engineering  
Department of Electrical Engineering and Electronics  
Kyushu Institute of Technology  
JAPAN

June 2013

# **DOCTORAL THESIS**

**Study on Co-occurrence-based Image Feature Analysis and  
Texture Recognition Employing Diagonal-Crisscross Local  
Binary Pattern**

By  
**Shahera Hossain**  
(Student ID: 08349450)

Supervisor:  
**Professor Seiichi Serikawa**

I certify that I have read this thesis and that in my opinion; it is fully adequate in scope and in quality as a thesis for the degree of Doctor of Philosophy.

---

**Professor Dr. Seiichi Serikawa**  
Faculty of Engineering  
Department of Electrical Engineering and Electronics  
Kyushu Institute of Technology, Japan

**June 2013**



---

# TABLE OF CONTENT

---

<b>Table of Content</b>	.....	i
<b>List of Figures</b>	.....	v
<b>List of Tables</b>	.....	viii
<b>Abstract</b>	.....	ix
<b>Acknowledgement</b>	.....	x
<b>List of Publications</b>	.....	xi

## CHAPTER 1

### INTRODUCTION

1.1 Introduction	.....	1
1.2 Motivation	.....	3
1.3 Thesis Outline	.....	4

## CHAPTER 2

### FEATURES FOR TEXTURE ANALYSIS

2.1 Introduction	.....	5
2.2 Various Applications	.....	8
2.2.1 <i>Medical imaging</i>	.....	8
2.2.2 <i>Remote sensing</i>	.....	8
2.2.3 <i>Defect detection</i>	.....	8
2.3.4 <i>Others</i>	.....	9
2.3 Different Approaches for Texture Analysis	.....	9
2.3.1 <i>Statistical Approaches</i>	.....	9
2.3.2 <i>Structural Approaches</i>	.....	14
2.3.3 <i>Model-based Approaches</i>	.....	15
2.3.4 <i>Filter-based approach/Transform</i>	.....	16
2.4 Conclusions	.....	18

## CHAPTER 3

### IN-DEPTH ANALYSIS OF TEXTURE DATABASES

3.1	Introduction	.....	20
3.2	Texture Databases in Medical Image Processing	.....	20
3.2.1	<i>MRI brain database</i>	.....	20
3.2.2	<i>USF database</i>	.....	21
3.2.3	<i>Others in medical analysis</i>	.....	21
3.3	Natural Texture Image Database	.....	22
3.3.1	<i>Brodatz texture database</i>	.....	22
3.3.2	<i>Vision texture database (VisTex)</i>	.....	24
3.3.3	<i>USC-SIPI texture mosaic database</i>	.....	25
3.3.4	<i>Texture library</i>	.....	26
3.4	Texture of Materials	.....	27
3.4.1	<i>Meastex database</i>	.....	27
3.4.2	<i>PhoTex database</i>	.....	28
3.4.3	<i>PhoTex 3D database</i>	.....	29
3.4.4	<i>ALOT database</i>	.....	30
3.4.5	<i>UMD dataset</i>	.....	32
3.4.6	<i>OUTex database</i>	.....	32
3.4.7	<i>CUReT database</i>	.....	32
3.4.8	<i>Textile database</i>	.....	35
3.4.9	<i>UIUC database</i>	.....	35
3.4.10	<i>CMU NRT database</i>	.....	36
3.4.11	<i>Rutgers skin texture database</i>	.....	37
3.4.12	<i>KTH-TIPS database</i>	.....	38
3.4.13	<i>KTH-TPIS2 material database</i>	.....	39
3.4.14	<i>PerTex database</i>	.....	40
3.4.15	<i>Building texture database</i>	.....	40
3.4.16	<i>Grain mixtures dataset</i>	.....	40
3.4.17	<i>Kylberg texture dataset</i>	.....	40
3.5	Dynamic Texture Database	.....	40
3.5.1	<i>UCLA dynamic texture database</i>	.....	41
3.5.2	<i>UCLA-pan database</i>	.....	42
3.5.3	<i>DynTex database</i>	.....	42
3.5.4	<i>Saisan's dynamic texture database</i>	.....	43

3.5.5	<i>DynTex++ database</i>	.....	44
3.5.6	<i>MIT Szummer dataset</i>	.....	45
3.6	Discussions	.....	45
3.7	Conclusions	.....	50

## CHAPTER 4

### ANALYSIS OF FEATURES BASED ON CO-OCCURRENCE IMAGE MATRIX

4.1	Introduction	.....	52
4.2	Co-occurrence Matrix-based Image Features	.....	53
4.3	Experimental Results and Analysis	.....	62
4.4	Conclusions	.....	72

## CHAPTER 5

### CATEGORIZATION OF FEATURES BASED ON CO-OCCURRENCE IMAGE MATRIX

5.1	Introduction	.....	74
5.2	Development of Multiclass Database	.....	74
5.3	Comparisons among Features	.....	76
5.3.1	<i>ASM vs. IDM features</i>	.....	76
5.3.2	<i>Contrast vs. entropy features</i>	.....	78
5.3.3	<i>Contrast vs. variance features</i>	.....	80
5.3.4	<i>Entropy vs. sum entropy features</i>	.....	80
5.3.5	<i>Correlation feature</i>	.....	81
5.3.6	<i>Variance vs. sum variance features</i>	.....	82
5.3.7	<i>Sum average vs. IMC1 vs. IMC2 features</i>	.....	83
5.4	Group-wise Analysis based on Key Features	.....	84
5.5	Category of Features based on Angular Directions	.....	87
5.5.1	<i>Rotation-invariant features</i>	.....	88
5.5.2	<i>Almost rotation-invariant features</i>	.....	88
5.5.3	<i>Rotation-dependent features</i>	.....	89
5.6	Conclusions	.....	90

## CHAPTER 6

### TEXTURE RECOGNITION BASED ON DIAGONAL-CRISSCROSS LOCAL BINARY PATTERN

6.1	Introduction	.....	92
6.2	Computation of Local Binary Pattern	.....	93
6.3	Diagonal-Crisscross Local Binary Pattern	.....	96
6.4	Median-RILBP	.....	99

6.5 Mean-RILBP	.....	100
6.6 Experimental Setup	.....	101
6.7 Recognition Strategy	.....	103
6.8 Result and Analysis	.....	106
6.9 Conclusions	.....	107
 <b>CHAPTER 7</b>		
<b>CONCLUSIONS AND FUTURE WORK</b>		
7.1 Summary of the Thesis	.....	109
7.2 Outline on Future Work	.....	110
 <b>REFERENCES</b>	.....	111

# List of Figures

---

Title	Page No.
Figure 1.1: Example of natural textures	1
Figure 1.2: Example of man-made textures	2
Figure 2.1: Three different texture images	5
Figure 2.2: Different stages for texture analysis	6
Figure 2.3: Examples of images from various application in which texture analysis is important	7
Figure 2.4: Statistical based feature extraction level	10
Figure 2.5: Model-based feature extraction	15
Figure 2.6: Filter-based feature extraction	16
Figure 3.1: Natural Texture Image Dataset: Four sample images of (a) Brodatz texture set; (b) VisTex dataset; (c) Texture library dataset	26
Figure 3.2: Four bodies of (a) Meastex dataset; (b) OUTex dataset	28
Figure 3.3: Some sample images from the Amsterdam ALOT database (ALOT class numbers are shown; Row#1: tea-wafers (9), brown bread (26), cotton (43), terry cloth (48), punched plastic (56); Row#2: cork (57), cotton (60), ribbed cotton (64), sponge (176), chamois (196)	30
Figure 3.4: Example of 25 classes each from: (a) UMD dataset; (b) UIUC database	31
Figure 3.5: Example of CUReT database: ‘white-bread’ image in three different poses and illuminations	33
Figure 3.6: Some sample images from: (a) KTH-TIPS dataset; (b) KTH-TIPS2 material database; (c) An example that depicts the KTH-TIPS dataset’s variations in terms of scale, pose and illumination	39
Figure 3.7: (a) DynTex database – few sample frames; (b) MIT temporal texture database – few frames from ‘smoke’	44

Title	Page No.
Figure 4.1: An example of the computation of gray-level co-occurrence matrix from (a). Sub-Tables sequentially presents the computation of the elements of the co-occurrence matrix for the 4x4 image, as in (a).	52
Figure 4.2: An image patch of size of 4x4 and the corresponding co-occurrence matrix as shown in the mid-table. A normalized matrix is shown here which can be computed after dividing each value by the total co-occurrence values	53
Figure 4.3: Eight different directions of adjacency for the calculation of co-occurrence matrices	53
Figure 4.4: Standard dataset of twenty-one texture images in gray-scale.	60
Figure 4.5: ASM for four angles: It is noticeable that the ASM value is higher for img#1 to img#7 than others. This is due to their uniform texture elements and the pixels are very similar in parts; whereas, img#19~21 has lowest value due to varied texture patterns.	61
Figure 4.6: Contrast evaluation for four angles: Contrast is zero when the neighboring pixels have constant values.	62
Figure 4.7: Entropy feature: more with varied pixel-cases and vice-versa.	63
Figure 4.8: Variance for four angles: Similar distribution of entropy.	63
Figure 4.9: The corresponding correlation values for four directions of images, as shown in Table 4.1.	65
Figure 4.10: Inverse Difference Moment for four angles: the IDM has higher value when all elements of the image are same.	66
Figure 4.11: Sum average feature.	66
Figure 4.12: Sum variance feature for four angular orientations.	67
Figure 4.13: Sum entropy for four angles: Note that F7~F9 depict varied nature for images with varied pixel positions.	67
Figure 4.14: Information Measures of Correlation – 1	67
Figure 4.15: IMC-2: It is evident that the IMC-2 is not similar to IMC-1 though the variations look alike a bit but the variation ranges are different	71

Title	Page No.
Figure 5.1: Sample images from our multiclass database.	76
Figure 5.2: Angular second moment VS. Inverse different moment features	77
Figure 5.3: Graphs to compare the characteristics of contrast and entropy features.	79
Figure 5.4: Contrast vs. Variance features	80
Figure 5.5: Entropy vs. Sum entropy features	81
Figure 5.6: Correlation variations for multi-class dataset.	81
Figure 5.7: variance vs. sum variance comparison	82
Figure 5.8: Sum average vs. IMC1 vs. IMC2 comparison	83
Figure 5.9: Seven key features for different groups of classes	85
Figure 5.10: Seven key features for different groups of classes: (a) forest vs. tree class; (b) wall vs. road class.	85
Figure 5.11: Seven key features for some natural image classes.	86
Figure 5.12: Seven key features for different groups of classes: (a) some material images; (b) some texture-based classes.	86
Figure 5.13: Graphs to demonstrate the rotation-invariant features.	87
Figure 5.14: Graphs to demonstrate the <i>almost</i> rotation-invariant features.	88
Figure 5.15: Graphs to demonstrate the rotation-variant features.	89
Figure 6.1: Calculating the basic local binary pattern for a 3x3 block of an image.	93
Figure 6.2: Another example of LBP computation.	94
Figure 6.3: LBP for 3x3 and 5x5 blocks for 8 neighborhood pixels.	95
Figure 6.4: Computation of DCLBP	97
Figure 6.5: Sample images for each class of USC-SIPI Rotated Textures dataset	101
Figure 6.6: Seven rotated images for bark class (USC-SIPI Rotated Textures dataset)	101
Figure 6.7: KTH-TIPS dataset	102
Figure 6.8: Ten different classes for KTH-TIPS dataset	102
Figure 6.9: Example of <i>k</i> -NN classifier	103

# List of Tables

---

Table Title	Page No.
<b>Table 3.1:</b> Various properties of important texture databases	46
<b>Table 3.2:</b> Database and respective websites or comments	47
<b>Table 3.3:</b> Comparison among some photometric texture databases	48
<b>Table 3.4:</b> Comparisons in terms of illumination and camera/sensors	48
<b>Table 4.1:</b> Correlation values for some symmetric or mirror-type images	68
<b>Table 4.2:</b> Some values for IMC-1 feature. The corresponding images are shown here. Note that borders for each image are provided here only for better visibility. In the calculation, there were no borders for these images.	71
<b>Table 5.1:</b> Proposed categories of features based on importance	84
<b>Table 6.1:</b> Recognition results for USC-SIPI Rotated Textures dataset	106
<b>Table 6.2:</b> Recognition results for KTH-TIPS database	107

# ABSTRACT

---

*In this thesis, we focus on several important fields on real-world image texture analysis and recognition. We survey various important features that are suitable for texture analysis. Apart from the issue of variety of features, different types of texture datasets are also discussed in-depth. There is no thorough work covering the important databases and analyzing them in various viewpoints. We persuasively categorize texture databases – based on many references. In this survey, we put a categorization to split these texture datasets into few basic groups and later put related datasets. Next, we exhaustively analyze eleven second-order statistical features or cues based on co-occurrence matrices to understand image texture surface. These features are exploited to analyze properties of image texture. The features are also categorized based on their angular orientations and their applicability. Finally, we propose a method called diagonal-crisscross local binary pattern (DCLBP) for texture recognition. We also propose two other extensions of the local binary pattern. Compare to the local binary pattern and few other extensions, we achieve that our proposed method performs satisfactorily well in two very challenging benchmark datasets, called the KTH-TIPS (Textures under varying Illumination, Pose and Scale) database, and the USC-SIPI (University of Southern California – Signal and Image Processing Institute) Rotations Texture dataset.*

# Acknowledgement

---

I would like to offer my utmost gratitude and gratefulness to my academic supervisor, **Professor Seiichi Serikawa** for his tremendous support and guidance throughout my studentship under his supervision. He helped me a lot and without his support and advice, it would not be possible for me to continue my study and research. I sincerely thank him from the bottom of my heart.

I am deeply thankful to **Professor Seiji Ishikawa** for introducing him as my supervisor and helped me for admission. Without that support, it would not be easily possible for me to enter for ME and PhD program at this university. I can never forget their helps and supports.

I specially thank to my PhD thesis examiners **Professor Seiichi Serikawa**, **Professor Seiji Ishikawa**, **Professor Hiroshi Maeda** and **Professor Takeshi Ikenaga** for their cordial supports and comments.

I thank all my teachers who supported me and taught me.

I thank my family members to motivate me and for their constant well-wishes.

I am grateful to some Japanese friends who supported me a lot during the time especially when I have been alone with my daughter in Japan, especially Mr. Kenji Kurokawa and Mrs. Keiko Matsuoka.

# List of Publications

---

## Journal:

1. Shahera Hossain and Seiichi Serikawa, "An In-depth Analysis of Co-occurrence Matrix-based Features", accepted in *Journal of the Institute of Industrial Applications Engineers*, IIAE, 2013.
2. Shahera Hossain and Seiichi Serikawa, "Texture Databases – A Comprehensive Survey", *Pattern Recognition Letters*, Elsevier, Available online, February, 2013.
3. Shahera Hossain and Seiichi Serikawa, "Co-Occurrence Matrix-based Statistical Model for Texture Analysis from Images", *Applied Mechanics and Materials*, Vol. 103, pp. 717-724, 2012.
4. Shahera Hossain and Seiichi Serikawa, "Statistical Analysis and Psychological Evaluation of Surfaces under Various Illuminations", *Applied Mechanics and Materials*, Vol. 36, pp. 422-429, 2010.

## Conference/Workshop Proceedings

1. Shahera Hossain and Seiichi Serikawa, "Features of Texture Analysis", *Proc. of SICE Annual Conference*, Japan, pp. 1739-1744, August, 2012 (available in IEEE xPlore Digital Library).
2. Shahera Hossain and Seiichi Serikawa, "Analysis of Texture Database", *Proc. of SICE Annual Conference*, Japan, pp. 1729-1733, August, 2012 (available in IEEE xPlore Digital Library).
3. Shahera Hossain and Seiichi Serikawa, "Image Texture Analysis using Second Order Statistical Model", *Proc. of International Symposium on Artificial Life and Robotics (AROB)*, Japan, January, 2012.
4. Shahera Hossain and Seiichi Serikawa, "Image Features Extraction Using Statistical Methods", *Proc. of International Conference on Precision Instrumentation and Measurement (CPIM)*, China, July, 2011.
5. Shahera Hossain and Seiichi Serikawa, "Study on Object Recognition on Vision System", *Proc. of SICEQ 2010 Conference*, Japan, Dec., 2010.
6. Yuhki Kitazono, Shahera Hossain, Lifeng Zhang, and Seiichi Serikawa, "An Improved Large Area-type Optical Sensor Using Frosted Glass", *IEEE Sensors*, USA, Nov., 2010 (available in IEEE xPlore Digital Library).
7. Shahera Hossain and Seiichi Serikawa, "Analysis of Statistical Characteristics of Image Surface", *Proc. of 9<sup>th</sup> World Congress on Computational Mechanics and 4<sup>th</sup> Asian Pacific Congress on Computational Mechanics (WCCM / APCOM 2010)*, Australia, July, 2010.

# Introduction

## Chapter 1

---

### 1.1 Introduction

A texture is one of the significant characteristics used in identifying objects of interest or regions in an image [1]. It is an important characteristic of various image surface properties [2]. A texture is also a crucial indicator in computer vision and image analysis [2]. It can be considered as a collection of images of similar texture appearances.

Textures are seen everywhere in the nature. We can also produce artificial textures. Various textures are seen in various materials, for example, river, wood, forest, leaves, stones, etc. (Fig. 1.1). Textures can be soft vs. hard, coarse vs. even, smooth vs. irregular-shape, not-glossy vs. glossy. Following Fig. 1.1 and Fig. 1.2 illustrate some natural textures and some man-made textures (also can be called as artificial textures).

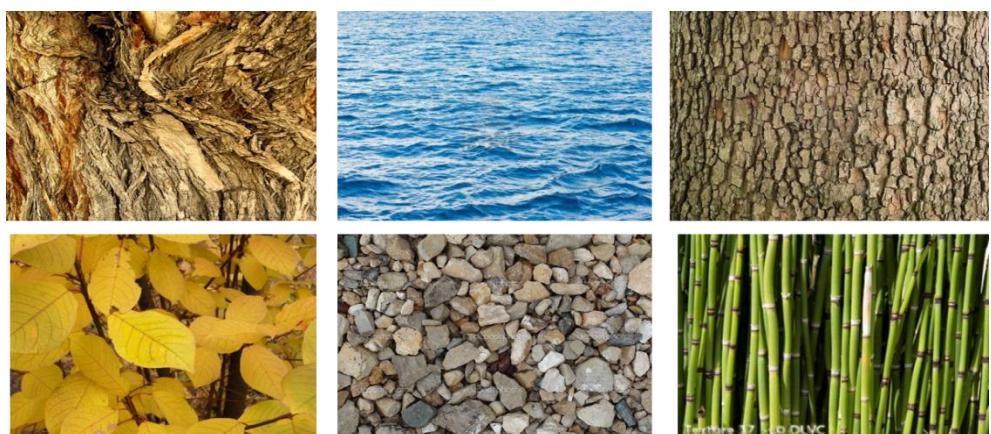


Figure 1.1: Example of natural textures.



Figure 1.2: Example of man-made textures.

In image processing aspect, an image texture is considered as an intense source of visual information. It can demonstrate about the nature and shape of any physical objects [3]. In another definition by [4], ‘Textures are complex visual patterns composed of entities, or sub-patterns that have characteristic brightness, color, slope, size, etc.’ Therefore a texture can be considered as a similarity grouping in an image’ [4]. A texture can be local sub-patterns which repeat themselves within the image. These properties can demonstrate various perceptions on visual information of an image. Some of these are: illumination, evenness, roughness, homogeneity, granularity, frequency, phase, arbitrariness, fineness, directionality, smoothness, etc. [3]. Sometimes, in an image, there may have several patterns or textures of different kinds. Hence, these different textures can represent partial information of the appearance, coarseness, lightness, structure and arrangement of that image [5]. Smarter extraction of features from image textures produce better cues for image analysis, which are pivotal for object recognition, surface analysis, action recognition, disease diagnosis, etc. There are various approaches for texture analysis, e.g., structural, model-based, statistical and transform methods [5].

## 1.2 Motivation

Texture analysis is an important research area in computer vision. Texture analyses have numerous applications in various areas on image processing, medical image processing, computer vision and related fields. In order to describe a texture properly, various efforts have been made by researchers. But, there is no easy way to describe an image texture. It is a challenging area of research, because both natural and artificial textures have many diversity as well as complexity. Since the last few decades, texture analysis has been a region of intense research. However, development on this area is not progressing rapidly. Most of the developed-approaches have not smartly considered real-life challenges and natural textures. It is known that real-world textures are very difficult to analyze. Therefore, in this thesis, we concentrate our work on three areas on texture analysis. One of them is various features analysis for different types of textural images. These features eleven features are widely used for various applications for the last forty years. But these are not well-explained by researchers why they are using this bunches of texture features computed from co-occurrence matrix. We explain and evaluate these features for the first time in details.

When we start texture analysis research, we noticed that there has been no overall study done, based on texture databases. Researchers use many texture databases of their own or they use benchmark datasets, which are available online. Our idea is to combine all texture databases based on their applications and properties to help future researchers to easily get idea from which database is suitable for their work. This is a state of the art in texture analysis research. We extend the local binary pattern (LBP) for texture recognition. LBP is a very simple and efficient for texture analysis. Based on LBP there are a lot of works, however,

mainly on face analysis. Also for rotational images, LBP is not very effective. So we focus on LBP for texture analysis and rotational images.

### 1.3 Thesis Outline

In the next chapter, we overview important features for texture analysis.

In chapter 3, we comprehensively review texture image database.

In chapter 4, we analyze co-occurrence-based features for standard database.

In chapter 5, we deeply analyze co-occurrence matrix-based features by using our developed real-world database that is very complex. We propose two different categories of these features based on directions and characteristics.

In chapter 6, we propose a method called *diagonal-crisscross local binary pattern* (DCLBP) for texture analysis. We also model two other operators of similar nature for texture analysis based on neighborhood pixel values.

Finally, in chapter 7, we conclude our thesis with our findings and future directions.

# Features for Texture Analysis

## Chapter 2

---

### 2.1 Introduction

In image processing and computer vision, *texture* is an important cue for analysis. As a human perception the word *texture* means what things are made of and how they feel or sense about the things. Human can easily identify textures property by their visual system or touch. Texture has its own property to understand. However, in computer vision and image processing, texture means uniformity of intensities of the image areas. Though sometimes, image property does not illustrate regions of uniform intensities. Usually, a typical wooden surface image is non-uniform because, the wooden surface repeated patterns with varied intensities. Even though a human can recognize texture when he looks into the surface of the object, it is not easy to define that texture.



(a) Brown bread

(b) Cotton

(c) Cordury

Figure 2.1: Three different texture images.

Figure 2.1 shows three image textures which are easy to understand for a person but not easy to define these textures. However, researchers define textures in various manners. For a human visual system, they key features are defined based on the following attributes: similarity, regularity and coarseness.

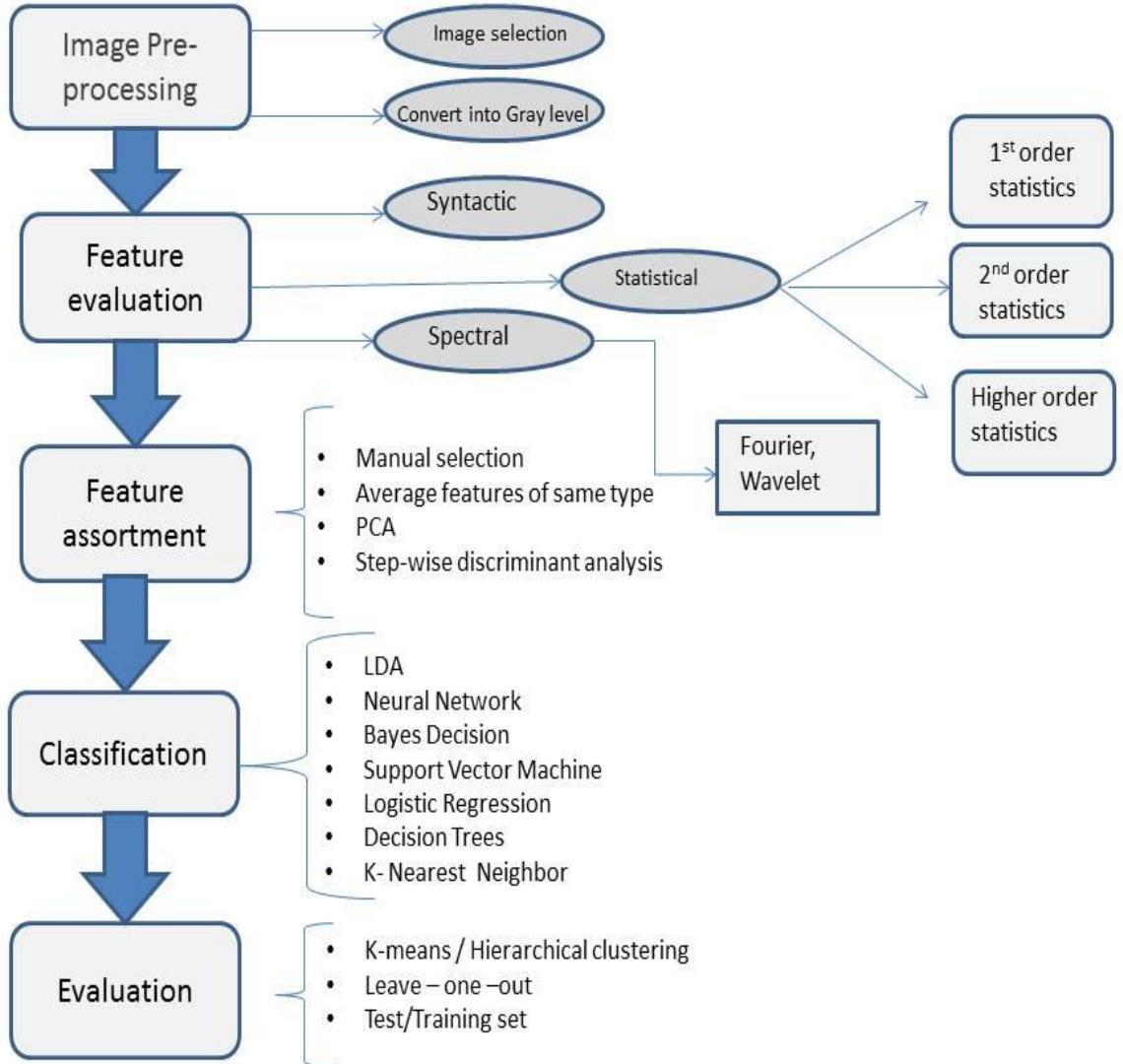


Figure 2.2: Different stages for texture analysis.

Now, we will explain the key aspects of texture analysis for non-visual analysis. Figure 2.2 shows different stages for texture analysis. After image acquisition, an image is preprocessed prior to feature evaluation. Converting a color image into a gray-level image can be considered an example of preprocessing. Image features are extracted through various ways.

Feature evaluation can be done through syntactic features, statistical features (e.g., 1<sup>st</sup> order statistics, 2<sup>nd</sup> order statistics, higher order statistics), or spectral features (e.g., Fourier, wavelet). It is necessary to point here that various ways of feature extraction is a foremost stage of texture analysis.

For the texture features, we can do texture classification using various classifiers. Some of the well-known classifiers are neural network, Bayes decision, support vector machine, K-nearest neighbor, decision trees, etc.

The review of this chapter is confined mainly to feature extraction and texture discrimination techniques. We will discuss some important feature models.

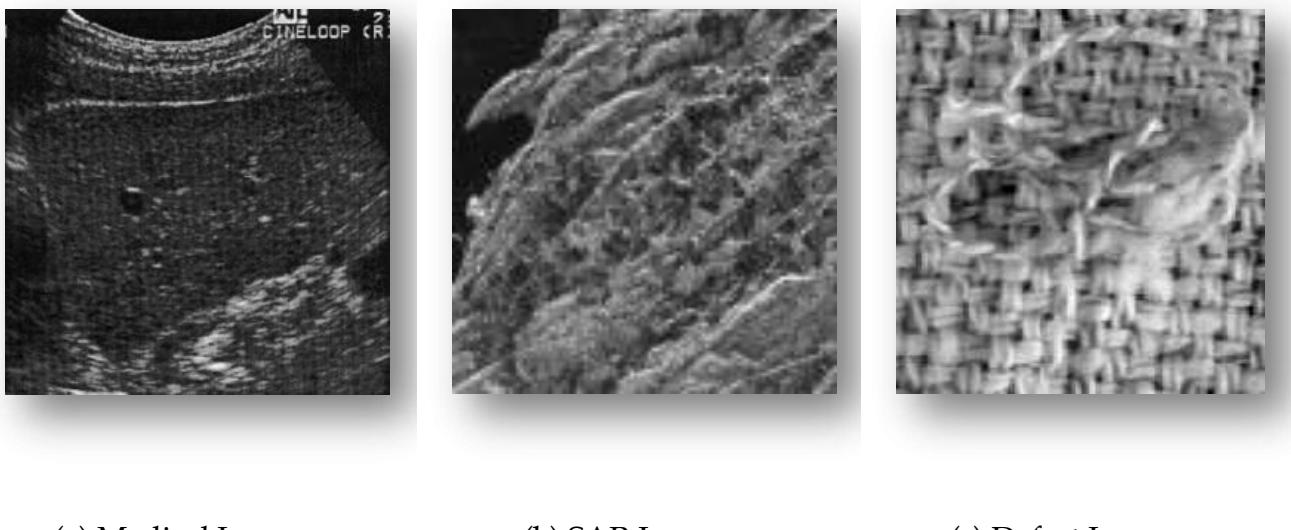


Figure 2.3: Three images demonstrate three different applications: (a) Medical image analysis; (b) SAR image; and (c) Defect image of a product. For these three applications, texture analyses are very important.

## **2.2 Various Applications**

Texture analysis is a part of computer vision and image processing. Analyses of textures are useful for different applications. Some of the major application areas are: image analysis, medical image processing, remote sensing, surface defect detection, and so on [7-12]. In this section, we briefly evaluate the application of texture in the area of defect detection for industrial inspection, medical or biomedical image analysis, and remote sensing, etc. In Fig. 2.3, we show three images based on the application.

### **2.2.1 Medical imaging**

Medical image processing is a very important area in the research community due to the rapid growth of computing power as well as the development of the medical equipments related to image processing (e.g., CT scan, MRI, PET, X-ray images, etc.) [7-9]. It is very much helpful for a physician to know the abnormal tissue, amount of abnormality in tissues, locations of abnormal tissues, etc. for a cancer patient or someone having tumor. Manual evaluations for a physician are difficult to properly identifying these information. Hence, automated feature analysis of various medical images for a problem becomes very crucial day by day. Automatic extraction of features can help for classification, e.g., differentiating normal tissue from abnormal tissue, finding locations or region of interest (ROI). Texture analysis is becoming a major part for medical analysis day by day.

### **2.2.2 Remote sensing**

Remote sensing is another important area where texture analysis is required. Remote sensing application takes satellite-based images of land, rivers, corp. forest, woodland, urban areas, rural areas, etc. And through various features and their analysis, we classify various geographical parameters, e.g., normal land vs. forest land, land vs. water-body, urban area vs. rural area.

### **2.2.3 Defect detection**

Industrial production requires identifying product defects, breaks, gaps, broken items, de-shaped or deformed products, etc. These are detected by using image processing through texture feature analysis. Industries related to textile,

automobile, food processing sectors, metal, tiles, etc. have enormous usage of defect detection [8,13-16].

#### **2.2.4 Others**

Apart from the above-mentioned applications, there are various other applications too. Some of these are: material recognition, content-based image retrieval, natural image analysis, text/document processing, GIS or environment modeling, texture synthesis, biometric identification, food-processing industry, etc. [17]. In computer vision, texture analysis is also an important part as a low-/mid-level image processing.

### **2.3 Different Approaches for Texture Analysis**

There are different approaches for texture analysis [18-39]. We categorize them as follows,

- Statistical approach-based texture analysis,
- Structural approach-based texture analysis,
- Model-based texture analysis, and
- Filter-based texture analysis.

#### **2.3.1 Statistical approaches**

Statistical approach-based texture analysis methods are one of the easiest approaches to extract image or texture features. These approaches can evaluate spatial distribution of gray values. It computes local features at different points in an image, and obtains a set of statistics from the distributions of the local features [6]. It is known that one of the key features of texture is the spatial distribution of pixel values within that image. These statistical approaches can be sub-divided into the following three categories [6] (as shown in Fig. 2.4):

- First-order statistical-based features: It deals each pixel. First-order statistics assess properties (e.g., average and variance) of distinct pixel values. It does not consider any spatial relations between image pixels.
- Second-order statistical-based features: deals two pixel values at specific locations next to each other.

- Higher-order statistical-based features: deals three or more pixel values at specific locations next to each other.

It is found that various methods employ statistical approaches for texture characterizations and recognition [11,24,26,29-34,37].

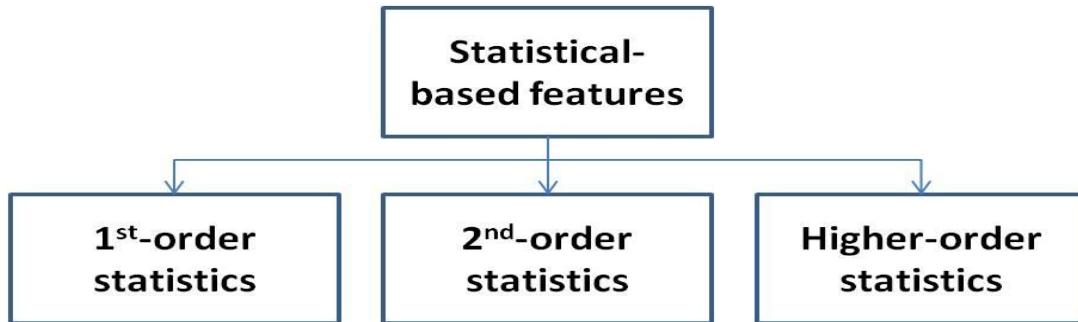


Figure 2.4: Statistical based feature extraction level.

#### *First-order histogram-based features*

In this sub-section, we present first-order histogram-based features that are elementary features for a texture. Let's consider an image  $I(x,y)$ , where  $x=0,1,\dots,N-1$  and  $y=0,1,\dots,M-1$ . We consider that the image  $I(x,y)$  has  $G$  different intensity levels, where  $i = 0 \sim G-1$ . For example, for a gray-scale image,  $i = 0 \sim 255$ , where  $G = 256$ . Then we can define the image *histogram* based on  $G$  different intensity levels according to the following equation:

$$h(i) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \delta(I(x, y), i),$$

where  $\delta(i,j)$  is called the Kronecker delta function. It can be defined as,

$$\delta(i, j) = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}$$

As the histogram considers each pixel separately and computes the frequency of intensity level, it is a first-order statistical feature. From a histogram, we can get an idea on the pixel distribution of an image. For a low-contrast image, the histogram distribution will be relatively smoother and not much varied. On the other hand, if an image has two different patches of pixel groups, then we may get a

bimodal histogram. However, histogram cannot maintain intra-pixel relationships based on neighborhood.

Another feature can be computed from the histogram, called approximate probability density of occurrence of the intensity levels. We can have this feature by dividing  $h(i)$  by the total number of pixels in that image. The following equation demonstrates this computation:

$$p(i) = \frac{h(i)}{NM}$$

Apart from the probability density of occurrence of pixel intensity values, we can also compute mean, variance, skewness and kurtosis. These are first-order statistical features. Mean or average can be computed according to the following equation:

$$\mu = \sum_{i=0}^{G-1} ip(i)$$

We get variance from mean value, which exhibits variation of intensity around the mean level of intensity of an image. It is:

$$\sigma^2 = \sum_{i=0}^{G-1} (i - \mu)^2 p(i)$$

Note that the mean and variance are inappropriate cues to provide adequate information about a texture. However, by exploiting images that are normalized against both the mean and variance, we can have improved texture distinction properties.

Skewness is another important first-order statistical feature. It can be computed as:

$$\mu_3 = \sigma^{-3} \sum_{i=0}^{G-1} (i - \mu)^3 p(i)$$

The value of skewness can be zero, or positive or negative value. If skewness is zero, then the histogram is symmetrical about its mean value. However, if the value is non-zero, then the polarity of the value defines whether it is skewed above or below the mean value. In fact, skewness indicates symmetrical distribution.

We also can compute kurtosis, which is a measure of flatness of the histogram. It is computed as:

$$\mu_4 = \sigma^{-4} \sum_{i=0}^{G-1} (i - \mu)^4 p(i) - 3$$

Though the above features cannot completely characterize texture, the key advantage of using the texture parameters by first-order statistics is mainly their simplicity.

### ***Co-occurrence and gray-level differences-based features***

The gray-level co-occurrence (GLC)-based statistics are described as a second-order statistical based feature extraction [41]. We can calculate gray-level co-occurrence matrices, in short called as GLCM [25,34-35]. It is a very well-known terminology and widely used for texture analysis and other purposes. It is a 2-dimensional co-occurrence matrix. In GLCM, the number of rows and columns are equal to the number of gray levels in the image. Note that in the gray-level co-occurrence matrix, computations are done on pixel pairs, not from individual pixel values [17].

### ***Local Binary Pattern-based features***

Local binary pattern (LBP) is a new operator that becomes very popular in computer vision and image processing researches. It characterizes the concept of local image contrast [17,42-45]. The local binary pattern operator is related to the statistical as well as structural texture analysis of an image. The LBP describes a texture by the smallest element called *texton*, which is a histogram of texture elements. For each pixel in an image, a binary code is produced by thresholding its neighborhood pixels comparing with the center pixel. Neighborhood can be the closest 8 pixels or more based on the variations of the LBP operations. In LBP, various binary codes illustrate

different kinds of edges, spots, flat areas, etc. The LBP becomes one of the most widely-used methods in many applications. The reason is: the computational cost of the LBP is very small and it is very easy to implement and fast [46]. There are a great number of different variants of LBP operators in the literature [46].

Another very well-known feature is the Scale-Invariant Feature Transform (SIFT) descriptor. The SIFT is a distinctive invariant feature set and is suitable for describing local textures [42]. It is rotation-invariant and very widely-used for object recognition and tracking purposes. Few LBP-related features or operators are: LTP (Local Ternary Pattern), LPQ (Local Phase Quantization) operator, Number-LBP (NLBP), Median-LBP, and Volume-LBP [44-45,47-48,17].

### **Autocorrelation**

The autocorrelation feature is another crucial statistical feature. An important characteristic of texture is the repetitive characteristic of the position of texture elements in that image. Based on observation, for textures with natural repetition, the autocorrelation feature becomes more useful [49-50]. The autocorrelation feature of an image is used to evaluate the fineness or roughness, smoothness or coarseness of the texture in the image [51]. It is related to the size of the texture primitive. For normal textures, the autocorrelation function shows peaks and valleys. However, for a rough or unsmooth texture, the autocorrelation function goes down slowly. On the other hand, for a smooth texture, the autocorrelation function goes down very quickly [51]. It has relationship with power spectrum of the Fourier transform. It is also responsive to noise interference. The autocorrelation function of an image  $I(x, y)$  is defined as,

$$P(x, y) = \frac{\sum_{u=0}^N \sum_{v=0}^M I(u, v)I(u+x, v+y)}{\sum_{u=0}^N \sum_{v=0}^M I^2(u, v)}$$

### **Laws texture energy features**

A novel texture energy feature is introduced by Laws, which are called Laws texture energy features [51-53]. Some operators (e.g., Laplacian operator, Sobel operator) can highlight the underlying microstructure of texture within an image. The

concept can be echoed with the concept of high-pass filter. The Laws filters or masks can be produced by convolving three basic  $1 \times 3$  masks, which are shown below:

$$L3 = [1 \quad 2 \quad 1]$$

$$E3 = [-1 \quad 2 \quad -1]$$

$$S3 = [-1 \quad 2 \quad -1]$$

Here, we can find three one-dimensional filters, which exhibit the following three concepts: Local average, Edge detection, and Spot detection.

We can also have  $1 \times 5$  masks. These masks can be attained by convolving pairs of the above three  $1 \times 3$  masks. The new masks are:

$$L5 = [1 \quad 4 \quad 6 \quad 4 \quad 1]$$

$$E5 = [-1 \quad -2 \quad 0 \quad 2 \quad 1]$$

$$S5 = [-1 \quad 0 \quad 2 \quad 0 \quad -1]$$

$$R5 = [1 \quad -4 \quad 6 \quad -4 \quad 1]$$

$$W5 = [-1 \quad 2 \quad 0 \quad -2 \quad 1]$$

These five masks represent the following concepts:

- L: Local average;
- E: Edge detection;
- S: Spot detection;
- R: Ripple detection; and
- W: Wave detection [52].

### 2.3.2 Structural approaches

A texture can have its primitives called *texton* or *microtexture*. In the structural approaches for feature analysis, we consider the microstructure and develop a hierarchy of spatial arrangements from these microstructures to produce *macrotexture* [41]. For this representation, apart from defining the microstructure, a placement rules should be introduced. The placement rules are defined based on microstructure or a set of primitive microstructures, and based on the probability of the placement of the

primitive. The structural approaches describe symbolic portrayal of an image. That is why, it is found to be more effective to synthesize image, rather than analyze it. However, it is not much effective and proven approach.

### 2.3.3 Model-based approaches

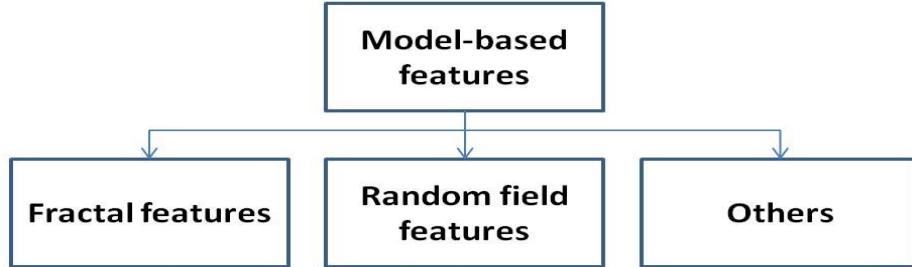


Figure 2.5: Model-based feature extraction.

In this sub-section, we present several model-based features for texture analysis. Model-based texture analysis makes an attempt to understand a texture employing one of the following two models:

- Generative image model (e.g., fractal features) and
- Stochastic image model (e.g., random field features).

#### ***Generative image model – Fractal feature***

Fractal features can be considered to develop discriminative and invariant features for texture classification, especially in cases where scale changes are prominent in textures [14-15]. Such variations can be handled by choosing local interest points and selecting their characteristic scales. But, apart from the problems related to have suitable local interest points, it is difficult to have an optimized characteristic scale that can be suitable for different textures of varied scales [57]. It is found that fractal features demonstrate promising results while handling scaling constraint. However, fractal-based classification approaches do not consider statistical characterizations of textures by using global features. Moreover, they cannot differentiate between key texture elements (e.g., edges, corners, etc.). These issues can be overcome by evaluating densely local fractal features. Fractal models can classify irregular textures too. A multi-fractal analysis is proposed for texture descriptor [58].

### ***Stochastic image model – Random field feature***

Image textures can be modeled as a Markov random field (MRF) of pixels gray-level. In this approach, relationships between the gray values of neighboring pixels are statistically defined [16,54-55]. For example, a Gauss-Markov random field-based probabilistic texture model is developed to characterize hyperspectral textures. Another similar model is proposed for texture classification and texture segmentation in Ref. [56]. There are few other approaches based on random fields [56]. However, usually it is noticed that feature-based approaches are less computationally-expensive, as well as more effective than MRF-based approaches.

### **2.3.4 Filter-based approach/transform**

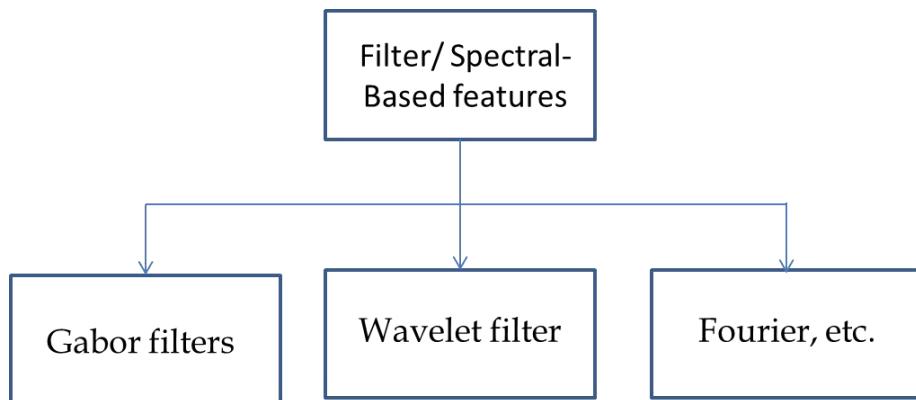


Figure 2.6: Filter-based feature extraction.

In this sub-section, we introduce filter- or, spectral-based features. These are mainly Gabor filter, Fourier filter and wavelet filter. Fourier transform-based features cannot function well, because it losses various spatial localization of an image. However, Gabor filters or wavelet filters are more widely considered for texture analysis.

#### ***Gabor filter-based texture features***

Gabor filter-based texture features are necessary features for texture analysis [59,42,60,56]. Gabor functions share lots of common features similar to our visual system [60]. Gabor functions consist of a sinusoidal plane wave of some frequency and orientation, which is modulated by Gaussian envelope. A Gabor filter is a band-pass filter which can be used to extract a specific band of frequency components from an

image [61]. Usually, it is a linear filter that is widely employed to detect edges in an image. However, they become very useful for texture representation as well as classification. For texture analysis, we extract a set of Gabor filters with different frequencies and orientations.

There are a number of Gabor features, which are used for feature analysis,. Some of these Gabor features are:

- Linear Gabor features,
- Thresholded Gabor features, and
- Gabor-energy features, etc.

### ***Wavelet-based feature***

Some transforms are developed to characterize localized frequencies specific to each pixel location based on various wavelet transform methods. Wavelet transform has multi-scale frequency contents (named as, wavelet coefficients), at each spatial location of an image [62]. It can decompose image texture scale, i.e., it can characterize textures at multiple scales [63]. The multi-resolution properties of the wavelet transforms are beneficial for distinguishing textures [63]. However, the wavelet transforms are usually computationally taxing [62]. A tree-structured wavelet transform is proposed for texture analysis and classification [53,64].

The Discrete Wavelet Transform (DWT) is the most widely used version of Wavelet transform. It encapsulates even tiny differences in rotation or scale. Some other representations on wavelet transform can be

- Discrete Wavelet Frame Transform (DWFT) [6],
- Discrete Wavelet Packet Transform (DWPT) [63],
- Haar wavelet [35],
- Modulus Maxima of a Continuous Wavelet Transform (MMWT) [58],
- Multi-Orientation Wavelet Pyramid [58],

- Dual-Tree Complex Wavelet Transform [65], or
- Scale-, rotation-, and translation-invariant wavelets [66-67].

In terms of spatial localization, Gabor filter performs better than its counterparts. Still, its usage is not profound. If we comparatively analyze the Gabor transform and the Wavelet transform, we notice that the later is better. The key reasons are:

- Wavelet transforms permit different spatial resolutions.
- One can decide to exploit a specific Wavelet for an application. So, Wavelet can offer flexibility to choose different functions for different applications.

## 2.4 Conclusions

Textures are all around us – in the nature, in man-made products, or through artificially-simulated texture scenes. Hence, texture analysis becomes a very important and popular research works for various applications. Application domains of texture analysis are broadly in medical image analysis, satellite image analysis, document extraction, fault analysis in products, computer vision, etc. Texture analysis is very important area for researchers especially in medical field. It is also going to play a pivotal role in future, because it has a variety of different application domains. However, the features and corresponding analysis are not yet finished in a reasonably applicable manner for different applications.

For texture analysis, at first we have to find the key features from image. There are lots of methods to extract features from an image, such as statistical method, structural method, model-based method and filter-based method. Among these four methods, statistical and filter-based methods are used more than other approaches. Statistical features are, therefore, one of the popular methods that are proposed in the computer vision literature. There are two broad approaches to study image texture: (i) By computing statistical properties from the extracted texture elements and by utilizing these properties as texture features; and (ii) By involving structural methods to analyze texture.

In this chapter, we explain how we can analysis texture from image and how many ways we can extract features. We survey various important features that are suitable for texture analysis. There are various approaches for different types of texture images. So far, there is no comprehensive work that deals with key features and experiment with different types of datasets. Apart from the issue of variety of features, different types of texture datasets are also available.

# In-depth Analysis of Texture Databases

## Chapter 3

---

### 3.1 Introduction

Though there are a number of texture datasets, till-to-date, there is no comprehensive works covering the important databases and analyzing these in various perspectives. In this chapter, various datasets on texture are categorized and critically surveyed. We categorize them in to the following four areas, namely texture databases in medical imaging; natural texture image database; texture of materials database' and dynamic texture database. We initially present the datasets related to medical imaging, then natural texture image datasets are covered. Datasets related to texture of various materials are discussed afterwards, which is the largest group of datasets from various groups. The final category is the dynamic texture dataset. An overall comparative discussion with Tables is presented at the end.

### 3.2 Texture Databases in Medical Image Processing

A significant amount of research has been done in the field of texture analysis in medical image segmentation [68-72]. The analysis of texture parameters is a useful way of increasing the information obtainable from medical images. It has various applications ranging from the segmentation of specific anatomical structures and the detection of lesions, to differentiation between pathological and healthy tissue in different organs, and texture analysis from radiological images [5]. Under this area, the important datasets are presented below.

#### 3.2.1 MRI brain database

Forty-three volumetric MRI-T1 brain dataset is developed from 28 patients and 15 controls. This dataset is obtained on a 1.5-T Siemens scanner (MPRAGE sequence, TR 11.4 ms, TE 4.4 ms, 128 slices, matrix 256x256, voxel size 0.9x0.9x1.5

mm). In this case, patients are diagnosed clinically as either suffering from WM encephalopathy and/or Alzheimer's disease, while controls are 15 healthy elderly individuals. Images are interpolated to an isotropic resolution of 1 mm and aligned with the stereo-tactical coordinate system using fourth-order b-spline interpolation tested additionally on the large sample of 210 MRI-T1 brain datasets of young healthy subjects including 103 male (group A) and 107 age-matched female (group B) with average age of 24.8 years and standard deviation of 3.9 years in the same way [72,74].

### 3.2.2 USF database

References [75-80] use the Digital Database for Screening Mammography (DDSM) or USF database by [81]. This database [81] contains approximately 2,620 cases. In each of the case, two images of each breast, associated patient information like age, stage of the tumor, subtlety rating for abnormalities, American College of Radiology (ACR) breast density rating are studied. The mammograms are digitized by various scanners depending on the source of the data. These images are available with the specification of  $3000 \times 4500$  pixels with 16-bit pixel depth. One hundred mammographic images from the Digital Database for Screening Mammography (DDSM) [81] are exploited by [82]. These are digitized with a LUMISYS 300 scanner at 12-bit pixel depth and spatial resolution  $50 \mu\text{m}$ .

All mammograms selected correspond to heterogeneously and extremely dense breast parenchyma (density 3 and 4 according to the ACR BIRADS<sup>TM</sup> lexicon) and contain subtle microcalcification (MC) clusters. There are 46 benign and 54 malignant clusters according to the database ground truth tables. Reference [82] considers regions of interest of  $128 \times 128$  pixels, containing the MCs, and these are employed for texture analysis.

### 3.2.3 Others in medical analysis

A significant amount of research has been done in the field of texture analysis in medical image segmentation, e.g., in [68-72,76]. Kovalev [72] demonstrates the use of multi-sort-co-occurrence matrices on MRI brain datasets. Karkanis *et al.* [7] classify regions containing cancers in colonoscopy images. They use a multi-layer feed-

forward neural network, which is based on second order gray-level statistics. Two different types of colonoscopic images are taken from two different colons in this experiment (i.e., macroscopically a Type III lesion, which is histologically a *low grade cancer*; and macroscopically a Type V lesion that is Histologically a *moderately differentiated carcinoma*). The images are digitized to a size of 2000x2000 pixels with 64 gray levels depth per pixel. Textures from 10 normal and 10 abnormal tissue samples, corresponding to a 256x256 image window, have been randomly chosen from each image [7].

### 3.3 Natural Texture Image Database

This is the second category of texture image database where natural images are dominant in terms of image classes in these datasets. In Section 4, we present texture datasets of materials and we will find that there are some overlapping in both types. However, the Brodatz dataset is purely based on natural texture images. We find that some datasets are created at the top of another dataset. For example, the USC-SIPI (University of Southern California – Signal and Image Processing Institute) Texture Mosaics dataset is based on the Brodatz dataset.

#### 3.3.1 Brodatz texture database

The Brodatz Texture Database or album is the most famous the most widely used dataset in the texture analysis [69,71,80,83-97]. The Brodatz texture database is derived from the Brodatz album. The Brodatz textures are the most commonly used texture data set, especially in the computer vision and signal processing community. Because they are so commonly used by previous texture analysis/synthesis works, it is almost inevitable to include at least some of them in a texture synthesis paper [98]. It has 112 classes, and a small number of examples for each class. The Brodatz album contains 112 images with size 512x512 and 256 gray values after digitizing, showing a variety of textures, both small and large grained, collected for artistic purposes [99]. These digital images are not scans of the pages from the texture book. The images are scans of a set of glossy black and white prints that were purchased from the author. While these prints are pictures of the same textures as in the book, in most cases they are not the same image as the one in the book [83]. Despite the

popularity, however, Brodatz textures are actually copyrighted [99], even though many researchers are using these images.

The *Brodatz* texture database is based on image rotated textures. Various algorithms exploit the *Brodatz* texture database for evaluations, though in most of the cases, the entire database is not employed. Different papers use various number of image sets from this dataset. For example, twenty-three distinct natural textures are selected from the Brodatz album in [63]. Some 209 natural texture images are used for training and 180 different texture images are used for testing by [100]. They also use 48 synthetic texture images for classification [100], along with all 112 Brodatz textures included in Brodatz [83]. References [56,101-102,94,67] use Brodatz database [83] too. Reference [56] exploit only 25 Brodatz textures from the Brodatz dataset. Reference [103] use 15 textures only. Only a few publications actually report results on the entire database [85,96,104-106] and most others use a partial set of textures. Figure 3.1(a) shows some sample frames from this database.

The Brodatz texture set is indispensable, even with the existence of VisTex [73,107]. The reason is that most of the Brodatz textures are photographed under controlled lighting conditions, so the images are of very high quality. In addition, they expose to the most amount of textures so that irrelevant information such as noise and ‘non-textural’ stuff are not there [98]. One key concern of this texture database is that it cannot provide photometric stereo image sets for each texture class. This dataset also cannot provide true surface rotation [94]. Moreover, in recent years, it has been criticized because of the lack of intra-class variation that it exhibits [85]. This dataset is diversity of textures. Some of these textures are almost similar, and some others are very inconsistent, inhomogeneous or non-identical. Therefore, a human may even fail to correctly classify these in groups [85]. Overall, it is a very difficult dataset to analyze [85]. Hence, it becomes a benchmark dataset for analyzing any new approach or model for texture analysis.

### 3.3.2 Vision texture database (VisTex)

The Vision Texture Database (VisTex) is another well-known database [73,107-110], formed by the Vision and Modeling group at the MIT Media Lab. It contains 167 colored reference textured images with size either a 786x512 or 512x786 image depending on the orientation of the scene (also have other sizes). Images are grouped according to their contents. Unlike other texture collections, the images in VisTex do not conform to rigid frontal plane perspectives and studio lighting conditions.

The goal of VisTex is to provide texture images that are representative of real world conditions. VisTex database can serve as a replacement for traditional texture collections [108]. It also provides some examples of many non-traditional textures (such as texture scenes and sequences of temporal textures). The experimental setup of [67] consists of images from the Brodazt and VisTeX [111] texture databases. They also consider a combination of some images. There are two image sets, namely:

- Homogeneous textures (called Reference Textures) (available in 128x128 pixel and 512x512 pixel images) and
- Multi-texture scenes (available in 192x128 pixel and 786x512 pixel images).

The images are available in raw files along with the information of the images, illuminations status and perspective. Figure 3.1(b) depicts few images for this dataset.

The most important difference between VisTex database and other texture databases is that it does not conform to *rigid frontal plane perspectives* and *studio lighting conditions*. For this dataset, the lighting conditions are daylight, artificial-florescent and artificial-incandescent, and some of the lighting conditions are imprecise. For example, descriptions are given as '*daylight, direct and from right*'. VisTex textures are photographed under natural lighting conditions, so they are tougher; i.e., the images contain more noise, and extra visual cues such as shadows, lighting, depth, perspective. [98]. With regard to perspective, the angle between films an object plane, there are two settings: frontal-plain and oblique. Considering

the limitations of VisTex database with unknown illumination directions, one may not use them with texture classification scheme, whereas, one has to recover the surface properties from several images with the known and controlled light conditions using photometric stereo. In future, they want to include video textures and video orbits [108].

### 3.3.3 USC-SIPI texture mosaic dataset

The USC-SIPI (University of Southern California – Signal and Image Processing Institute) Texture Mosaics dataset has three texture mosaic images [112] that are available on the USC-SIPI website [113]. Texture mosaic#1 [114] has image of 512×512 pixel, containing eight textures in square regions of 128x128, 32x32, and 16x16 pixels. Mosaic#2 is created to provide a test image similar to mosaic#1 but with information about the contents of the image. It is composed of eight different texture samples (i.e., grass, water, sand, wool, pigskin, leather, raffia, and wood) taken from the Brodatz texture book [83]. All eight textures are present in the image in squares of size 128x128, 64x64, 32x32, and 16x16. And the mosaic#3 is developed to test the effect of non-horizontal and non-vertical texture boundaries. It also contains the same eight Brodatz textures that are present in the first mosaic.

The textures are present in the image in approximately equal proportions. The mosaic consists of three basic regions [112]: the upper-left portion contains three textures in an arrangement where two textures are converging along curved paths; the upper-right portion of the new mosaic contains regions with non-vertical and non-horizontal boundaries, both straight and slightly curved. The bottom half of the image is made up of the eight textures in irregularly shaped regions of approximately equal size [112].

There is another dataset under the same group, called the USC-SIPI (University of Southern California – Signal and Image Processing Institute) ‘Rotated Textures’, which is part of the textures volume consist of thirteen of the Brodatz texture images each digitized at seven different rotation angles: 0, 30, 60, 90, 120, 150, and 200 degree (total of 91 images) [113]. The images are all 512x512 pixels with 8 bits/pixel. These rotated texture images are scanned using a 512x512 pixel video

digitizing camera. Hence, the qualities of the scanned images are probably not as good as those in the main part images.

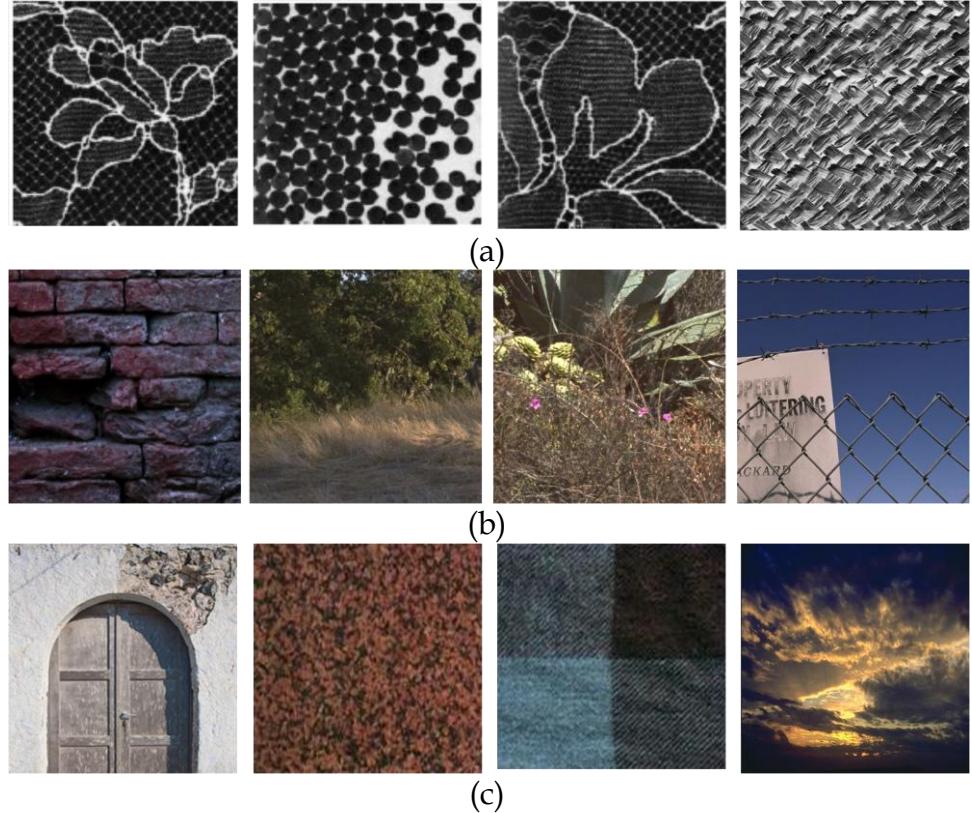


Figure 3.1: Natural Texture Image Dataset: Four sample images of (a) Brodatz texture set [83]; (b) VisTex dataset [73,107]; (c) Texture library dataset [116].

### 3.3.4 Texture library

This is a library of various textures from different types of images [116]. It has 17 albums of various high resolution images on doors, forest, bump, cloud, fabric, maps, nature, etc. Though it has a good variability in terms of different classes as well as resolutions, no published report has been found on its usage. Also, TILDA (Textile Texture Database) and Forrest Texture Library have various images from forests, e.g., trees, bark, moss, rock, etc. in high resolution. Figure 3.1(c) shows some image frames.

### 3.4 Texture of Materials

This section presents most of the texture datasets available in the literature. Most of the datasets are based on various real-life materials, though partially these cover natural textures and scenes as well. MeasTex (Measurement of Texture) database has both natural as well as artificial texture images. We split these datasets under this group considering the types of images and/or applications. These texture datasets are suitable to analyze materials, surfaces, object recognition, pattern analysis, etc. A good number of these are photometric texture databases. We will notice more distinct features in Discussions and Tables presented below.

#### 3.4.1 Meastex database

The MEASurement of TEXture (Meastex) sets contain examples of artificial and natural (real world) texture images [51,110-111]. The real world image has a size of 512x512 pixels (in raw PGM format) [111]. Each image is split into 16 sub-images to increase the number of samples available for each class. The textures are of four classes: asphalt (64 images), concrete (192 images), grass (288 images) and rock (400 images), which provide total 944 images. The artificial textures are generated by the makeBrick program that creates brick-like textures [111]. Brick textures may then be created by either overlaying the stone texture with a mortar pattern or transferring randomly cut bricks from the stone texture and placing in a new image, separating the bricks with mortar.

There are three texture sets are available here, namely - bomb (normal background texture), lattice (overlaying normal background texture with mortar), and mortar (separating randomly cut bricks with mortar). On the other hand, some natural textures are compiled (and continued process) to create a database of natural textures. The images have been obtained from 6"x4" color photographs taken with a 35 mm camera [111]. Each photograph is scanned at 256 dpi and stored in PPM format. The distributed images are 512x512 pixels areas, which have been cropped from these full-size images and converted to PGM format. In this dataset, similar to the VisTex annotations, annotation header is used to label each image. For example, a comprehensive database of grasses (most labeled with species) makes up a large

proportion of the textures in this set. Other significant texture sets are based on materials and surface textures. Reference [103] propose a multi-channel approach to texture description where they compare the Brodatz textures (15 textures only) and MeasTex datasets.

Although a number of texture sets in MeasTex have been compiled by other texture databases such as the Brodatz texture database and the VisTex database, most of their natural textures are 2D texture rather than 3D texture. The MeasTex provides a database of homogeneous texture images, several test suites of texture classification problems, implementations of major texture classification paradigms, and a framework for the quantitative measurement of texture classification algorithms on the texture problem test suites [115]. See Fig. 3.2(a) for few images for MeasTex dataset.

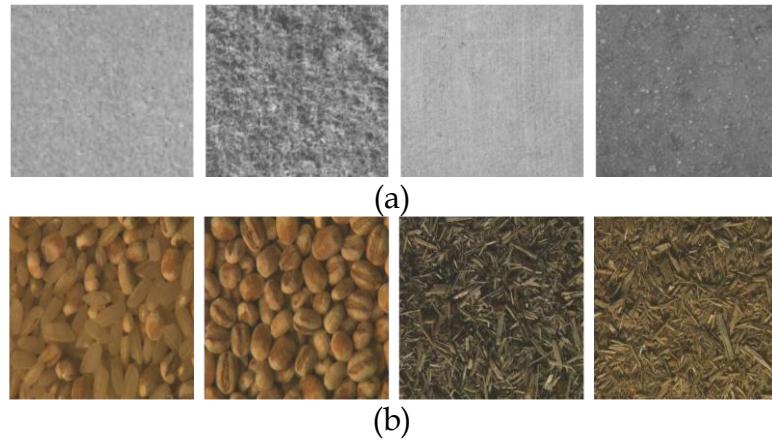


Figure 3.2: Four bodies of (a) Meastex dataset [51]; (b) OUTex dataset [117].

### 3.4.2 PhoTex database

Heriot-Watt University at Edinburgh creates Photometric Texture database (PhoTex) [118-119,90,99] that consists of images of surface textures that are observed from a constant viewpoint for 40 different illumination directions. It is a texture database of rough surfaces, and a few smooth surfaces. The main variables in the database are azimuth and zenith of the illumination. In few cases, the surface sample is also rotated. This database therefore mainly focuses on the changes of illumination condition rather than the surface rotation. This database is intended to provide physics-based, photometric data for texture research. It is called this database as

physics-based (which means it is applied to algorithms that model, or account for, the physical process of image formation and capture) because image surfaces are taken under controlled conditions, and calibration images are taken that enable the imaging process to be modeled [118]. Images that allow the user to calibrate the image transfer function, and measure the noise in the process are also held in the database.

The images are grayscale with resolution of 1280x1024 pixels. These are unprocessed TIFF images. Images are captured by using a Vosskuhler CCD 1300LN digital camera, with a Matrox PC-SIG framestore. The measured intensity in the images of this dataset is proportional to irradiance. This database seems important for texture researchers who are working on illuminant direction invariance, surface rotation invariance and photometric shape estimation. Another potential point is that researchers interested in extracting and processing information from rough surfaces can use this dataset.

### **3.4.3 PhoTex 3D database**

The photometric texture (PhoTex) project has a recent target to develop a new methodology for texture classification based on the direct use of surface gradient (relief) and surface reflectance (albedo) information [120]. This dataset provides real surface rotation rather than image rotation, where most of currently existing texture databases only support image rotation; and registered photometric stereo data sets, where 3D texture surfaces are captured under the controlled illumination conditions. This is the first texture database that provides both full real surface rotations and registered photometric stereo data, which is different from other existing texture database [120]. It contains 30 real textures, and there are currently 1680 images in the database. This will enable the design of image texture analysis systems that exploit and take into account the illumination conditions and the three-dimensional nature of texture. The dataset can be available from [121].

### 3.4.4 ALOT database

Amsterdam Library of Textures (ALOT) database [122-125,90,119] consists of 250 rough texture classes (100 number of images per class), which systematically varied the viewing angle and illumination angle so that it can capture the sensory variation in texture recordings. It has eight different illuminations in three orientations, having four viewpoints. These are divided into tune, train, and test parts, each with 2400 samples. Figure 3.3 shows some textures from the ALOT dataset. The ALOT dataset includes some very easily recognizable materials as well as extremely difficult ones. Although the number of view-illumination directions per material is only half the BRDF bi-directional reflectance distribution function) resolution of CUReT (Columbia-Utrecht Reflectance and Texture) database, the ALOT extends the number of materials almost by a factor 5, and it improves upon image resolution and color quality. Furthermore, different light source colors have been added to test (texture) color constancy algorithms. The acquisition setup for the ALOT is very similar to the ALOI collection of objects [90].



Figure 3.3: Some sample images from the Amsterdam ALOT database (ALOT class numbers are shown in ( )); Row#1: tea-wafers (9), brown bread (26), cotton (43), terry cloth (48), punched plastic (56); Row#2: cork (57), cotton (60), ribbed cotton (64), sponge (176), chamois (196) [90,122].

Note that due to small misalignments between the two cameras, the viewpoints are not perfectly (pixel accuracy) identical [90]. For materials with not too much depth variation, the distortion can be well approximated by a planar

rotation and translation between the two views. Hence, image homography registration can be applied to align the images between the two cameras [90].

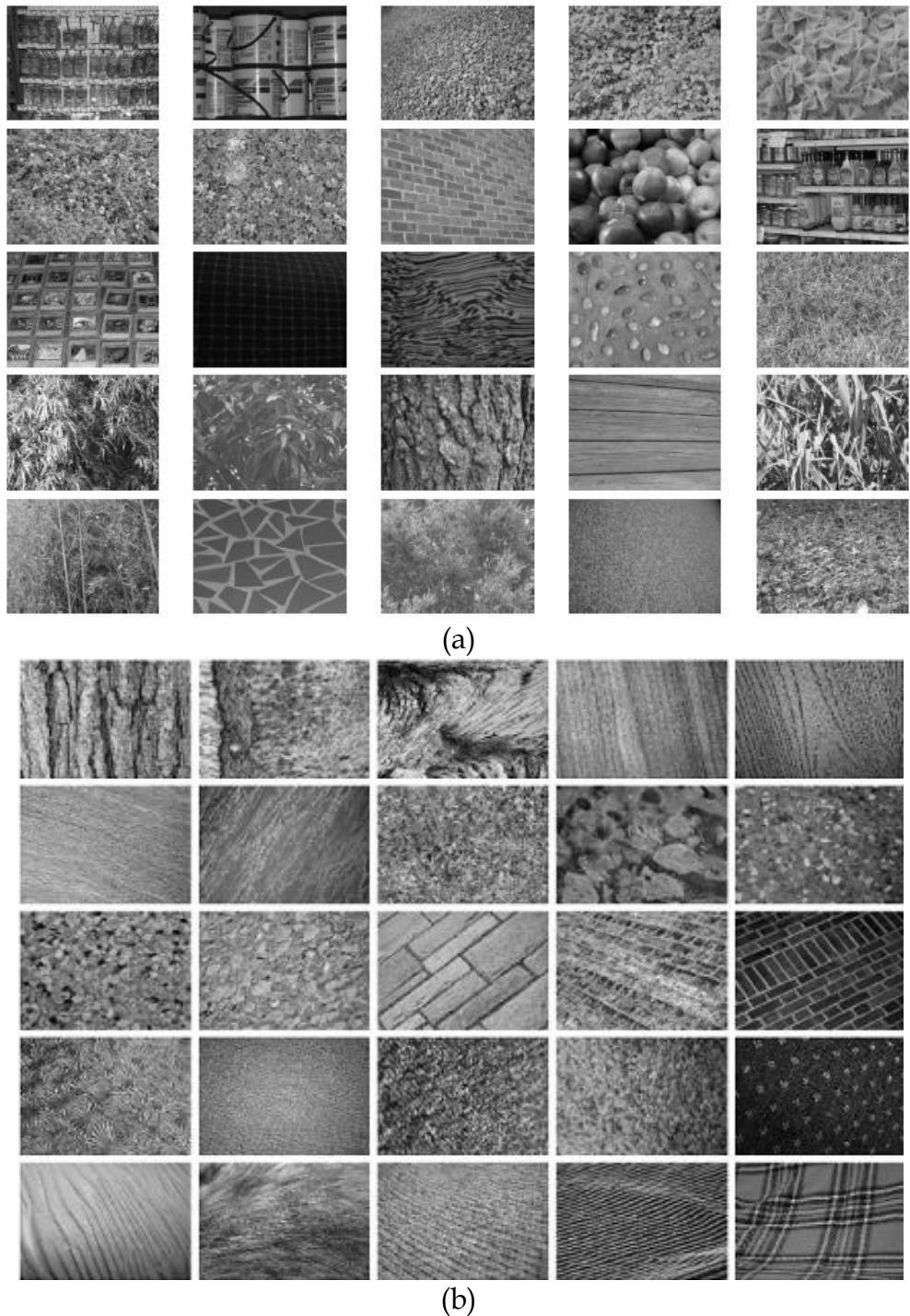


Figure 3.4: Example of 25 classes each from: (a) UMD dataset [126]; (b) UIUC database [127].

### **3.4.5 UMD dataset**

The UMD (University of Maryland, College Park) is a dataset of high-resolution texture [126,89,128-130,58] that consists of 1000 un-calibrated, unregistered images, taken from a family camera. It has 25 texture classes with 40 samples, having resolution of 1280x900 pixels. Similar to the UIUC database [131], within each class – the UMD texture dataset has significant viewpoint changes and scale differences. Moreover, the illumination conditions are uncontrolled for the UMD dataset. The textures of this dataset are non-traditional, including images of fruits, various plants, floor textures, shelves of bottles and buckets [126]. The dataset can be available upon request to the authors, as mentioned in Ref. [126]. Fig. 3.4(a) shows a sample texture image per class.

### **3.4.6 OUTex database**

The OUTex database [132-134,124,468] stands for University of Oulu Texture database. This image database contains a large collection of textures, both in the form of surface textures and natural scenes. The OUTex is a framework for empirical evaluation of texture classification and segmentation algorithms. The surface textures have wide variations in terms of illumination direction (three illumination sources), surface rotation, and spatial resolution. Note that the OUTex database cannot be used to provide registered photometric stereo data [135]. In this case, all three different illumination directions are lying on the same plane – either coplanar or collinear. And these illumination variations vary only in the change of slant angle, while the tilt angle remains constant. Hence, it is not possible to correctly resolve the surface partial derivatives from these images using photometric stereo, as the inverse of the lighting matrix in the photometric stereo solution does not exist when the three illumination vectors lie in the same plane [135]. Fig. 3.2(b) shows some images for OUTex dataset.

### **3.4.7 CUReT database**

The Columbia-Utrecht Reflectance and Texture (CUReT) database is another well-known and very challenging database [136-149,125,95,91,85,87]. The CUReT database is a considerable improvement over the Brodatz collection. The CUReT is developed at Columbia University and Utrecht University. This is a variety of tile

datasets with different types of defects including physical damage, pin holes, textural imperfections, pattern mis-registrations, etc. The test samples have resolution of  $512 \times 512$  pixels. It has 61 different materials. It has many real world textures, taken under varying image conditions, and the effects of specularities, shadowing and other surface normal variations are evident [139]. The set of images for each texture sample is obtained over a wide range of viewing and illumination directions. Figure 3.5 shows three images for the same materials in different illuminations and poses. Various material images from the Columbia-Utrecht database (CUReT) are exploited by [148].

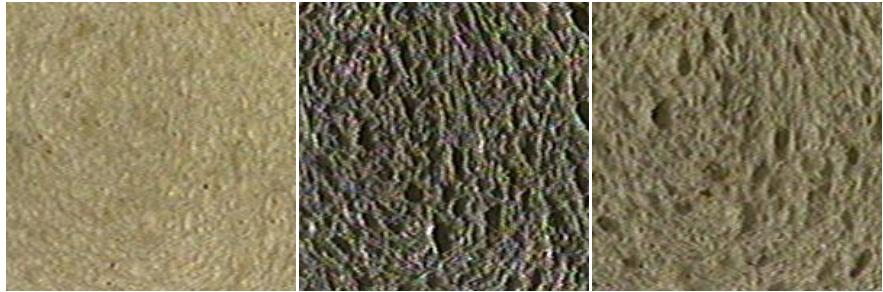


Figure 3.5: Example of CUReT database: ‘white-bread’ image in three different poses and illuminations [150].

Though the CUReT database is a good one, it has some constraints too. For example, there is no significant scale change for most of the textures and limited in-plane rotation. In this case, for all measurements of a selected texture sample, the light source remains fixed. It is noted that a camera is mounted on a tripod and its optical axis is parallel to the floor. Therefore, it can be positioned to any one of seven different locations in a plane during measurements. For each camera position and a given light source direction, the texture sample is rotated, however, without significant variation in scale or in-plane rotation [147].

The images in the database do not exhibit significant scale variation, hence, scale-invariant features tend to perform worse than features with just rotation or even no invariance but higher discriminative power [147]. The images are not suitable to sparse interest point-based methods. Apart from this, it is noticeable that multiple instances of the same texture are present for only a very few of the

materials, so intra-class variation cannot be investigated; and therefore, it is difficult to make generalizations [139]. The standard methodology on the CUReT database is to report results for 92 images per class [147]. Some images of this dataset do not have a sufficiently large portion of the texture visible to be cropped from the background. The measurement setup for CUReT database is not suitable for photometric stereo, as for a given texture sample at a certain orientation, it is necessary to capture the different images by fixing the position of camera and moving the light source, not by fixing the light source and moving camera.

The CUReT has three texture databases, namely -

- i. BRDF(bi-directional reflectance distribution function) database;
- ii. BRDF parameter database; and
- iii. BTF (bi-directional texture function) database.

#### *i. BRDF (bi-directional reflectance distribution function) database*

The BRDF database has 61 different samples for reflectance measurements. Each of these samples is observed with over 205different combinations of viewing and illumination directions.

#### *ii. BRDF parameter database*

The BRDF parameter database with fitting parameters from two recent BRDF models (Oren-Nayar reflectance model [150-151] for surfaces with isotropic roughness; and the Bidirectional reflection distribution function [152]for both anisotropic and isotropic surfaces) are developed. These BRDF parameters can be directly used for both image analysis and image synthesis.

#### *iii. BTF database*

Similar to the BRDF database, the BTF(bi-directional texture function) database also has 61real-world surfaces that are measured by using new texture representation called BTF [153]. A good survey on BTF is done by Filip and Haindl [154]. BTF is exploited by a number of researchers, e.g., [155-158].

The CUReT database combines the foreshortening effect of the texture and the associated changes in its corresponding illumination directions. The availability of

the CUReT offers rare but important databases to analyze the BRDF and BTF of textures.

### 3.4.8 Textile database

Another database is developed to detect defects in the textile products [13,159]. These are gray-level images of dimension 256x256, taken by a Sony CCD Iris SSC-M370CE camera in a laboratory environment. While acquisition of this database, frontal lighting system is employed, where the camera and the light source are placed on the same side of the fabrics. Each image corresponds to 8.53cm x 8.53cm fabric with a resolution of 3.33 pixels/mm, which is the same resolution that is required in the factory environment. In [13,159], the texture image sample set on which the experiments are performed contains images of eight different texture types. The first texture type has 35 images: 19 of them contain defects and the rest are defect-free or clean. Remaining seven sets of texture types contain four images per type, where two are defected and two are defect-free. The defects are considered according to the common defects.

### 3.4.9 UIUC database

The UIUC database [85,160,87,138,129-130,57-58,147,97] contains 40 images each of 25 different texture classes, hence total 1000 un-calibrated, unregistered images. These are gray-scale images having image resolution 640×480 pixels. It is available publicly in [131]. The database includes surfaces whose texture is due mainly to albedo variations (e.g., wood and marble), 3D shape (e.g., gravel and fur), as well as a mixture of both (e.g., carpet and brick) [85]. Moreover, within each class, viewpoint changes and scale differences are strongly evident. Illumination conditions are uncontrolled too for this database [85,147]. The database contains materials imaged under significant viewpoint variations and also contains fabrics which display folds and have non-rigid surface deformations [127]. Figure 3.4(b) shows some images for 25 classes.

It has significant changes in scale and viewpoint as well as non-rigid deformations; although with less severe lighting variations than CUReT [147]. Some additional variability issues are also considered, e.g., non-planarity of the textured

surface (bark), significant non-rigid deformations between different samples of the same class (e.g., fur, fabric, and water), inhomogeneities of the texture pattern (e.g., bark, wood, and marble), and viewpoint-dependent appearance variations (e.g., glass) [85]. The dataset has relatively few sample images per class but high intra-class variability, including non-homogeneous textures and unconstrained non-rigid deformations [85].

In terms of intra-class variations in appearance, this is the most challenging of the commonly used testbeds for texture classification [147]. Though the UIUC database demonstrates improvement over the CUReT textures in terms of the former's significant viewpoint variations and having some considerable surface deformations, it is much smaller than the CUReT database, both in the number of classes as well as the number of images per class [57]. Apart from this shortcoming, the UIUC database has very few instances of a given material so that it is difficult to perform categorization or figure out generalization properties of features [57]. In terms of scale and other viewpoint variations are concerned, the UIUC database is by far the most challenging database [57]. For example, the CUReT images have no scale variation (all materials are held at the same distance from the camera, only the orientation is changed), limited in-plane rotation, and the same physical surface patch is represented in all samples. Moreover, the appearance of each patch in that database is systematically sampled under different combinations of viewing angles and lighting directions, making it straightforward to select a fixed representative subset of samples for training, as is done in most CUReT evaluations [85]. Some methods [85,57,89,87] show comparatively better results on real world datasets, e.g., UIUC dataset [85] and CUReT databases [136].

### 3.4.10 CMU NRT database

The CMU near-regular texture (NRT) database [85,107,161] is a difficult database and it covers the spectrum of textures from completely regular to near-regular to irregular [107]. It also includes video of near-regular textures in motion. This database also includes test image sets with ground truth for translation,

rotation, reflection/glide-reflection symmetry detection algorithms. There are 15 top-level albums (89 total), covering 5775 images.

### **3.4.11 Rutgers skin texture database**

The Rutgers Skin Texture Database is a dermatology bidirectional texture function (BTF) database [92]. It has various skin diseases taken from the illumination and camera-controlled positions, containing bidirectional measurements of normal skin and of skin affected by various disorders. It has two combinations of viewing angles and illuminations: (i) four viewing and eight illumination positions; and (ii) three viewing and ten illuminations positions. Regarding measurement setup, this database has two different measurement setups, namely - (i) quartz halogen or fiber optic illuminator-based light arc; and (ii) camera mounted on a manually articulated arm on a tripod. They use a Sony DFW-V500 IEEE-1394 digital camera equipped with InfiniMini video lens with variable focus. The complete database contains more than 3500 images, and is made publicly available. The database has two components:

#### ***Normal skin component***

A normal skin component for recognition and rendering in computer vision and graphics has more than 2400 texture images of normal facial skin corresponding to 20 human faces, 4 locations on the face (forehead, cheek, chin and nose) and 32 combinations of imaging angles for each imaged surface (four camera poses, and eight light directions for each camera pose). The images in the database are acquired from both female and male subjects (7 females and 13 males), while the subjects age ranges from 24 to 53.

#### ***Skin disorder component***

A skin disorder component for quantitative imaging in dermatology has 75 clinical cases, which include conditions like acne, rosacea, psoriasis, seborrheic keratosis, dermatomyositis, actinic keratosis, congenital melanocytic nevus, keratosis pilaris, and dermatofibroma. Each case may correspond to a different patient and a different body location. Depending on the location of the disorder, there are images from the face, arms, legs, back and abdomen. Each case has been measured with

multiple texture images, and each image is characterized by a certain combination of imaging conditions.

### 3.4.12 KTH-TIPS database

The KTH-TIPS (Textures under varying Illumination, Pose and Scale) database [85,87,90,97,124,138,147,160] expands CUReT database by photographing new samples of ten of the CUReT textures at a subset of the viewing and lighting angles used in CUReT but also over a range of scales. Each class contains 81 images. Texture samples are 200x200 images except some sample of two classes (i.e., cracker and brown-bread). Images of the materials present in the KTH-TIPS database (which are also present in the CUReT database) are sandpaper, crumpled aluminium foil, styrofoam, sponge, corduroy, linen, cotton, brown bread, orange peel and cracker B.

These images are imaged at nine distances from the camera to give equidistant log-scales over two octaves [160]. At every direction, images are captured using 3 directions of illumination (i.e., front, side and top) and 3 poses (i.e., central, 22.5° turned left, 22.5° turned right), which provides a total of  $3 * 3 = 9$  images per scale, and  $9 * 9 = 81$  images per material [160]. It was created to extend the CUReT database in two directions: (i) by providing variations in scale as well as object poses and illumination directions, and (ii) by imaging other samples of a subset of its materials in different settings. Figure 3.6(a) shows some example frames of KTH-TIPS database.

Since the CUReT database contains little scale variation, this dataset is introduced which images ten CUReT materials at different distances, while also maintaining some change in pose and illumination [160]. As mentioned above, one of the key objectives of this database is to attempt to recognize ‘different samples’ of the CUReT materials. It is *not* possible to do recognize different samples with any acceptable degree of accuracy [160].

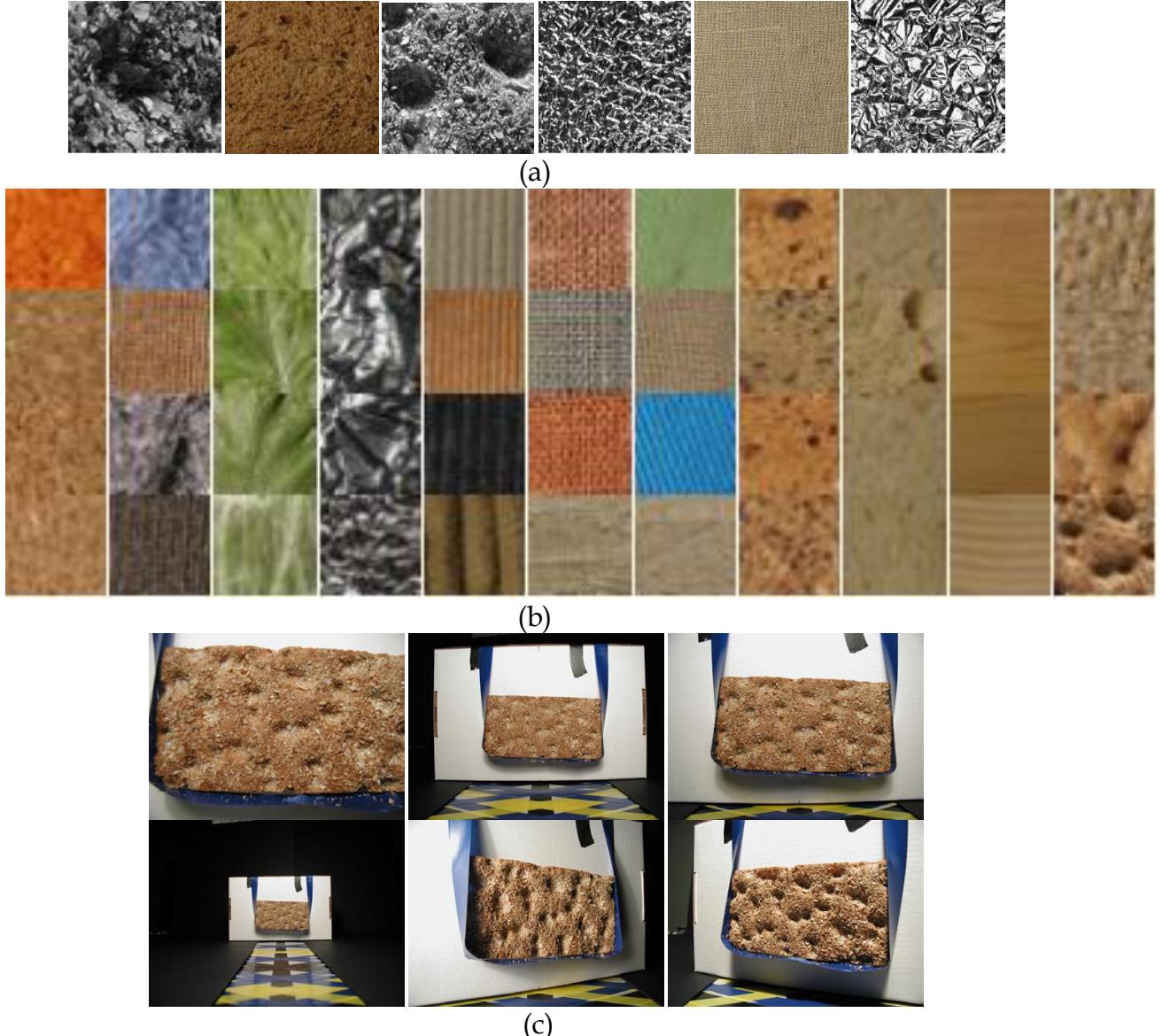


Figure 3.6: Some sample images from: (a) KTH-TIPS dataset [85,160]; (b) KTH-TIPS2 material database (each column represents (sequentially) cork, wool, lettuce leaf, aluminium foil, corduroy, linen, cotton, brown bread, white bread, wood and cracker; (c) An example that depicts the KTH-TIPS dataset's variations in terms of scale, pose and illumination [160].

### 3.4.13 KTH-TIPS2 material database

The KTH-TIPS2 material database [91,160,162] is built on the KTH-TIPS database and provides a considerable extension by imaging *multiple* different samples of different materials. Many of these materials have 3D structures, implying that their appearance can change considerably as pose and lighting are changed. It has all types of images of the KTH-TIPS, in addition to ‘lettuce leaf’ image. Figure 3.6(b) shows some images for this dataset. The database contains images at 9 scales,

equally spaced logarithmically over two octaves. It has 3 poses and 4 illumination conditions (frontal, 45°from the top and 45°from the side, all taken with a desk-lamp with a Tungsten light bulb) (e.g., Fig. 3.6(c)), and for the fourth illumination condition fluorescent lights in the laboratory are considered. Although some variation in pose and illumination is present, both KTH-TIPS and KTH-TIPS2 contain significantly fewer settings for lighting and viewing angle than CUReT [91].

#### **3.4.14 PerTex database**

PerTex database is developed by Fraser Halley [127]. The PerTex database has 334 photometric height maps with perceptual similarity matrix.

#### **3.4.15 Building texture database**

The Building Texture Database [163] has blocks, finding a mesmerizing abstraction in the buildings' facades. Some of the structures in the series are photographed without reference to the context of sky or ground, and many buildings are seen in a state of repair or construction.

#### **3.4.16 Grain mixtures dataset**

The Grain Mixtures [164-165] used images of eleven different mixtures of rice and barley grain. Image resolution is 128x128 pixels. Both of these were rather similar in size, rice was a little more elongated than barley and barley had a wider range of gray levels than rice.

#### **3.4.17 Kylberg texture dataset**

The Kylberg texture dataset [166-167] has a number of textured surfaces, including fabrics and surfaces of stone, were imaged in the local surroundings. Textured surfaces were also arranged using articles such as rice grains, sesame seeds and lentils.

### **3.5 Dynamic Texture Database**

The above-mentioned databases are based on static images. However, there are some new types of datasets called dynamic texture dataset where the temporal textures are variable and changing temporally. Now what is the meaning of 'dynamic texture'? It is known that for a single image, a texture can be defined as a

realization from a stationary stochastic process with spatially invariant statistics [2], which portrays the intuitive notion of texture. For a sequence of images (i.e., time-varying texture), individual images are clearly not independent realizations from a stationary distribution, for there is a temporal coherence intrinsic in the process that needs to be captured. The underlying assumption, therefore, is that individual images are realizations of the output of a dynamical system driven by an independent and identically distributed (IID) process.

Dynamic textures are sequences of images that exhibit some form of temporal regularity [168]. According to Yuan et al. [169], dynamic texture can be defined as a temporally continuous and infinitely varying stream of images that exhibit certain temporal statistics. In another term, dynamic textures are video sequences of images of moving/non-rigid scenes that exhibit certain stationarity properties in time; these include sea-waves, smoke, foliage, whirlwind, etc. [170]. Another definition of the term dynamic texture is usually used with reference to image sequences of various natural processes that exhibit stochastic dynamics [171].

There are some dynamic texture databases [172], which are more difficult to analyze. Between computer graphics and vision researchers, dynamic texture analysis or synthesis texture analysis become very popular areas [169]. Dynamic texture analysis is almost new steps for video sequence analysis [169]. A number of recent works have concentrated on dynamic textures and their evaluations, segmentation, synthesis, and recognition [44-45,73,84,137,168-169,171,173-192]. The following sub-sections present the benchmark datasets for dynamic or temporal texture databases.

### 3.5.1 UCLA dynamic texture database

The UCLA dynamic texture database [137,168,174,178,180,183-184] contains 50 classes of various dynamic texture video textures, including boiling water, fountains, fire, waterfalls, rippling water, and plants and flowers swaying in the wind. Each class contains four gray-scale sequences with 75 frames of 160×110 pixels. Each sequence was clipped to a 48×48 window that contained the representative motion. For each scene, all four example sequences are captured with

the same viewing parameters (e.g., identical viewpoint). In total there are 200 sequences. Reference [44] obtains the recognition results of 95.6% for this database, whereas Ref. [174] achieves 97.5% accurate classification rate. The UCLA dataset is at present the benchmark for dynamic texture recognition. However, a bigger and more dissimilar database (the DynTex database [84,190]) available [44]. The UCLA dataset hold the position of benchmark because to the following properties [44]:

- i. These database have dynamic texture sequences by now it has pre-processed from their raw form.
- ii. Only a single dynamic texture is present in each dynamic texture sequence.
- iii. In each dynamic texture sequence, no panning or zooming is performed.
- iv. Ground truth labels of the dynamic texture sequences are provided.

### 3.5.2 UCLA-pan database

A second database containing panning video textures is produced from the original UCLA video textures [175]. Each video texture is generated by panning a  $40 \times 40$  window across the original UCLA video. In this case, four pans (two left and two right) are generated for each video sequence, resulting in a database of 800 panning textures, which is called the UCLA-pan database [175]. The motion in this database is composed of both video textures and camera panning, hence the dynamic texture is not expected to perform well on it. Details of these UCLA dynamic texture databases are available in Ref. [175].

### 3.5.3 DynTex database

The European FP6 Network of Excellence MUSCLE develops a dynamic texture database called DynTex [84,171,190-191]. The DynTex dataset contains more than 650 varied dynamic texture videos, but the information about the type of textures shown in the sequences is not provided for all the videos in the set. The image size is  $352 \times 288$  and the compressed videos provided are coded using DivX codec. A subset of 202 sequences, spanning some 23 classes of very varied dynamic textures has been selected to evaluate the method of [171]. Some of the DynTex sequences, such as mixtures of essentially different processes, are obviously inhomogeneous and non-stationary. Examples of some texture classes of this

database are – textile, vegetation, grass, NA, streaks, water, steam, fire, smoke, branch, cloud, leaf, car, needle, fur, tentacle, insects, CD, foam, light and paper.

Central to the database is the so-called golden set of high-quality dynamic texture sequences that satisfy all criteria of the acquisition protocol [191]. This set can be further structured by its underlying physical processes, e.g., turbulent motion, waving motion, discrete units, etc. Additionally, they will be classified by their shot type, i.e., either as close-up (shot consisting entirely of the dynamic texture, no segmentation required), or as context (the dynamic texture is shown in its context) [191]. Next to the golden set we will offer various other sets. Examples are a collection of dynamic texture sequences recorded with a moving camera, and a collection with several dynamic textures per sequence [191]. However, the DynTex database lacks the four key properties of UCLA dynamic texture database. It is exploited by various researchers, e.g., [44-45,181]. More details on this dataset are available in [191]. Figure 3.7(a) shows five frames for five sequences of this dataset, namely tree sequence, river sequence, grass and river sequence, wave sequence, and field sequence.

### 3.5.4 Sainsan's dynamic texture database

The dynamic texture database is developed by Saisan et al. [174]. It has a total of more than 250 sequences, consisting of moving scenes of smoke, boiling, fire, flowers, sea, water, fountain and waterfall. Each sequence is 150 frames long and pixels. Each sequence can be divided into two sub-sequences of 75 frames for a total of more than 500 sequences. Included in the database are similar sequences with different dynamics. For example, it has water stream, recorded from different angles, thus moving at different orientations and speeds.

The database includes 76 different kinds of dynamic textures. Each of them is represented by 8 distinct instances. Each sub-sequence consists of 75 frames. All frames are in color where the size of an individual frame is 220x320 pixels. It is used by [44,174-175,182]. Reference [175] achieved recognition results with this dataset as 97:5% by using the kernel version of the dynamic texture model and 96:5% by using just the Martin distance on the LDS parameters. Reference [182] achieved 100%

recognition results for eight classes of this database. Note that the fountain and waterfall classes are the most difficult to classify because there are no clear appearance differences.

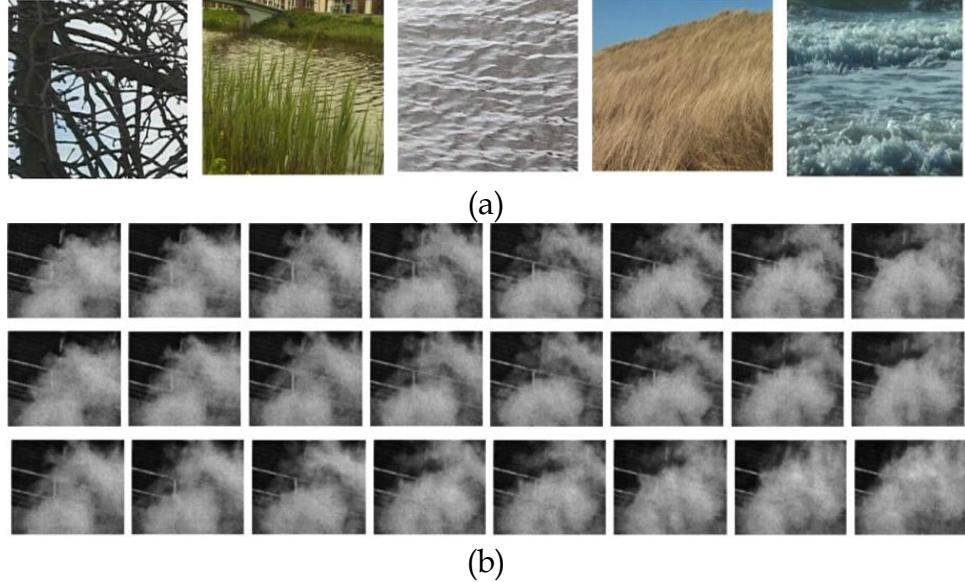


Figure 3.7: (a) DynTex database – few sample frames (sequentially – tree sequence, river sequence, grass and river sequence, wave sequence, and field sequence) [84,190]; (b) MIT temporal texture database – few frames from ‘smoke’ [170,176].

### 3.5.5 DynTex++ database

The UCLA benchmark dynamic texture dataset lacks much variety along the three DT dimensions, and therefore, [44] proposed a new dynamic texture dataset, called DynTex++ dataset. The goal here is to organize the raw data in the DynTex database in order to provide a richer benchmark for future dynamic texture analysis. The original database is already publicly available; however, only the raw AVI videos are provided. These sequences are filtered, pre-processed, and labeled. While DynTex contains a total of 656 video sequences, DynTex++ uses only 345 of them.

This dataset has 36 classes. It means that some sequences are eliminated that contained more than one dynamic texture, contained dynamic background, included panning/zooming, or did not depict much motion. They were not uniformly distributed among the classes. Reference [44] obtained an average recognition result of 63.7%. It is a challenging database.

### 3.5.6 MIT Szummer dataset

MIT Szummer dataset or Temporal Texture dataset [45,73,173,176-177,179] is a small dataset. The quality of the data of this dataset in terms of image resolution, contrast, motion stability is quite poor [175,181] compare to Ref. [84]. Classification results for the 10 Szummer sequences [176] are accomplished by [177]. Temporal textures are textures with motion. The dataset consists of commonly occurring temporal textures, such as river-near (close-up of water), river-far (wide-angle shot of water), steam (steam from manhole), boil-heavy (vigorously boiling water), boil-light (lightly boiling water), plastic (windblown plastic sheet), and toilet (swirling water in toilet) [179]. The image sequences of temporal textures are recorded using a Hi-8 video camera and a tripod is used to ignore any camera motion. The resolution is about 170x115 and the length is 120 frames. This is tough dataset because fluid surfaces are highly specular, and in some cases, perspective chirping is prevalent. The 'toilet' image sequences have spiral motion. Figure 3.7(b) shows some sequential form ahead.

## 3.6 Discussions

In the previous sections, we categorize various texture datasets and pointed important issues regarding their merits, constraints, few comparative discussions. In this section, we summarize some key properties of important textures databases in Table 3.1. We can notice that in terms of number of classes and variability, there are diverse databases. A number of databases are not developed in controlled and organized manner. For example, the Brodatz database is not organized and scanned from copyrighted book, even though it is the most widely used texture dataset.

We can notice at the same time that some databases are available with different image sizes, as the original images are down-scaled for smaller image size. Regarding color or gray-level issue, most of the databases are available in color image formats. As shown in the Table 3.1, we notice that cameras or sensors are diverse as well. In some cases, various properties are not available in the original paper or respective websites (e.g., in Table 3.2) – therefore, it is not possible to put all properties comprehensively.

**Table 3.1:** Various properties of important texture databases

Database	Image Size	No. of Class or Images	Gray-level or Color	Image Rotation	Surface Rotation
Brodatz	512x512	112 classes	Gray-level	Yes	Yes
Columbia-Utrecht Reflectance and Texture (CUReT)	512x512 or, 200x200	10 classes, usually 92 images per class		Yes, multi-view	Limited in-plane rotation
Vision Texture (VisTex)	786x512 or 512x786	167 images	Color	Yes	No
MEASurement of TEXture (MeasTex)	512x512; also in raw PGM format	4 classes, total 944 images		Yes	No
Photometric Texture (PhoTex)	1280x1024		In various formats and in SCOPE format	Constant view	No
OUTex	516x716	319 textures, 162 images per texture	Color and gray-level images	Yes, 6 different resolutions	9 rotation angles
MRI Brain					
Digital Database for Screening Mammography (DDSM)	3000x4500	approximately 2,620 cases			
UCLA dynamic texture database	160x110	50 classes, 75 frames per class		Camera panning	
Dynamic Texture (DynTex)	220x320	656 sequences, each has 150 frames	Color	Inhomogeneous	Different orientations and speeds
DynTex++		36 classes, total 345 sequences	Video	Varied	
MIT Szummer	170x115	7 classes, 120 frames per class		No camera motion	Perspective chirping, spiral
UIUC database	640x480	25 classes, 40 images per class		View-point change	Scale difference
Texture library	Varied but high-resolution	17 classes, total 928 images	Color	No	No
CMU Near-Regular Texture (NRT) Database	Image and videos	89 classes, total 5775 images			

Textures under varying Illumination, Pose and Scale (KTH-TIPS) database	1280x960 or, 200x200	10 classes, 108 Images per class	Color and gray-level	9 different scales	3 in-plane orientations
KTH-TIPS2 material database	1280x960	44 classes, total 4608 images; also have another set of 11 classes	Gray-level; have color image of 200x200 pixel patches	Variability	Varied pose
Amsterdam Library of Textures (ALOT) database		250 classes, 100 images per class; Also, 12 classes (27500+ images)		3 orientations	4 view-points
UMD Texture database	1280x900	25 classes, 40 image per class		No	No
Kylbergtexture	576x576	28 non-rotated classes, 4 images per class; 28 classes with 12 rotations, 1920 images per class		Have some rotated images	No rotation; 12 rotations

**Table 3.2:** Database and respective websites or comments

Database	Website	Contact Group/Person	Comments
Columbia-Utrecht Reflectance and Texture (CUReT)	<a href="http://www.cs.columbia.edu/CAVE/software/curet/html/about.php">http://www.cs.columbia.edu/CAVE/software/curet/html/about.php</a>	Columbia University and Utrecht University	3 databases: (i) BRDF(bi-directional reflectance distribution function) database; (ii) BRDF parameter database; and (iii) BTF (bi-directional texture function) database.
VisTex	<a href="http://vismod.media.mit.edu/vismod/imagery/VisionTexture/">http://vismod.media.mit.edu/vismod/imagery/VisionTexture/</a>	MIT Media Lab	Images are grouped according to their contents; Have traditional and non-traditional textures
MeasTex	<a href="http://www.texturesynthesis.com/meastex/meastex.html">http://www.texturesynthesis.com/meastex/meastex.html</a>		
PhoTex		Heriot-Watt University at Edinburgh	

OUTex	<a href="http://www.outex.oulu.fi/index.php?page=outex_home">http://www.outex.oulu.fi/index.php?page=outex_home</a>	University of Oulu, Finland	
Digital Database for Screening Mammography (DDSM)	<a href="http://marathon.csee.usf.edu/Mammography/Database.html">http://marathon.csee.usf.edu/Mammography/Database.html</a>	UCF	Possibly the best dataset on textures in medical texture analysis
DynTex	<a href="http://projects.cwi.nl/dyntex/database.html">http://projects.cwi.nl/dyntex/database.html</a>	European FP6 Network of Excellence MUSCLE	Dynamic texture
PerTex	<a href="http://www.macs.hw.ac.uk/texturelab/resources/databases/">http://www.macs.hw.ac.uk/texturelab/resources/databases/</a>	Fraser Halley	334 photometric height maps with perceptual similarity matrix

**Table 3.3:** Comparison among some photometric texture databases

Database	Image rotation	3D surface rotation	Illumination	Photometric stereo
Brodatz	Yes	No	Varied	Unregistered
CUReT	Yes	Yes	Controlled	Unregistered
OUTex	Yes	Yes	Controlled	Unregistered
PhoTex	Yes	No	Controlled	Registered
PMTex	Yes	Yes	Controlled	Registered
VisTex	Yes	No	Controlled	Unregistered
MeasTex	Yes	No	Uncontrolled	Unregistered

**Table 3.4:** Comparisons in terms of illumination and camera/ sensors

Database	Illumination	Camera/Sensors
Brodatz	Varied illuminations as photos are taken in different time and places	Scanned images from a book
CUReT	Controlled but 7 illumination directions	Fixed camera with tripod
VisTex	Mixed (daylight, artificial-florescent, artificial-incandescent, & some are imprecise)	Camera (having frontal-plain and oblique settings)
MeasTex	Varied	35mm camera film, then scanned
PhoTex	40 different illumination directions	Vosskuhler CCD 1300LN digital camera
OUTex	3 illuminations	Sony DXC-755P three chip CCD camera attached to a GMFanuc S-10, a 6-axis industrial robot
MRI Brain	Constant	1.5-T Siemens scanner
DDSM	Constant	Mammograms
MIT Szummer	Varied	Hi-8 video camera

Texture library	Varied	Digital camera
KTH-TIPS	4 illuminations	Olympus C-3030ZOOM digital camera
KTH-TIPS2	3 illuminations	Olympus C-3030ZOOM digital camera
A LOT	8 different illuminations	Sigma SD10, SD9 (c3) Foveon X3 3CMOS camera

Table 3.3 presents some comparative properties for datasets that are photometric texture database [135]. In Table 3.4, we pointed the illumination conditions and camera or sensor details. This Table demonstrates the variation on lighting conditions and cameras, from which we get the notion that it is not easy at all to compare all datasets in single platform. Considering the above-mentioned databases, the future direction could be in the 4D dataset [193]. The dynamic databases are also not well-organized and lacking a good number of classes. So far, the recognition approaches covering the dynamic textures are very few. Therefore, a more in-depth analysis in this arena is crucial and challenging.

Though there are a few well-known databases related to medical texture analysis, several more datasets are presented in some papers though these are not available and not even well-structured. We recommend that some better datasets on texture in medical image processing are required. In future, we are expecting to have a few more challenging datasets for static and dynamic texture. But it is an essential part for the researchers to do comparative researches on the existing datasets by employing various related methods. Note that this task is not trivial and a challenging one due to the fact that these datasets are varied in terms of varieties in classes, per class instances, image resolutions, lighting conditions and angles. Therefore, all datasets may not be covered to compare. The ranges of applications for most of the datasets are understandable from their names, and added discussions provide these issues as well. We feel that the above-mentioned criterions and nomenclature are helpful for further analysis and evaluations.

### **3.7 Conclusions**

This chapter is about texture databases. We convincingly categorize texture databases, based on many references. In this survey, we put a nomenclature to split these texture datasets into few basic groups and later put related datasets. This chapter presents an in-depth discussion and analysis on texture datasets. It points important comparative issues and parameters after a comprehensive presentation of the datasets.

We categorize various texture datasets and pointed important issues regarding their merits, constraints, and important comparative discussions. We summarize some key properties of important textures databases in Table 3.1. We can notice that in terms of number of classes and variability, there are diverse databases. A number of databases are not developed in controlled and organized manners. For example, the Brodatz database is not organized well and it is based on images, which are scanned from a copyrighted book. Even though it is the most widely used texture dataset, it is not developed through a controlled and organized approach.

From the Tables, we can notice at the same time that some databases are available with different image sizes, as the original images are down-scaled for smaller image sizes. Regarding color or gray-level issue, most of the databases are available in color image formats. Rotation-invariant texture classification is essential for lots of applications [194-195]. There are some image rotation-based datasets available.

One query may appear on what kind of texture datasets seem more important for future research. According to our analysis, it is case-specific and application-dependent. For medical image analysis, the perspectives and necessity are different and crucial respectively. For various material texture analyses, it is necessary to understand textures for applications in industry, automations and robotics. Understanding an image surface or a material is one of the primitive stages to evaluate a scene for a robot or intelligent system for applications on material evaluation, human action understanding, object identification and recognition, etc.

In future, we need to develop few more large datasets, especially in the field of medical image analysis. Also, more comparative analysis on various features on different datasets will provide strong cues for better applications.

# Analysis of Features Based on Co-occurrence Image Matrix

## Chapter 4

---

### 4.1. Introduction

Texture and material understanding and recognition are very important tasks in the area of computer vision, medical image processing, image processing and computer graphics [8,102,196]. In these fields, we try to understand and evaluate image textures and its features. Feature extraction is important and it is a method of capturing visual content of images for indexing & retrieval. These are crucial for various applications in image understanding, medical image processing, image surface analysis, image defect analysis, robotics, etc. [5,7-8,13,101,159]. A machine-vision system usually cannot perform to understand an object's surface like a human being can. A computer machine-vision system usually is unable to identify an object based on its surface of similar patterns.

Distinguishing an objects surface, from rough or smooth, clean or dirty, liquid or solid, even soft or hard is a very daunting task even for a smart machine-vision system. Hence, it is essential to understand the differences of similar textures and their variations on various image features. To accomplish this task, we need to do in-depth analysis on similar images on standard datasets and find out their characteristics on different conditions and orientations. We consider gray-level images for computing their co-occurrence matrices and on the top of those matrices, eleven different features are computed. These are statistical image features, which

are computed based on the co-occurrences on neighborhood pixel values and their respective relationships.

From our analysis, we hypothesize various points that will be depicted in the following parts of this chapter. This kind of in-depth analysis on co-occurrence matrices with eleven statistical image features are not accomplished as per our study. The chapter is organized as follows: after the introduction in Section 1, we present background of this work along with related works in Section 2. In Section 3, we present the gray-level co-occurrence matrix and explain its orientation. Image features are presented in this Section. In the next section (Section 4), we illustrate the experimental texture dataset with varied complexities step by step. We evaluate the performances of these eleven features. Finally, we conclude the chapter in Section 5.

## 4.2 Co-occurrence Matrix-based Image Features

In this section, we concentrate on 2<sup>nd</sup> order statistics and the corresponding features based on this. We compute gray-level co-occurrence image matrices from gray-level images. From these matrices, we extract 2<sup>nd</sup> order statistical texture features, instead of having these directly from images. The difference between a gray-level image and the corresponding gray-level co-occurrence matrix is that the matrix has equal number of rows and columns, and it is equal to the number of distinct gray-levels or pixel values in that image. It is a matrix that shows the frequency of one gray-level appearing in a specified spatial linear relationship with another gray-level within the field of inspection [41,197].

Let us consider that an image contains  $N$  gray-levels from 0 to  $N-1$ . Also, consider that  $f(m, n)$  is the intensity at sample  $m$ , line  $n$  of the neighborhood, then we

can have the following co-occurrence matrix from the gray-level pixel values of the image,

$$p(i, j | \Delta x, \Delta y) = W Q(i, j | \Delta x, \Delta y)$$

where,

$$W = \frac{1}{(M - \Delta x)(N - \Delta y)}$$

$$Q(i, j | \Delta x, \Delta y) = \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} A$$

where,

$$A = \begin{cases} 1 & \text{if } f(m, n) = i \text{ and } f(m + \Delta x, n + \Delta y) = j \\ 0 & \text{otherwise} \end{cases}$$

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

(a) A 4x4 image.

$i/j$	0	1	2	3
0	$\#(0,0)$	$\#(0,1)$	$\#(0,2)$	$\#(0,3)$
1	$\#(1,0)$	$\#(1,1)$	$\#(1,2)$	$\#(1,3)$
2	$\#(2,0)$	$\#(2,1)$	$\#(2,2)$	$\#(2,3)$
3	$\#(3,0)$	$\#(3,1)$	$\#(3,2)$	$\#(3,3)$

(b) The number of co-occurrences of pixel  $i$  to the neighboring pixel value  $j$ .

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



2			

(c) Calculation for co-occurrence matrix for horizontal orientation.

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



2	...		
			1

(d) Another step for computing the co-occurrence matrix.

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

(e) The co-occurrence matrix for the image in (a).

Figure 4.1: An example of the computation of gray-level co-occurrence matrix from (a). Sub-Tables sequentially presents the computation of the elements of the co-occurrence matrix for the 4x4 image, as in (a).

Figures 4.1 and 4.2 show co-occurrence matrices  $P_{ij}$  with distance or radius  $\delta = 1$  and horizontal direction  $\theta = 0^\circ$ . Figure 4.1 has four pixel values – 0, 1, 2 and 3. Hence, the size of the corresponding co-occurrence matrix will be 4x4. The relative distance ( $\delta$ ) between the pixel pair is measured in pixel number. The distance can be 1 or more based on the considered neighbor distance of pixels. And the orientation or angle can be one of the eight options centering each pixel.

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

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

0.16	0.08	0.25	0
0.16	0.08	0	0
0	0.08	0	0.08
0	0.08	0	0

Figure 4.2: An image patch of size of  $4 \times 4$  and the corresponding co-occurrence matrix as shown in the mid-table. A normalized matrix is shown here which can be computed after dividing each value by the total co-occurrence values (e.g., in this case, each value is divided by the total of 12. Hence, the 1<sup>st</sup> cell in the normalized matrix is  $2/12=0.16$ ).

For example, the relationship  $\delta = 1$  and  $\theta = 0^\circ$  is the nearest horizontal neighbor. There can be  $(N - 1)$  neighboring pixel pairs for each row and there are  $N$  rows, providing  $R = (N - 1)N$  nearest horizontal pairs. The co-occurrence matrix can be normalized by dividing each of its entry by  $R$ . In Fig. 4.1, the normalized matrix is computed by dividing each value by 12. Moreover, there can be separate co-occurrence matrices for vertical direction and both diagonal directions, where directional angle  $\theta$  can be  $(45^\circ, 90^\circ, 135^\circ)$ . If the direction from bottom to top and from left to right is considered, there will be eight different directions, as shown in Fig. 4.3.

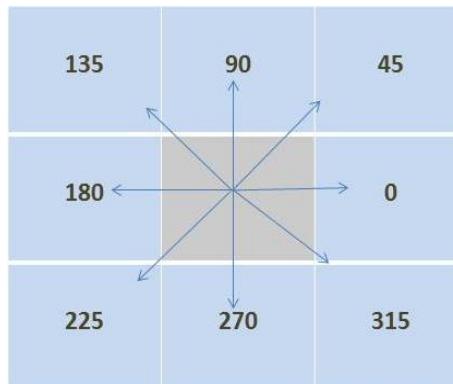


Figure 4.3: Eight different directions of adjacency for the calculation of co-occurrence matrices. In this thesis, we consider  $\theta$  as horizontal ( $0^\circ$ ), front diagonal ( $45^\circ$ ), vertical ( $90^\circ$ ) and back diagonal ( $135^\circ$ ).

However, these gray-level co-occurrence matrices can only capture properties of a texture but they are unable to produce differentiable properties directly and cannot be functional and effective to provide information directly. Hence, we need to compute statistical features based on these matrices, for example, to compare textures or images. Therefore, numerous features should be exploited and computed from the co-occurrence matrix that can be used to represent the texture more compactly. This is not a trivial task unless we have a robust analysis of various statistical features on distinctive as well as closely-related textures.

Reference [196] presents various features. In this thesis, we target eleven different features, which are computed from co-occurrence matrix. A number of features are extracted for an image from co-occurrence matrix and different applications exploited a few of these randomly. It is very essential to understand image features and their respective rational and impact on orientation. This chapter attends this task that is based on a standard dataset created for evaluating eleven different features. Textures can be rough or smooth, vertical or horizontal. In our dataset, these varieties of dimensions are considered. These features are presented as follows:

#### *F1. Angular Second Moment (ASM) feature*

The Angular Second Moment (ASM) is known as energy or uniformity or uniformity of energy. It measures the uniformity of an image. When pixels are very similar, the ASM value will be large. For an image of single color of no variation, the ASM values for different angles are the Angular Second Moment takings the sum of squared component in the matrix. It is calculated according to the following equation,

$$ASM = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p_{ij}^2$$

## *F2. Contrast feature*

Contrast can be defined as local intensity or gray-level variations between the reference pixel and its neighbor. In the visual perception, contrast is considered by the variance in the cheeriness and color of the object. The contrast returns a measure of the intensity contrast between a pixel and its neighbor over the entire image. It can be recognized as variance and inertia. The contrast is zero when the neighboring pixels have constant values. For example, for an image of single color of no variation, the contrast values for different angles are 0. It is calculated as,

$$\text{Contrast} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (i - j)^2 p_{ij}$$

## *F3. Entropy feature*

It is difficult to define the term entropy. The idea comes from thermodynamics. The entropy means the amount of energy that is eternally lost to heat every time. The entropy cannot be improved to do useful work. Because of this, the term can be understood as amount of irremediable chaos or disorder. Hence, in other words, the entropy can measure disorder or complexity of an image. We know that entropy demonstrates randomness and the corresponding statistical variations, which can be used to define the texture. It is maximal when all elements are equal. When an image has high entropy, we can understand that it has a great contrast from one pixel to its neighbor pixel. Complex textures tend to have high entropy. However, for an image of single color of no variation, the entropy values for different angles are 0. The equation of entropy is:

$$\text{Entropy} = -\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p_{ij} \times \log(p_{ij})$$

#### *F4. Variance feature*

The variance is a measure of the dispersion of the values around the mean of combinations of reference and neighbor pixels. It is similar to entropy. The variance explains the dispersion of the difference between the reference and the neighbor pixels in a kernel or image block. For an image of single color of no variation, the variance values for different angles are 0. The variance is -

$$\text{Variance} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (i - \mu)^2 \times p_{ij}$$

#### *F5. Correlation feature*

The correlation feature (i.e., the Pearson product-moment correlation coefficient, or Pearson's correlation coefficient) shows the linear dependency of gray-level values in the co-occurrence matrix. The correlation feature can define the relationship of a pixel with its neighbor in terms of correlation among them in the entire image. This feature can present how a reference pixel is related to its neighbor, 0 is uncorrelated, and 1 is perfectly correlated. In other words, it can measure the joint-probability occurrence of the specified pixel pairs. The correlation will be high if an image contains a considerable amount of linear structure. It can be measured as,

$$\text{Corr} = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \frac{(i - \mu_i)(j - \mu_j)}{\sqrt{(\sigma_i^2)(\sigma_j^2)}} p_{ij}$$

#### *F6. Inverse Difference Moment (IDM) feature*

The Inverse Difference Moment (IDM) feature can determine any local homogeneity or a sense of uniformity of an image. It can also be termed as homogeneity. This feature provides a sense of the measures of the closeness of the

distribution of the matrix elements to its diagonal. Homogeneity and contrast can be related. A maximum value can be achieved if an image is found with same value for all elements in the image. For an image of single color of no variation, the inverse difference moment feature values for different angles are 1. It provides a strong response at the central locations of the features of interest.

The IDM is computed according to the following equation,

$$IDM = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \frac{1}{1+(i-j)^2} p_{ij}$$

#### *F7. Sum Average feature*

The sum average feature is computed based on the following equation. For an image of single color of no variation, the sum average values for different angles are 2. Usually, for an image of varied pixel values, the sum average is high value. For example, an image of grass field has value for sum entropy feature of typically a few hundred. It can be calculated as,

$$\sum_{Average} = \sum_{i=0}^{2N-2} (i) \times p_{x+y}(i)$$

where,

$$P_{x+y}(k) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p(i, j)$$

and,

$$k = i + j = \{0, 1, 2, \dots, 2(N-1)\}$$

### F8. Sum Variance feature

The sum variance is computed based on the sum average feature. Among the all feature values we are computing here, the sum variance feature produce the highest value for each image. As per our computation for 18 different grass-land images, the *mean* sum variance is about 1 million, whereas the sum average is about 200 only. The equation for sum variance feature is given below:

$$\sum_{SumVariance} = \sum_{i=0}^{2N-2} p_{x+y}(i)(i - f_7)^2$$

where,

$$f_7 = \sum_{Average} = \sum_{i=0}^{2N-2} (i) \times p_{x+y}(i)$$

### F9. Sum Entropy feature

As per the definition of entropy (as mentioned above for F3), the value goes higher for an image of more variations. The sum entropy is about 8 only for 18 different grass-land images, where the *mean* sum variance is about 1 million, and the sum average is about 200. The average entropy (F3) of the same 18 images is about 13. For one color image having no variations in pixel values, the sum entropy is also zero. It can be computed based on the following equation:

$$\sum_{entropy} = \sum_{i=0}^{2N-2} p_{x+y}(i) \log(p_{x+y}(i))$$

#### *F10. Information Measures of Correlation feature – 1*

The Information Measure of Correlation – 1 (IMC1) is measured based on the equation below. It has invariance property. It is based on the calculations of entropy values of  $p_x$  and  $p_y$ . For an uniform image (i.e., no pixel changes in that image), the IMC1 is zero. It produces negative values.

$$IMC1 = \frac{- \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p_{ij} \log_2 p_{ij} + \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p_{ij} \log_2 (p_x(i)p_y(j))}{\max \{ \text{entropy}(p_x) \text{entropy}(p_y) \}}$$

#### *F11. Information Measures of Correlation feature – 2*

The Information Measure of Correlation – 2 (IMC2) is measured based on the equation below. Similar to the IMC1, the IMC2 also has invariance property. For an uniform image (i.e., no pixel changes in that image), the IMC2 is zero. It produces values in the range of 0 to 1.

$$IMC2 = \left\{ 1 - \exp \left[ -2 \left( \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p_x(i)p_y(j) \log(p_x(i)p_y(j)) + \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p_{ij} \log_2 p_{ij} \right) \right] \right\}^{1/2}$$

### **4.3Experimental Results and Analysis**

In this section, we demonstrate the experimental results based on a standard dataset that covers twenty-one different textures – from uniform values to very much random structure at the end. The purpose of this dataset is to evaluate the image features computed based on the respective gray-level co-occurrence matrices. Figure 4.4 shows the images of the standard dataset.

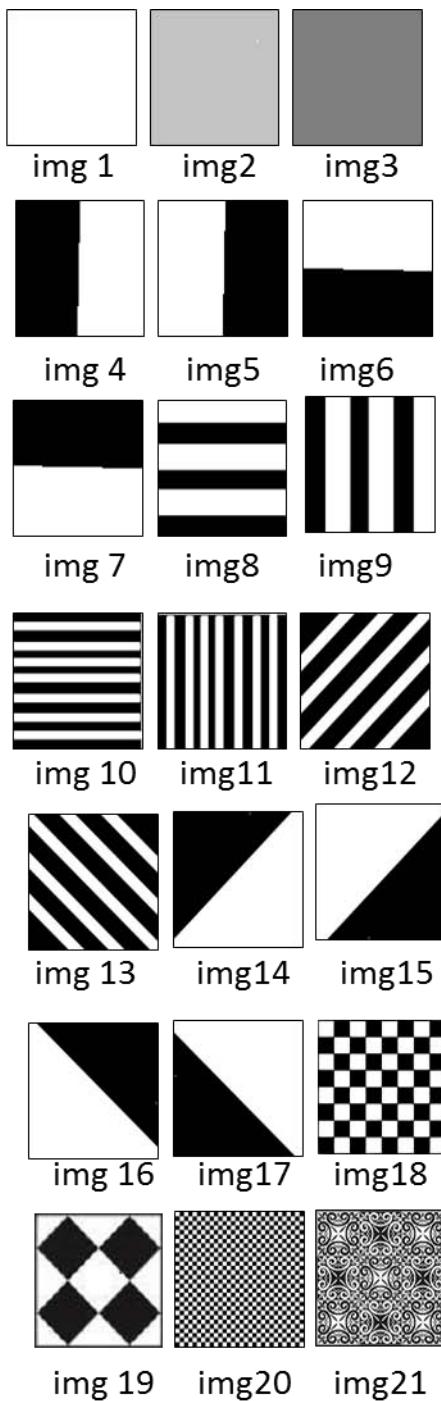


Figure 4.4: Standard dataset of twenty-one texture images in gray-scale.

In this dataset, we can notice that it starts with complete white (all 255 values) image (image#1). It has two other gray-level images (image#2 and image#3) having uniform single value all around. The next four images (image#4 till image#7) have

almost half as black and half as white in four orientations. Similarly, image#14 to image#17 have similar patterns of almost half in black and rest are in white. These mirror images are useful to understand an image feature. Image#8 till image#13 has white-black patches in varied number and different directions, for example, horizontal, vertical and diagonal. Image#18 is a chessboard image that is similar to image#20 (having densed-chessboard pattern). Image#19 is diamond-shape and depicts hugely varied structure than others. The last image is also very complex of different combinations of textures.

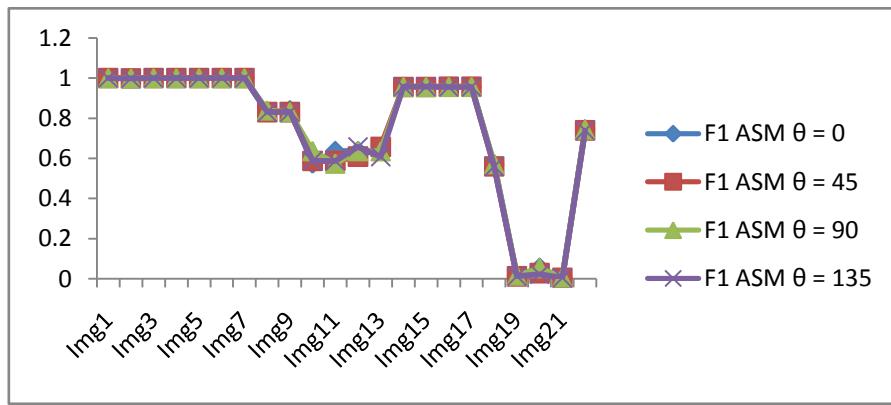


Figure 4.5: ASM for four angles: It is noticeable that the ASM value is higher for img#1 to img#7 than others. This is due to their uniform texture elements and the pixels are very similar in parts; whereas, img#19~21 has lowest value due to varied texture patterns.

Figure 4.5 shows the angular second moment for all 21 texture images and finally the last one is for the average values for ASM values for each of the four directions. It is visible from these that ASM is almost invariant to directions. Only for image image#10~13 (having more black-white bars in horizontal, vertical and front-and back-diagonal directions) and image#20 (densed-chessboard type) show a bit angular-variant characteristics. Also, first seven images have highest value (one) for

ASM, which demonstrates that these images have no variations. The most variations are visible for the last 4 images having more variations.

In the similar fashion, the later graphs for other ten features also demonstrate the characteristics of feature values in four angular directions ( $\theta = 0, 45, 90, 135$ ) for radius  $\delta=1$ . The last parameter is the mean value (denoted as 'Avg.') of each feature for each direction. So, we compute twenty-two different values for each feature in four orientations.

From Fig.4.6, we get the contrast values and it shows that contrast is zero or almost zero when the neighboring pixels have constant values. For other images, contrast values are higher when they have variations. It is angular-variant.

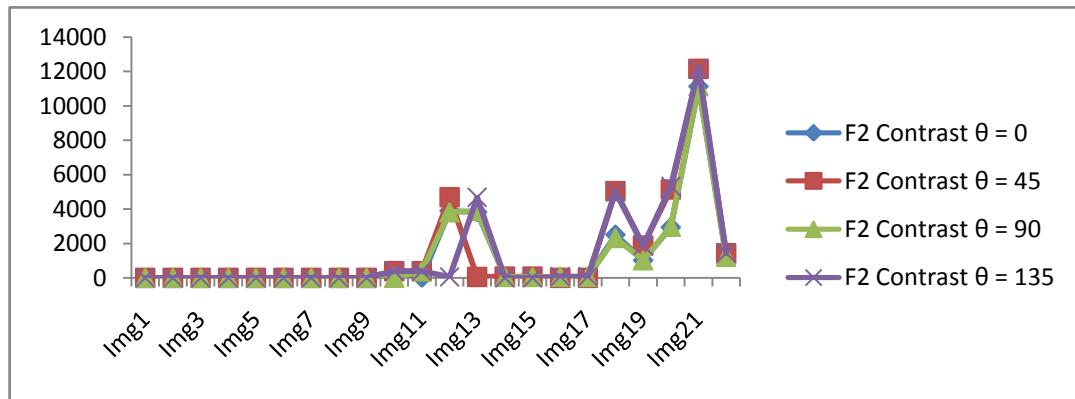


Figure 4.6: Contrast evaluation for four angles: Contrast is zero when the neighboring pixels have constant values.

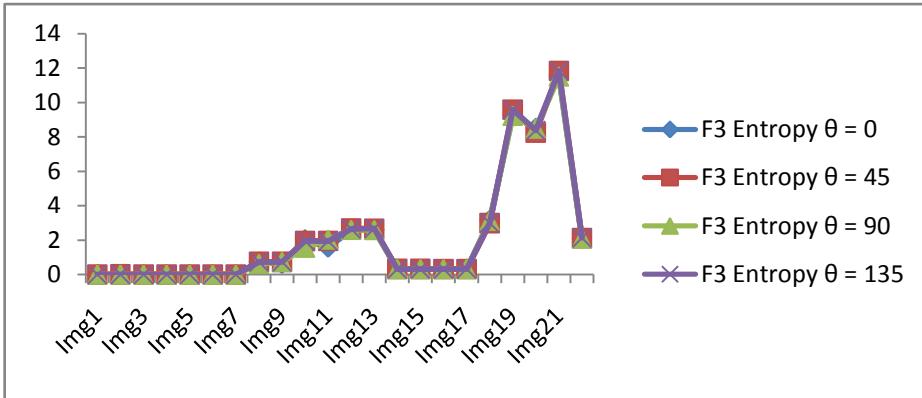


Figure 4.7: Entropy feature: more with varied pixel-cases and vice-versa.

The entropy features demonstrate that it is almost directional-invariant. Only for image#10 and image#11, the values varied slightly for horizontal and vertical directions. For image#10 which has horizontal lines has less value in vertical angular direction and for image#11 shows the opposite nature due to its vertical bars. If an image has no variations, its entropy is zero and vice versa. Similar nature is visible from Fig. 4.8 for variance feature. In this case, image#12 and image#13 shows mirrored values for horizontal and back diagonal directions, and vertical and front diagonal angular directions. For other cases, it is almost equal for four angular directions. Hence, the nature of variance is almost independent of angular variations.

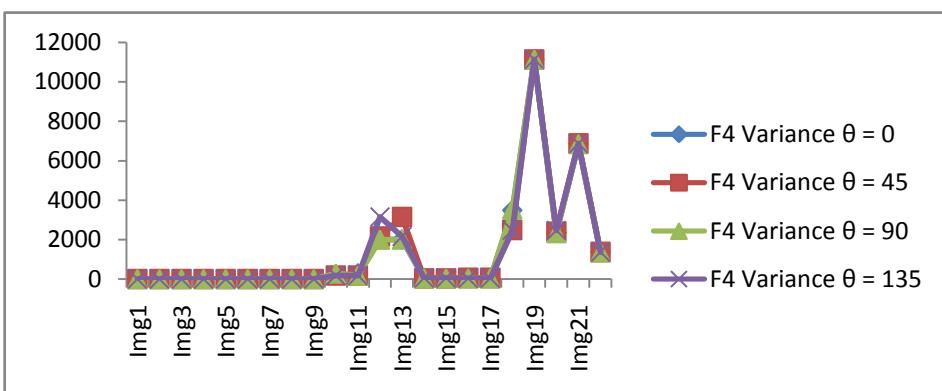


Figure 4.8: Variance for four angles: Similar distribution of entropy.

The correlation is computed for the images and it has diversely varied values for all four directions and hence, this feature presents directional-variant characteristics for image textures. In the case where a slope of 0 is present, the correlation coefficient is undefined because the variance of Y is zero. To illustrate a bit more on correlation and its positive vs. negative values, we can say that the correlation between two random variables ( $X$  and  $Y$ ) provides a measure of the degree to which  $X$  and  $Y$  tend to ‘move together’:

- $\text{corr}[X, Y] > 0$  indicates that deviations of  $X$  and  $Y$  from their respective means tend to have the same sign. Hence, the variables are positively linearly correlated (or simply positively correlated).
- When  $\text{corr}[X, Y] < 0$ , it means that deviations of  $X$  and  $Y$  from their respective means tend to have opposite signs. So these are negatively linearly correlated (or simply negatively correlated).
- When  $\text{corr}[X, Y]$ ,  $X$  and  $Y$  do not display any of these two tendencies and the two variables are uncorrelated.
- If the linear dependence between two random variables is high, the correlation is closer to 1.

Table 4.I and Figure 4.9 show various correlation values for some symmetric or almost-symmetric images. From the values (positive or negative) and the angle-dependencies of symmetric images of the same type – the concept of the correlation can be understood for image texture analysis. Values closer to 1 (or, -1) indicate an almost perfect linear relation between the two random variable – with positive (or

respectively, negative) slope. However, values close to 0 indicate any significant absence of a linear relation between them.

**Table 4.1:** Correlation values for some symmetric or mirror-type images

		$\theta = 0$	$\theta = 45$	$\theta = 90$	$\theta = 135$
	Img8	-0.03993	-0.03994	0.999511	-0.03988
	Img9	0.999511	-0.03988	-0.03993	-0.03994
	Img10	-0.10052	-0.10436	0.989081	-0.09837
	Img11	0.989081	-0.09837	-0.10052	-0.10436
	Img12	0.028453	-0.08613	0.045042	0.990858
	Img13	0.045042	0.990858	0.028453	-0.08613
	Img14	0.120478	-0.00807	0.146615	0.989201
	Img15	0.120478	-0.00807	0.146615	0.989201
	Img16	0.146615	0.989201	0.120478	-0.00807
	Img17	0.146615	0.989201	0.120478	-0.00807

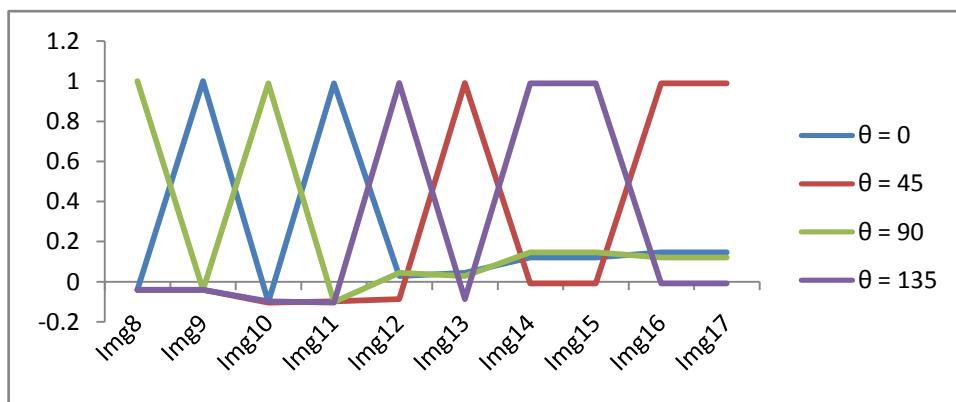


Figure4.9: The corresponding correlation values for four directions of images, as shown in Table 4.1.

On the other hand, the inverse difference moment (as shown in Fig. 4.10) is also theta-variant and has higher values if an image has less variation. Figure 4.11 and onwards have more complex computation but provide important information on image feature based on the co-occurrence matrix. The sum average feature is theta-invariant and similar to sum the variance feature; it shows slightly mirrored values for two mirrored images (image#12 and image#13, which are diagonally different - in forward and backward directions). The sum entropy features (as shown in Fig. 4.13) for the textures demonstrate almost similar nature that of the entropy features above. Though the patterns resemble each other significantly, the feature values are not alike. The mean of sum entropy for all texture images for four directions is 1.6, whereas, the same value for the entropy feature is 2.1.

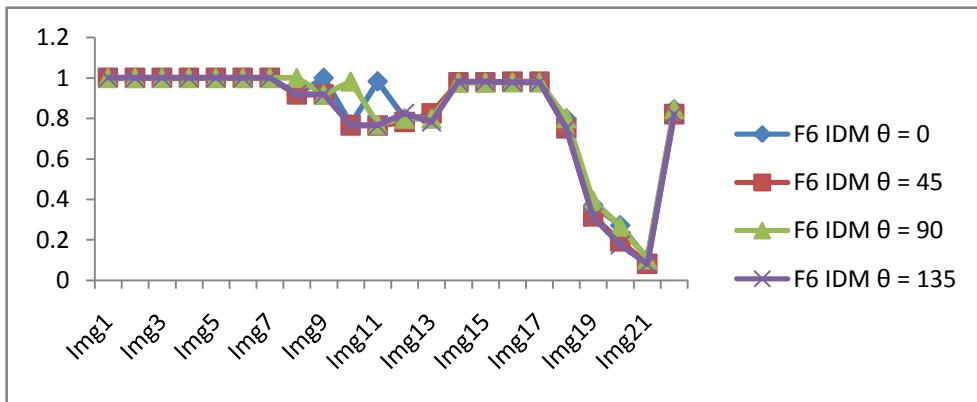


Figure 4.10: Inverse Difference Moment for four angles: the IDM has higher value when all elements of the image are same.

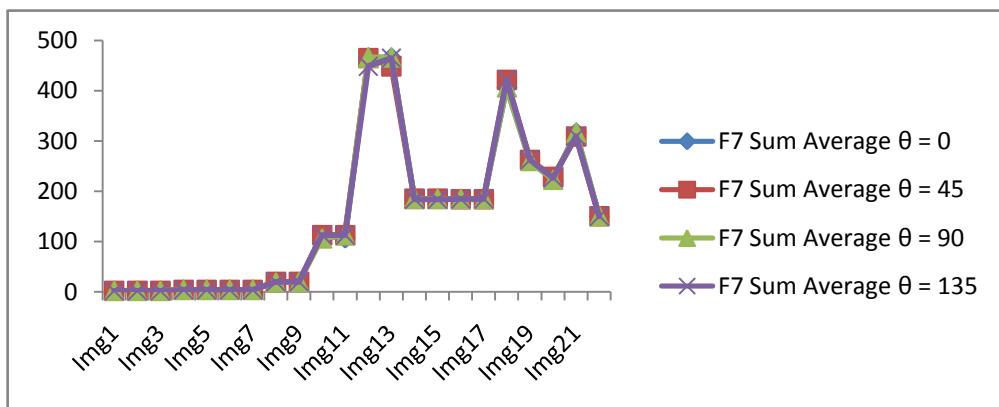


Figure 4.11: Sum average feature.

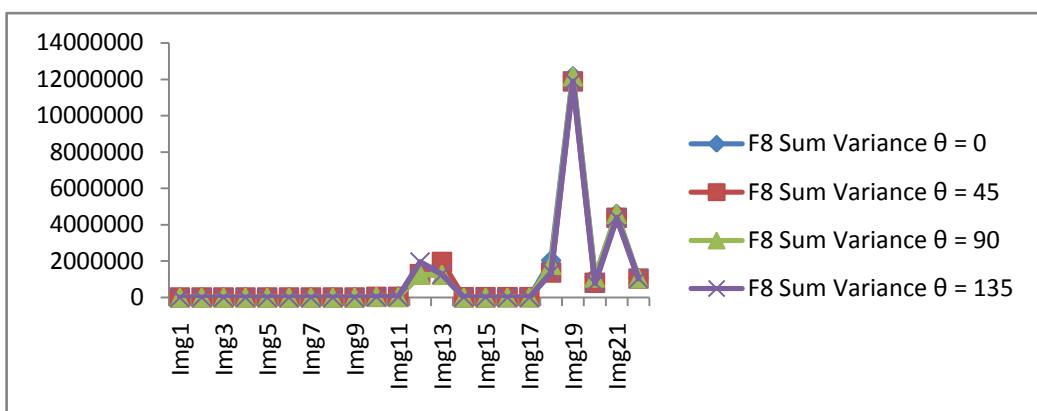


Figure 4.12: Sum variance feature for four angular orientations.

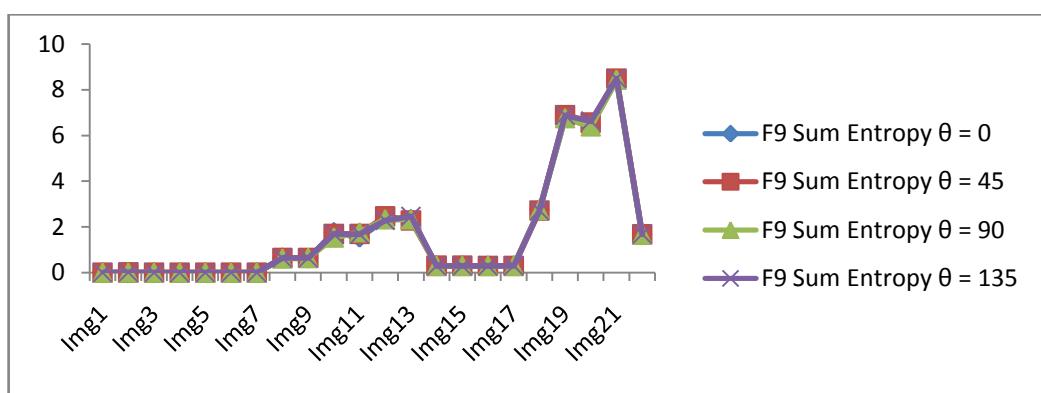


Figure 4.13: Sum entropy for four angles: Note that F7~F9 depict varied nature for images with varied pixel positions.

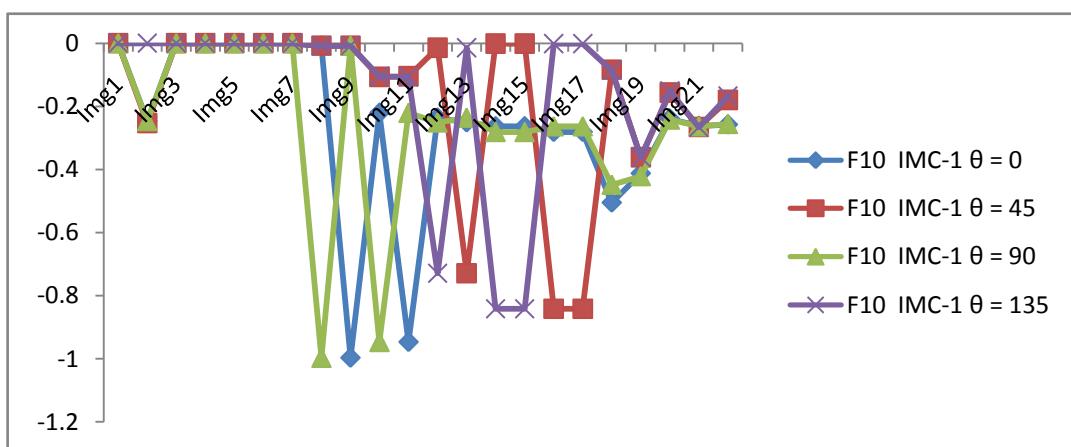


Figure 4.14: Information Measures of Correlation – 1: It produces negative values for all cases.

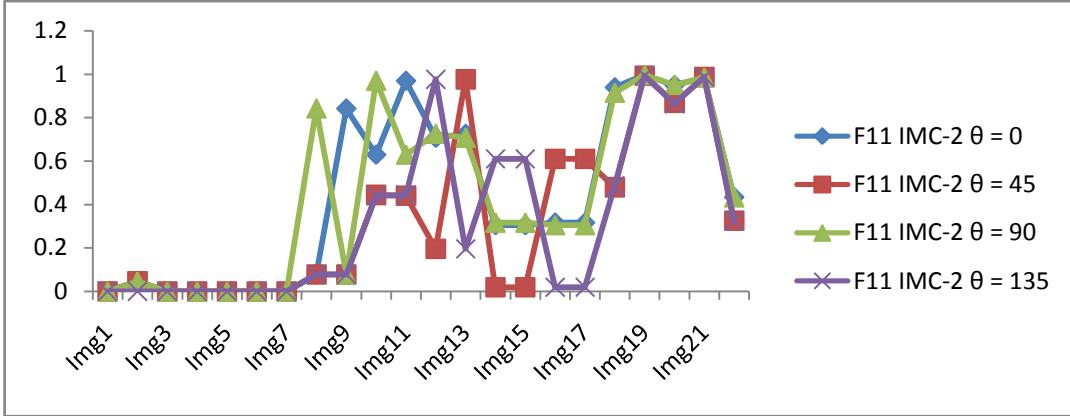


Figure 4.15: IMC-2: It is evident that the IMC-2 is not similar to IMC-1 though the variations look alike a bit but the variation ranges are different.

Table 4.2: Some values for IMC-1 feature. The corresponding images are shown here. Note that borders for each image are provided here only for better visibility. In the calculation, there were no borders for these images.

		$\theta = 0^\circ$	$\theta = 45^\circ$	$\theta = 90^\circ$	$\theta = 135^\circ$
Image#14		<b>-0.26341</b>	<b>-0.00115</b>	-0.28146	-0.84144
Image#15		<b>-0.26341</b>	<b>-0.00115</b>	-0.28146	-0.84144
Image#16		-0.28146	-0.84144	<b>-0.26341</b>	<b>-0.00115</b>
Image#17		-0.28146	-0.84144	<b>-0.26341</b>	<b>-0.00115</b>

Figures 4.14 and 4.15 present the last two features, called information measures of correlation - 1 and information measures of correlation - 2. These two features are computationally taxing but reveal different kind of information that might be monumental for feature analysis for textures. It is evident that the IMC-2 is not similar to IMC-1 though the variations look alike a bit but the variation ranges are different. The IMC-1 features are negative values (except for image#1, 3~7). The

image#14~15 show same value for all angular orientations even though these are mirror-images. On the other hand, similar but diagonally different image#16 and imag#17 compute exactly the same values but for switching the angles. Table 4.2 shows the feature values for these four images. The final feature, IMC-2 depicts variability in different directions and values are varied for different textures. The IMC-2 also shows similar nature as in table 4.2.

Hence, from these image features based on the co-occurrence matrices on gray-scale images, we can conclude that sum average and sum entropy are  $\theta$ -invariant; the ASM, entropy, variance and sum average are almost  $\theta$ -invariant (only a few cases for each, the values are slightly varied and mainly due to their mirrored-alignment of their textures); and other are  $\theta$ -variant. The contrast, correlation, inverse difference moment, IMC-1 and IMC-2 features demonstrate varied values for varied angular directions while computing the matrices. In this chapter, our analyses are based on four directions as other four directions just replicate the reverse values for these four directions. Hence, 4 orientations or directions can be good enough for any computation. However, if no conclusion can be made for a feature in different orientations or distances (radius), we can perform exhaustive computation for all possible orientations and radius values. Then evaluate to find the most appropriate combinations for orientation and radius for specific classes for analysis.

#### 4.4 Conclusions

This chapter exhaustively analyzes eleven second-order statistical features or cues based on co-occurrence matrices to understand image texture surface. These features are exploited to analyze properties of image texture. The features are also categorized based on their angular orientations and it is noticeable that we can

categorize these features into three different groups – orientation-invariant features, almost invariant to orientations, and orientation-variant features. In this chapter, we consider four angular variations or directions or orientations. The distance is kept as 1 for every case. Based on our experimental evaluations, four orientations seem to be sufficient for any computation. It is also understood that when a conclusion cannot be made based on the above settings, we can perform exhaustive computation for all possible orientations and radius values. Afterwards, we need to find the most appropriate combinations on orientation and radius for specific classes for analysis.

Based on a standard dataset, which has twenty-one texture images, ranging from no complexity to varied complexity, we vividly unearth the corresponding nature and variations in feature-characteristics for each feature. This dataset is a newly-developed dataset that can be exploited for characterizing various features on gray-scale variations. This in-depth analysis is significant for image surface analysis and it allows a researcher to select the most-suitable features on a specific dataset to evaluate these and understand their distinctive characteristics. In the next chapter, we do more in-depth analysis of the above statistical features on another large dataset and distinctively categorize these features into several groups.

# Categorization of Features Based on Co-occurrence Image Matrix

## Chapter 5

---

### 5.1 Introduction

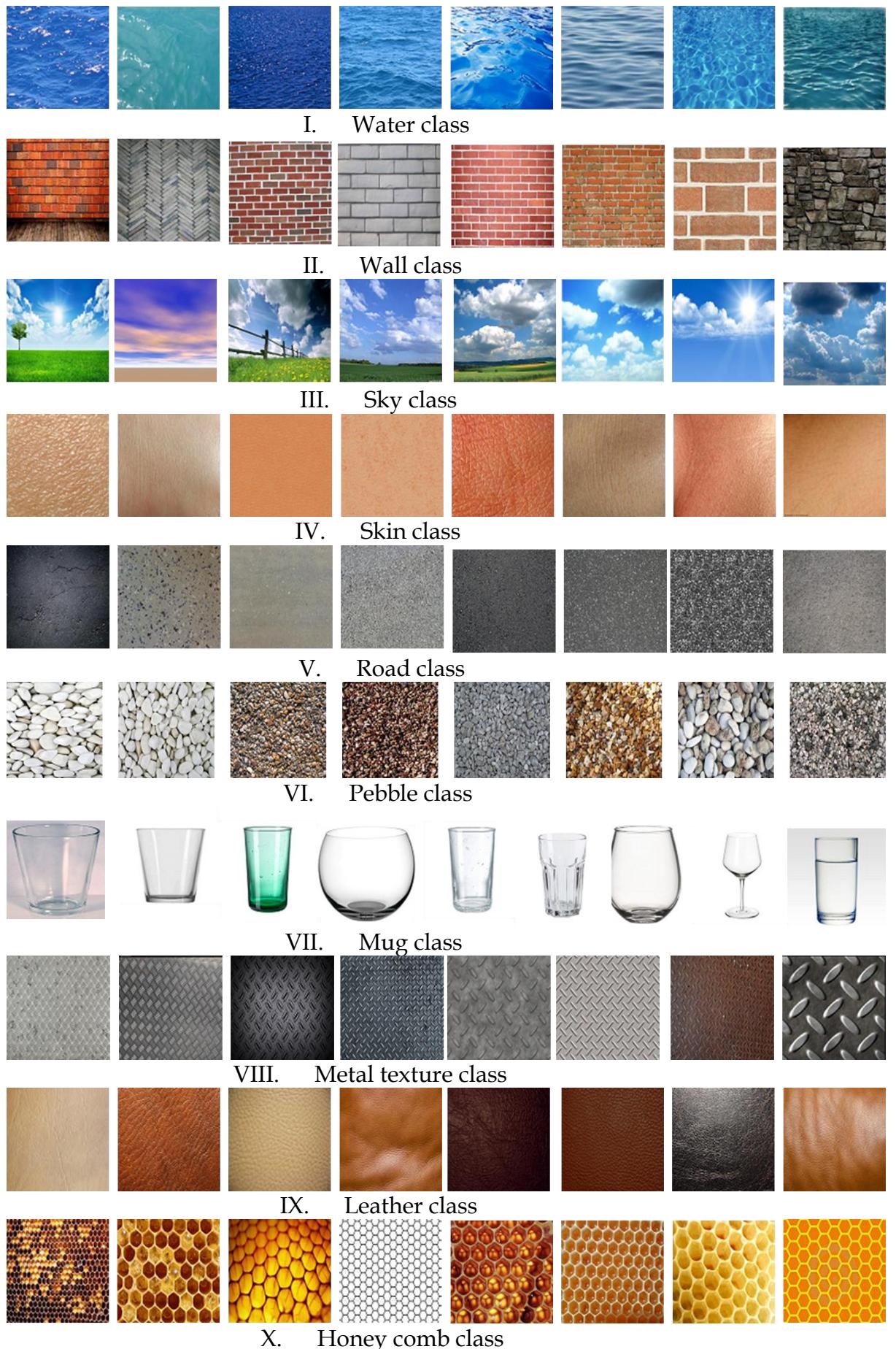
In this chapter, we rigorously investigate the gray-level co-occurrence features. We analyze these features with verity for image classes. We attempt to explain these features and want to demonstrate that which feature is good for which types of classes. Is really these features can categorize based on different classes? For this target, we propose feature groups that are crucial for texture analysis based on various criterions.

### 5.2 Development of Multiclass Database

In this chapter, we develop a new database called multiclass database. It contains seventeen classes of image data and each class contains lots of images. All are real-world images. Our database has the following image classes (N.B.: in the braces, the number of images per class is mentioned), as shown in Fig. 5.1:

ball (18), cow (14), fingerprint (18), forest (17), glass (18),  
honeycomb (14), leather texture (22), metal texture (20),  
mug (20), one-color (23), pebble (8), road (18), skin (12),  
sky (17), tree (12), wall (20), and water (20).

Figure 5.1 shows eight sample images from each class. From this Fig., it is understandable that it is a diverse dataset. We use this dataset in this paper to evaluate the eleven features.



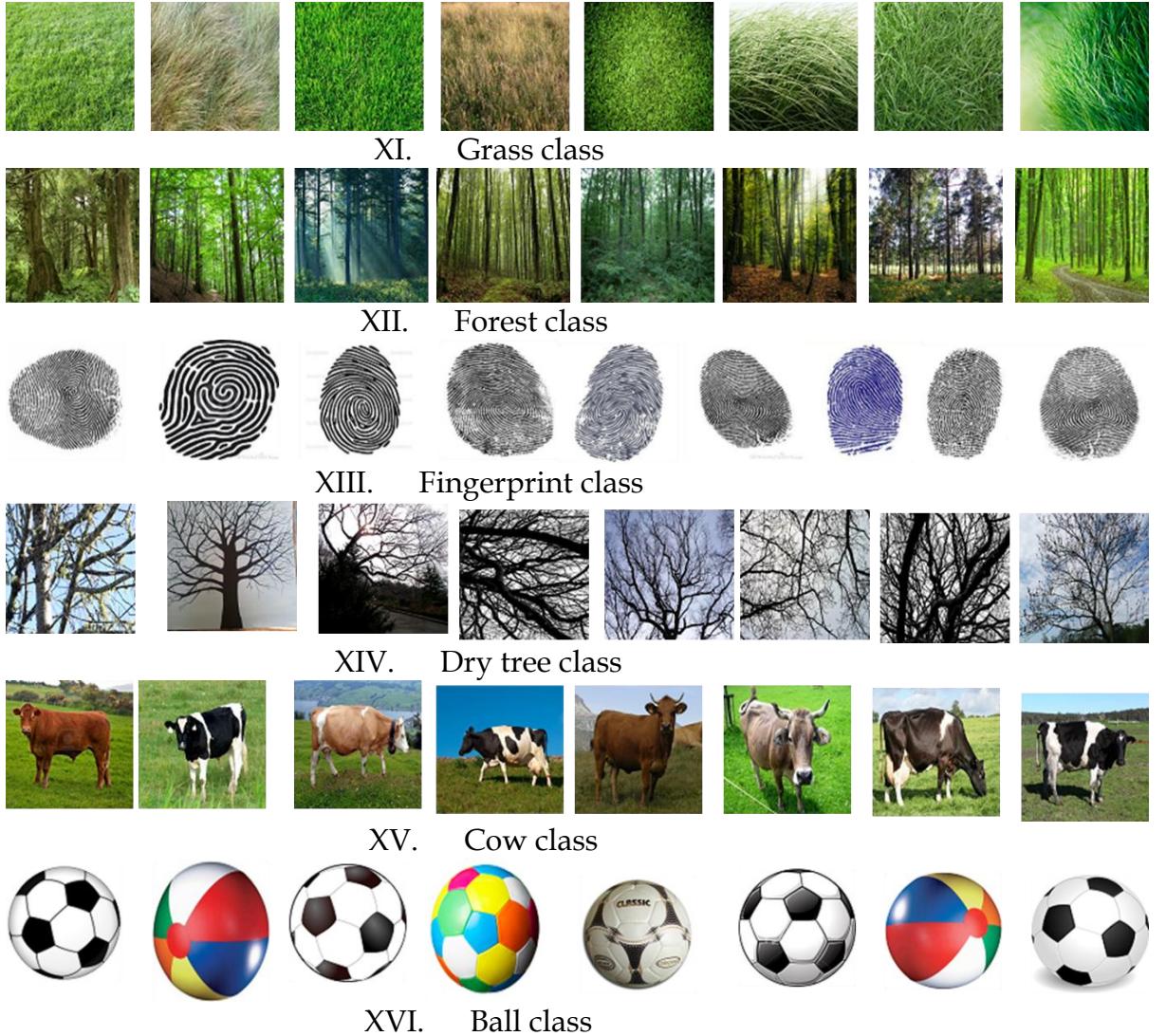


Figure 5.1: Sample images from our multiclass database.

### 5.3 Comparisons among Features

In this section, we extensively analyze our multiclass database on various features and these features are compared into different sub-categories to understand their inherent meanings and characteristics. The following sub-sections illustrate on the eleven features and finally, we propose two different categories of these features.

#### 5.3.1 ASM vs. IDM features

The angular second moment (ASM) shows the uniformity or energy of the image. From the graph of Fig. 5.2(b), it is evident that except for soccer, mug, fingerprint and sky classes, others have presence of much non-uniformity. For soccer, mug and fingerprint, a significant percentage of the images have uniform or white

`background.sky` class has images having significant of uniform blue-parts and hence, the values for ASM is a bit more than others (e.g., grass, pebble or metal-texture). From the analysis on ‘standard dataset’ as presented in Chapter 4, we can vividly claim that unless there is a necessity to differentiate between less-uniform textures versus much-uniform textures, the ASM feature is not necessary to compute for normal texture analysis, especially for analyzing natural images. In the case of one-color image where there is no variation in terms of pixel values, the ASM value is zero. So, this feature can be exploited based on dataset and applications.

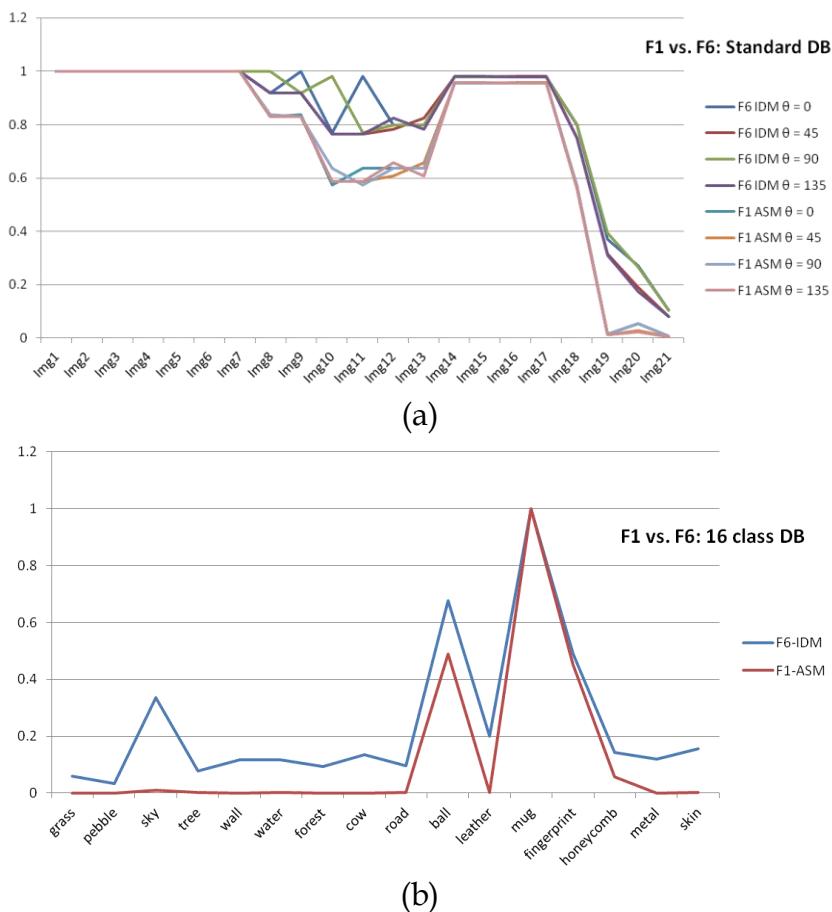


Figure 5.2: Angular second moment VS. Invers different moment features

For the case of Inverse Difference Moment (IDM), which can be called homogeneity (F6), we notice that it is angle-variant. Similar to F5, for `forest` class,  $0^\circ$  shows more values than others. A tree's trunks are black or brownish and like a big column in forest and green leaves and grounds are different than the vertical pillar-like trees. For `one-color` class, the homogeneity is 1 for all images, as each image of

this class is purely homogeneous. Except a few cases, like mug class, soccer class, sky class and fingerprint class – othervalues are usually closer to zero and it depicts that F6 shows relatively similar nature of Angular Second Moment (F1).

Figure 5.2 here show two graphs for comparative analysis for F1 (angular second moment) and F6 (inverse different moment). These graphs visibly exhibit that the characteristics for both features are almost similar. In fact, both of the features demonstrate similar characteristics, which is uniformity or energy of the image.

For *standard dataset*, the variations of the values have similar pattern. Later images (i.e., image#18 and onwards) have less F1 and F6 values, because these images have lots of variations (i.e., chessboard, etc.). These variations for our *multiclassdatabase* are similar too. The four picks for F6 feature are for mug, ball, fingerprint and sky classes, and we find that these classes have more uniformity in terms of the presence of pure white values. Hence, the F1 or F6 values are higher for these classes. Other classes have less F1 or F6 values and they altogether demonstrate that the ASM feature or the IDM feature show similar characteristics and hence, less importance for texture analysis. However, the IDM has little better variations than the ASM.

### 5.3.2 Contrast vs. entropy features

The contrast feature (F2) provides intensity or gray-level variations in an image. We notice variations in contrast values in the graph of Fig. 5.3. From our analysis, we notice for pebble class that if the pebbles or stones are very small compare to others, we get more variance for smaller pebble-based images. Also based on the alignment, the values vary for different angles. For example, pebble shows more values in  $135^\circ$ , and less in normal condition; whereas, tree provides more contrast for  $45^\circ$  and less in  $90^\circ$ .

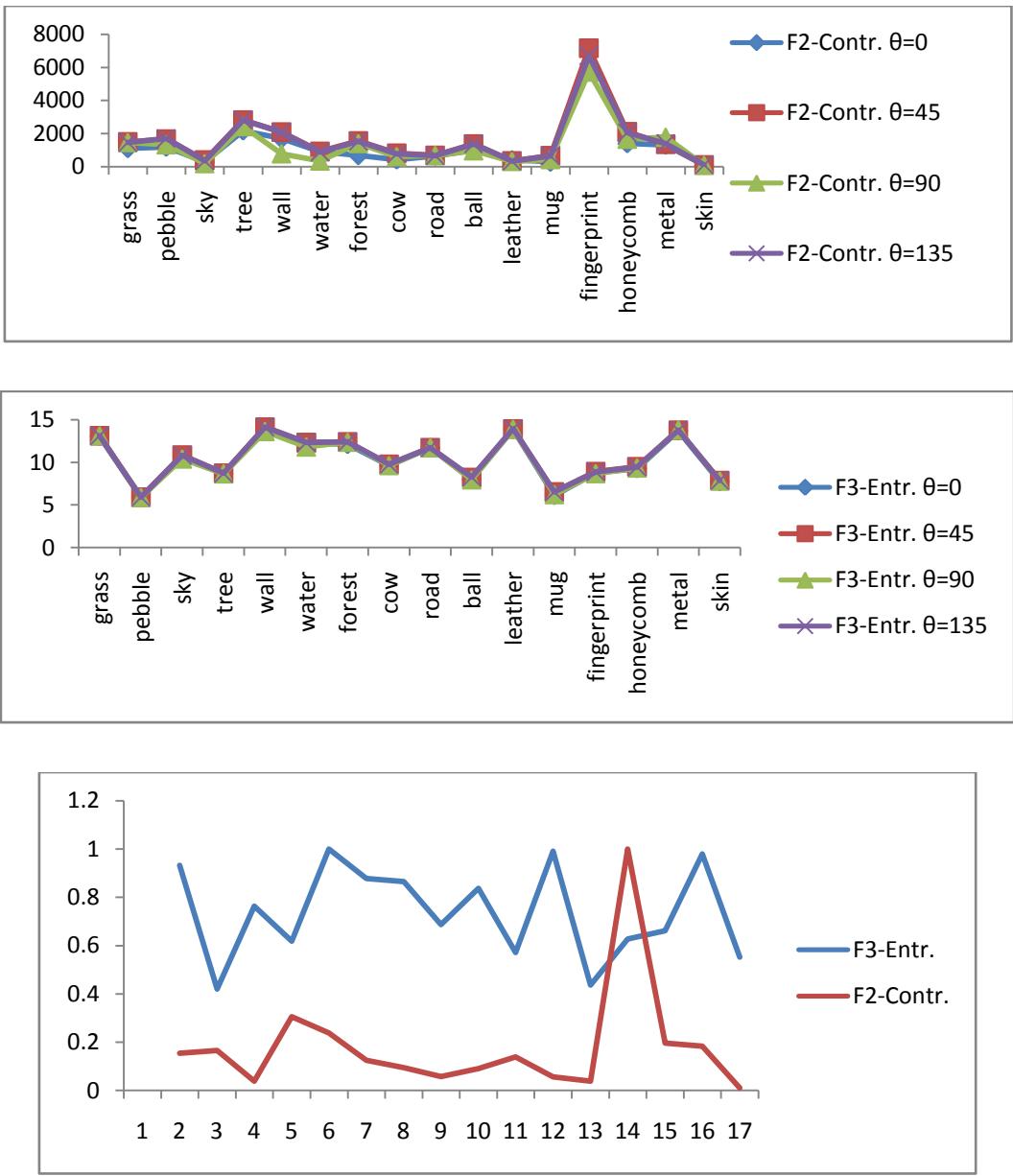


Figure 5.3: Graphs to compare the characteristics of contrast and entropy features.

On the other hand, the entropy (F3) graph demonstrates that if the image has more irregularities or more contrast (F2), the entropy will be higher. Hence, except for one-color class, all classes have varied entropy value. The one-color class has zero entropy for each image, due to the fact that all images have just one color with uniform distribution, therefore, zero entropy. Figure 5.3 shows the relative relationships between these two features. As per our study, it is evident that both of them are important for analysis. The physical significances of these features are

demonstrated in the earlier chapter. And both of them vary in terms of their nature as well. So it is not possible to consider one by leaving another. Though entropy is directional-invariant, the contrast is opposite. The bottom graph of Fig.5.3 clearly shows their differences. Note that it is shown based on their individual normalized values, else both characteristics cannot be shown in one graph due to the range of values (for contrast, the range is in few thousand, whereas, the entropy range varies within 6 to 14).

### 5.3.3 Contrast vs. variance features

From the Fig.5.4, it is visible that the natures of contrast and variance features have similarities to some extend but the range varies. Also, variance is theta-invariant – opposite to the variance. Based on our analysis, we claim that both of these features are required for texture analysis.

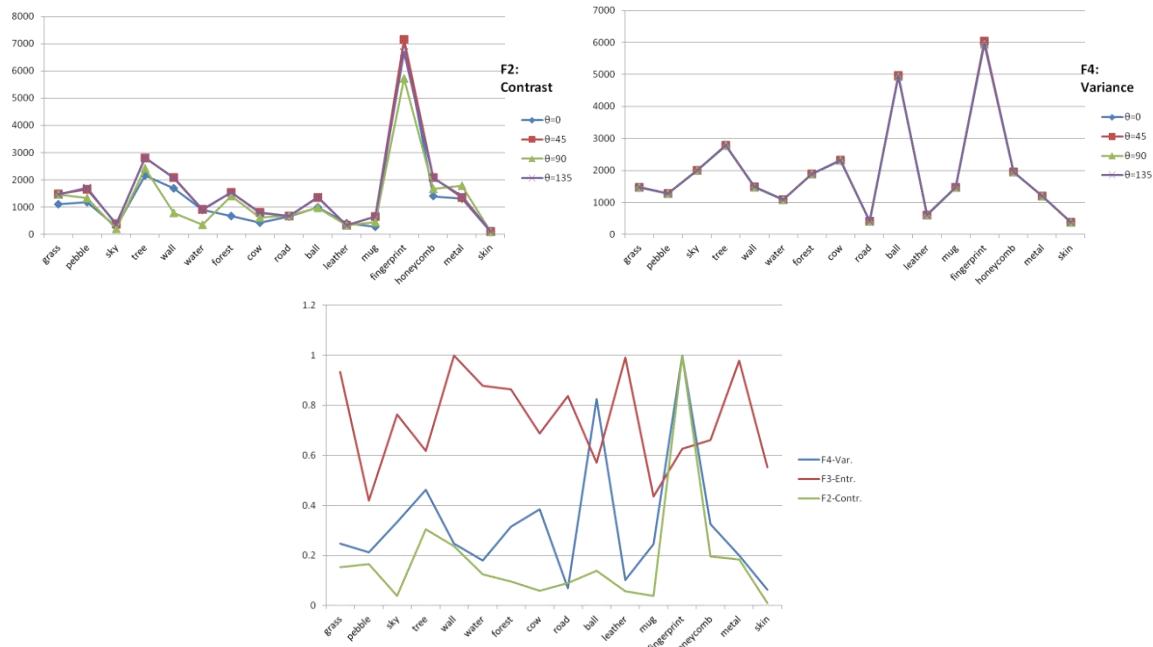


Figure 5.4: Contrast vs. Variance features

### 5.3.4 Entropy vs. sum entropy features

For all classes, 'entropy' and 'sum entropy' features demonstrate strong discrimination abilities. Therefore, we can consider both features. However, both of these features exhibit almost similar patterns for all classes. Figure 5.5 becomes an evidence of that fact that both are similar and there are not much differences.

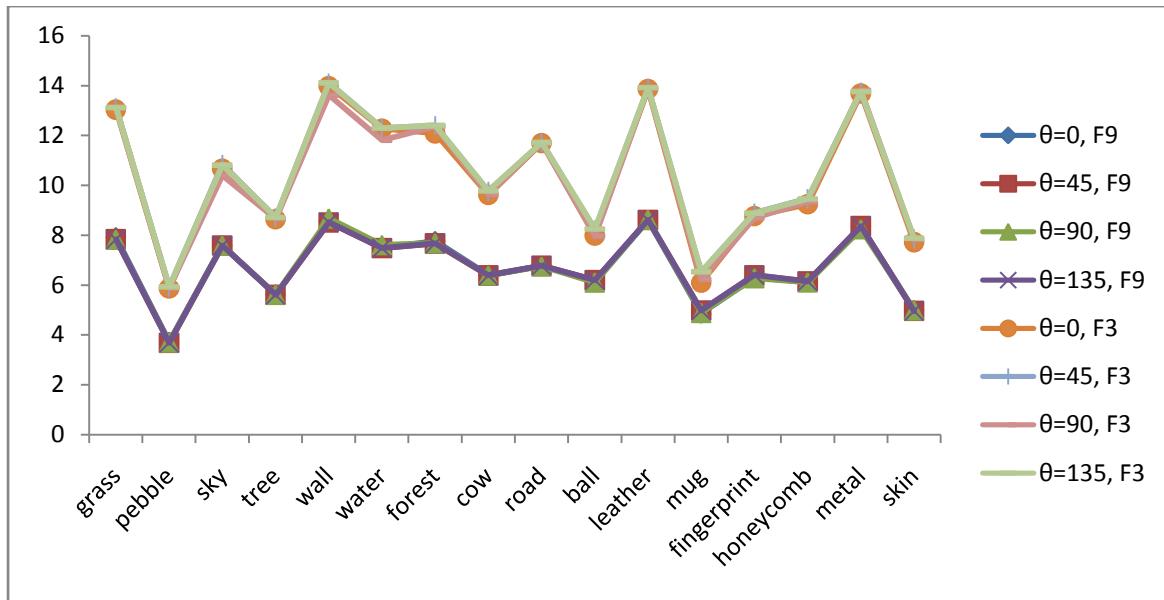


Figure 5.5: Entropyvs. Sum entropy features

Moreover, these both features demonstrate almost similar nature in terms of angular variations as well. Actually, from our analysis, we claim that ‘entropy’ has a very slight angular variation, whereas, ‘sum entropy’ is purely angular-invariant. Hence, we consider to take one of these features. Another point is the computational cost, and ‘entropy’ has less computational cost.

### 5.3.5 Correlation feature

The F5 (correlation) will be high if an image contains a considerable amount of linear structure. Correlation relates to *dependence*. The correlation or F5 is also important and shows varied values for different angles for different classes. Hence, it is angle-variant.

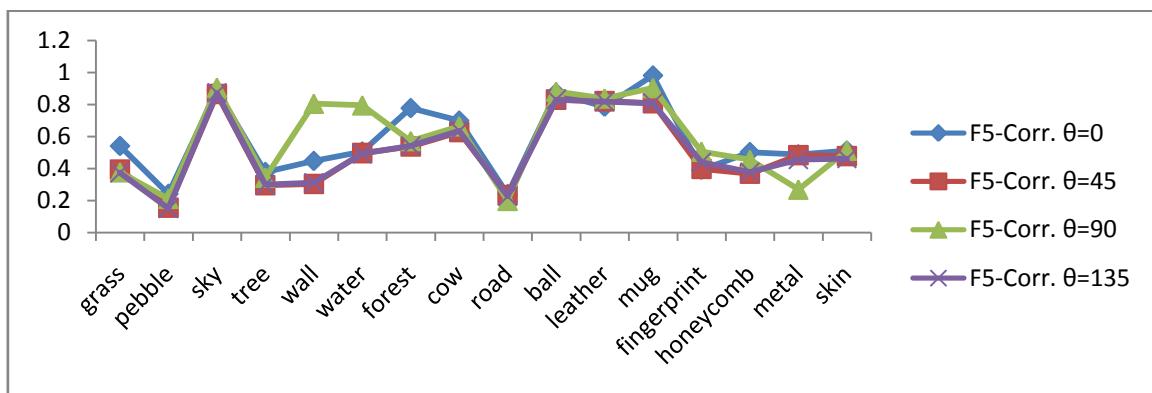


Figure 5.6: Correlation variations for multi-class dataset.

For pebble class, if an image has horizontal arrangement of the pebbles or stones, then it shows more values in 90 direction, on the other hand, if the alignment is vertical, then 0 angle direction shows more correlation among pixels. The late point is more visible in the case of forest class where almost every image has vertical trees in the scenes, hence, forest class demonstrates almost twice values for 0 angle than other angles. From the discussions on correlation in Chapter 4, it is evident that it is a very critical and crucial cue for texture analysis. Therefore, we need this feature and it has distinct patterns than others. Figure 5.6 shows the variations of correlation values for the multiclass dataset.

### 5.3.6 Variance vs. sum variance features

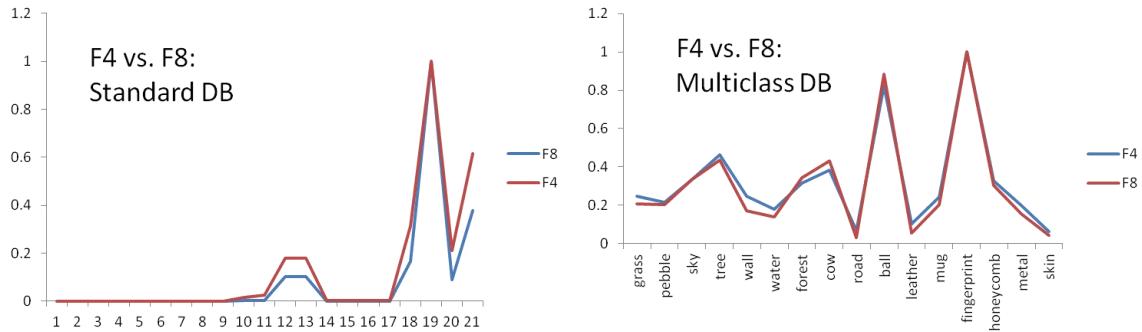


Figure 5.7: variance vs. sum variance comparison.

In this sub-section, we evaluate the pair of variance feature (F4) and the sum variance feature (F8). In the Figure 5.7, we demonstrate two graphs – one for *standard database* and another for *multiclass database*. For F4 and F8 features, it is clearly visible that they have exactly the same nature. Even though the actual values or the ranges vary significantly but the inherent characteristics are same. Therefore, we do not need to compute both features at the same time. As the computation of F4 is easier to the same for F8, we propose not to consider the sum variance (F8) for texture analysis.

### 5.3.7 Sum average vs. IMC1 vs. IMC2 features

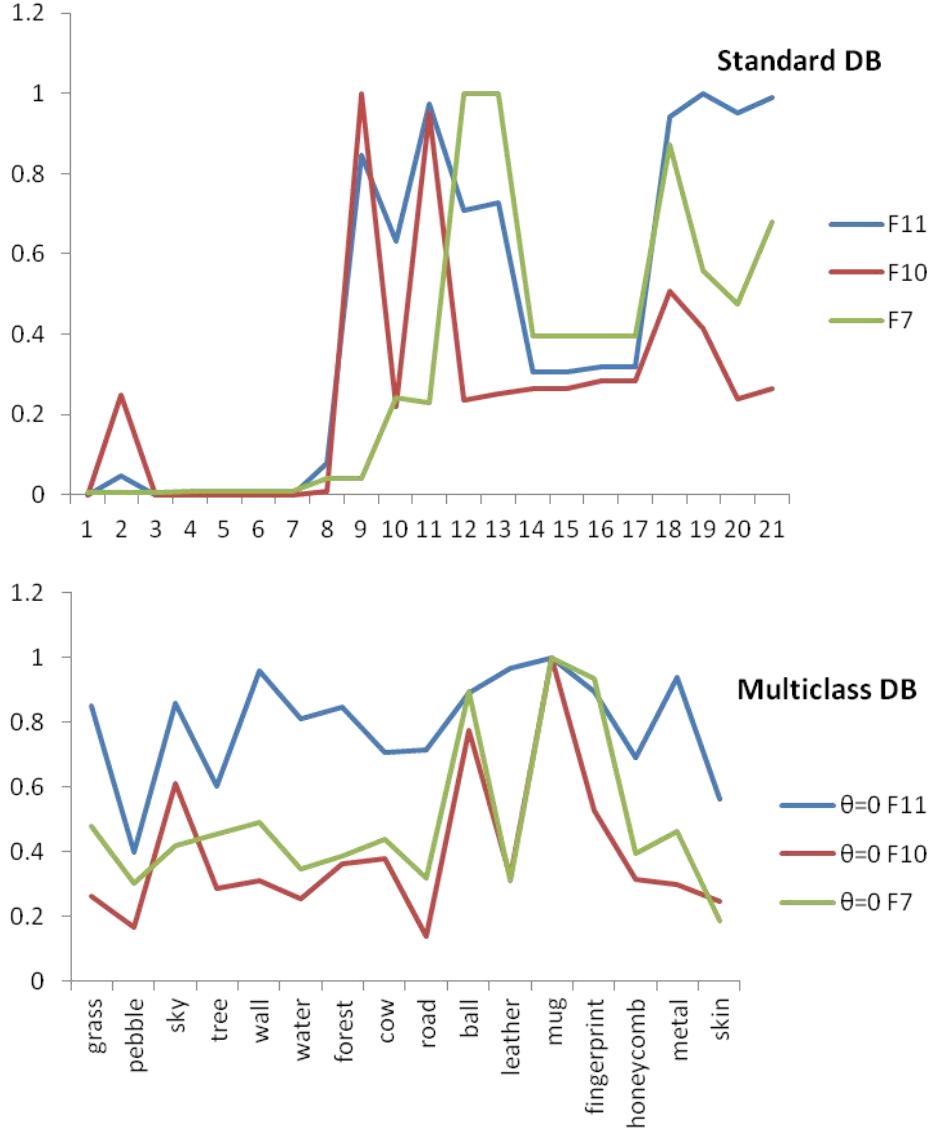


Figure 5.8: Sum average vs. IMC1 vs. IMC2 comparison.

From the analysis of sum average (F7), IMC1 (F10) and IMC2 (F11) features, we can notice some similarities as well as few varied characteristics among themselves. From the Figure 5.8, it is visible that for both datasets, the average normalized values demonstrate similar patterns for F7 and F10 more than F11 features. Based on our analysis, we conclude that these three features can be computed for texture analysis though these can be ignored too. The physical significance cannot be realized except that these features are based on a bit more complex computations and based on entropy computation.

Hence, from rigorous empirical analysis, we propose the following sub-categories of these co-occurrence-based image features, as shown in Table 5.1.

**Table 5.1:**Proposed categories of features based on importance.

Category	Feature	Comment
Required Features	Contrast	- Have discriminating ability. - Rotationally-variant.
	Entropy	- Have strong discriminating ability. - Almost rotational-invariant
	Variance	- Have discriminating ability. - Rotational-invariant.
	Correlation	- Have strong discriminating ability. - Rotational-dependent feature.
Less Important Features	Sum average	- Characteristics are related to 'entropy' - Rotational-invariant
	Information Measure of Correlation-1	- It has almost similar pattern of 'sum average' but vary for various classes - Varies significantly with rotation
	Information Measure of Correlation-2	- It is computationally expensive compare to others. - Rotation-variant
Not required Features	Angular Second Moment / Energy	- No distinguishing ability
	Inverse Different Moment	- Similar to 'angular second moment'
	Sum Variance	- Similar to 'variance'
	Sum Entropy	- Similar to 'entropy'

## 5.4 Group-wise Analysis based on Key Features

In this sub-section, we analyze between relatively closer classes. Here, we categorize these classes on the *multiclassdatabase* into the following different groups:

- Group 1 : grass vs. sky vs. cow
- Group 2: skin vs. leather vs. water
- Group 3 : forest vs. tree

- Group 4 : wall vs. road
- Group 5: Natural classes
- Group 6: Material classes
- Group 7: Texture-based classes

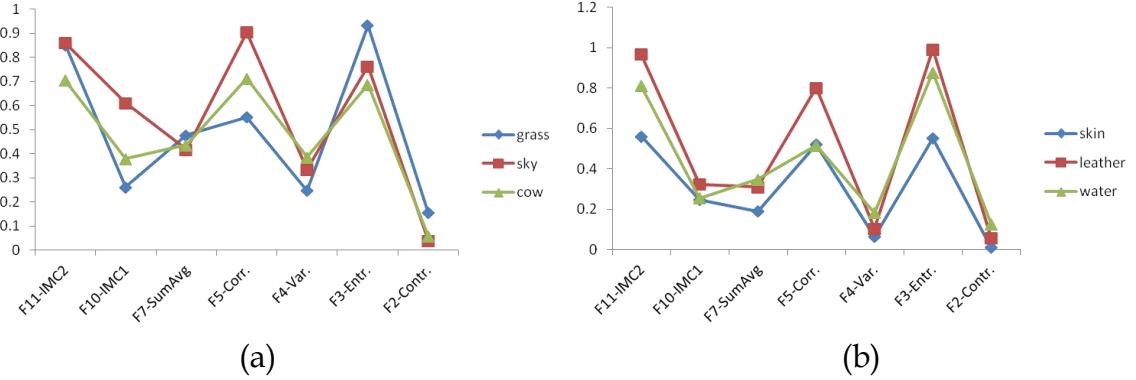


Figure 5.9: Seven key features for different groups of classes: (a) grass, sky and cow class; (b) skin, leather and water class.

In Fig. 5.9(a), we make group with grass, sky and cow. We make them group because in cow class, it contains sky and grass. So, they have similar property with each other. We easily can say that variance feature does not show significant difference. Correlation and IMC1 features can distinguish each other. So we can say that Correlation, IMC1 is good for this type of image.

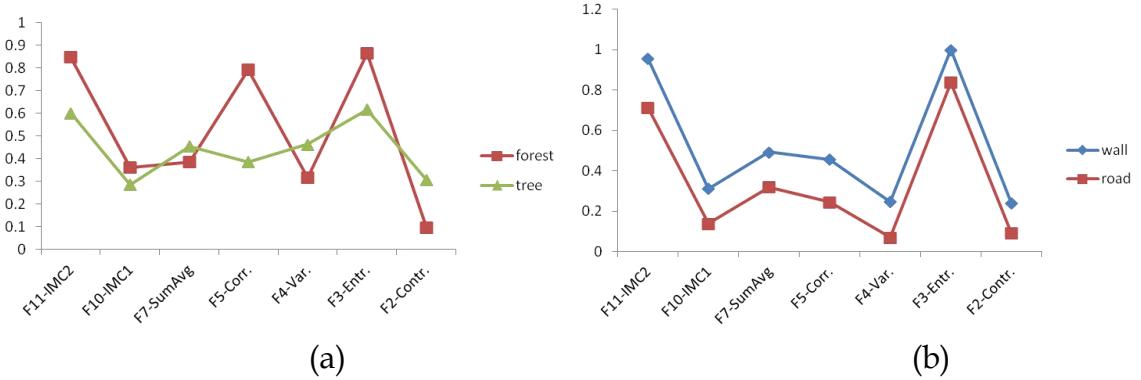


Figure 5.10: Seven key features for different groups of classes: (a) forest vs. tree class; (b) wall vs. road class.

skin, leather and water make another group. They have similarity as a surface property. For example, their surfaces are relatively plain; however, water has little bit wave texture or ripple. From Fig. 5.9(b), we can say, skin, leather and water can separated from these features.

forest and tree make another group. We make them in group because forest and tree have some similar properties. After analysis, we find that the co-occurrence features can easily distinguish between forest and tree. In Fig. 5.10(a), IMC1 and sum average features show little bit similarity between them. In Fig. 5.10(b), we make group between wall and road. wall and road have almost similar property. The co-occurrence features easily can differentiate them.

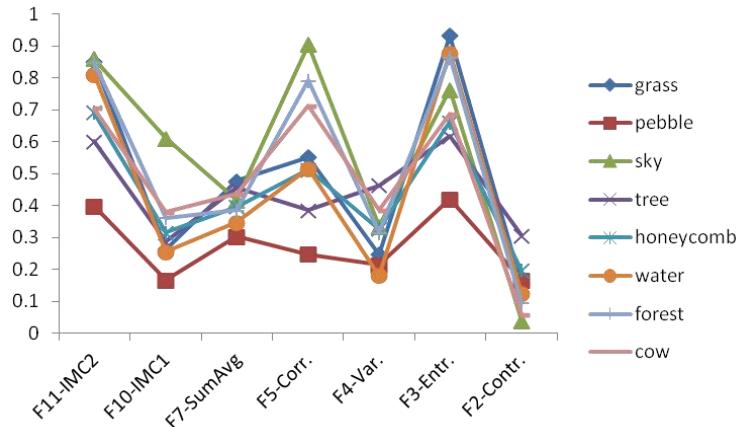


Figure 5.11: Seven key features for some natural image classes.

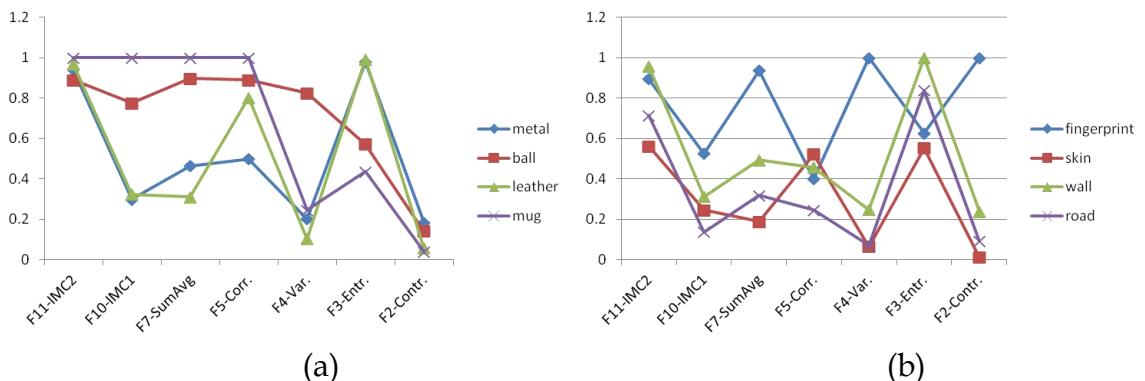


Figure 5.12: Seven key features for different groups of classes: (a) some material images; (b) some texture-based classes.

In Figs. 5.11 and 5.12, we make three groups between all natural images, material images and texture images from our database. And from these graphs in Figs. 5.11 and 5.12, we can say that co-occurrence features can distinguish each other very clearly.

## 5.5 Category of Features based on Angular Directions

After some rigorous experimental analysis, we propose the following category of the features based on co-occurrence image matrix. These are clustered into three different groups, as follows:

- i. Rotation-invariant features
- ii. Almost rotation-invariant features
- iii. Rotation-variant features

These are explained below with appropriate analytical graphs.

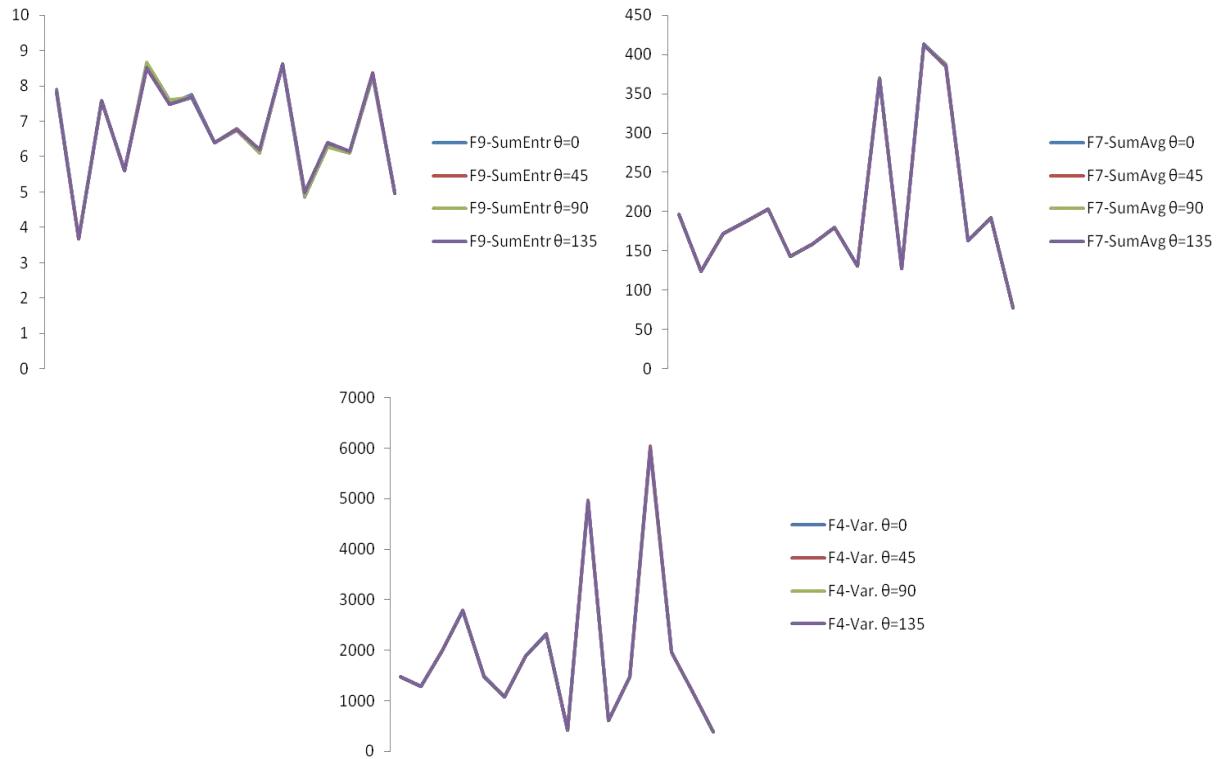


Figure 5.13: Graphs to demonstrate the rotation-invariant features.

### 5.5.1 Rotation-invariant features

They show theta-invariant with respect of different classes. Three features are found to be rotational-invariant. Figure 5.13 shows this invariance characteristic for the following three features,

- Variance
- Sum average
- Sum entropy

### 5.5.2 Almost rotation-invariant features

In this category, we put three features. However, these features are very close to the above category where only a few cases, we notice some variations or deviations from being invariant. Hence, we categorize these into *almost* invariant to angular rotation. Figure 5.14 shows these characteristics. These features are,

- Angular second moment
- Entropy
- Sum variance

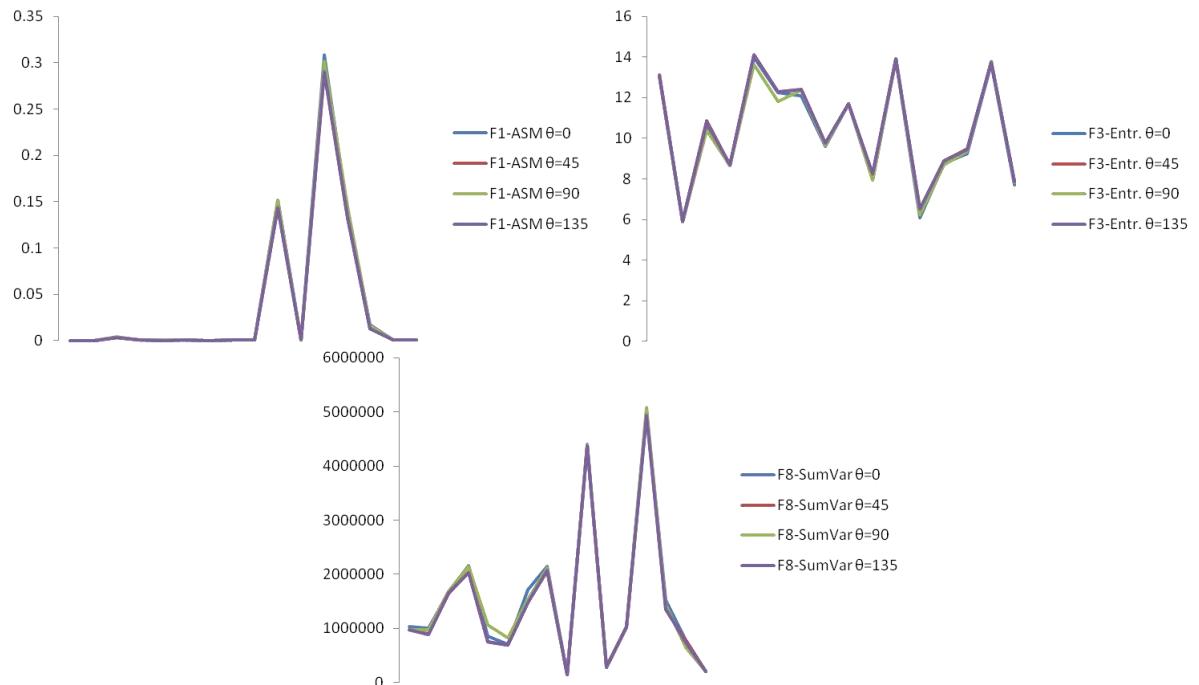


Figure 5.14: Graphs to demonstrate the *almost* rotation-invariant features.

### 5.5.3 Rotation-dependent features

The other five features vary significantly with different directions ( $\theta^0, \theta^{45}, \theta^{90}, \theta^{135}$ ). In some cases,  $\theta=0$  and  $\theta=90$  demonstrate mirror-like values and the same for  $\theta=45$  and  $\theta=135$ . Hence, it is necessary to keep these information while analyzing various image textures so that only necessary features can be considered for analysis.

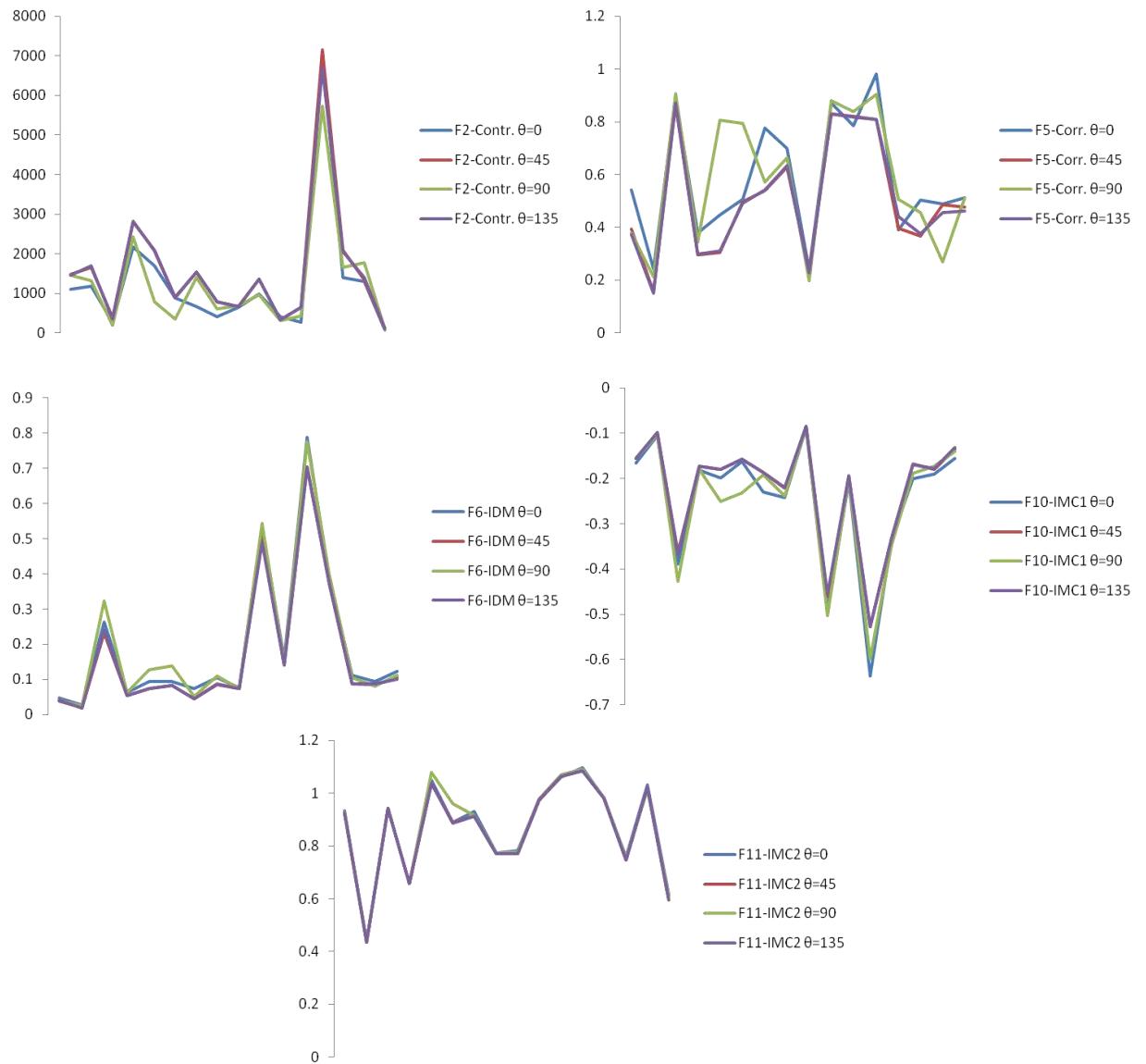


Figure 5.15: Graphs to demonstrate the rotation-variant features.

From the Figure 5.15, it is understandable that some features vary with directions. In this category, we put five features, namely,

- Contrast
- Correlation
- Inverse different moment
- Information measure of correlation - 1
- Information measure of correlation - 2

## 5.6 Conclusions

In this chapter, we thoroughly investigate the gray-level co-occurrence features. We analyze these features with variety for image classes. We attempt to explain these features and want to demonstrate that which feature is good for which types of classes. For this target, we propose feature groups that are crucial for texture analysis based on various criterions. We develop and use a large real-world *multiclass database* and the *standarddatabase*(that we introduce in Chapter 4). We study eleven features to find their inherent characteristics by using these datasets. Hence, after rigorous analysis, we propose three categories of features in terms of co-occurrence-based image matrix considering direction. These are clustered into three different groups such as rotation-invariant features, almost rotation-invariant features, and rotation-variant features.

We also categorize these features in terms of their applicability on image rotation. The rotation-invariant features show theta-invariant with respect of different classes. Three features are found to be rotational-invariant:variance, sum average, sum entropy. Next, we categorize almost rotation-invariant features. These features are very close to the above category (rotation-invariant features), where except a few cases, we do not notice any variations or deviations on angular changes. Hence, we categorize these into *almost* invariant to angular rotation. These features are: angular second moment, entropy, sum variance. The third category in this regards is the rotation-dependent features, which vary significantly with different directions. In some cases,  $\theta=0$  and  $\theta=90$  pair, and  $\theta=45$  and  $\theta=135$  pair demonstrate

mirror-like values. These rotation-dependent features are: contrast, correlation, inverse different moment, information measure of correlation - 1, and information measure of correlation - 2. It is necessary to keep this information while analyzing various image textures so that only necessary features can be considered for analysis.

From rigorous empirical analysis, we also propose the following sub-categories of these co-occurrence-based image features: required features, less important features, and not required features. These are pointed in Table 5.1. The required features (such as contrast, entropy, variance, and correlation) have strong discriminating ability among closer images. On the other hand, the not required features such as angular second moment and inverse different moment have no distinguishing ability. From our analysis, we conclude that the characteristic of sum variance is similar to that of variance; and sum entropy is similar to entropy. We categorize another sub-group called the less important features. These are sum average, information measure of correlation - 1 and information measure of correlation - 2. Sum average feature shows same characteristics related to entropy and information measure of correlation - 1 shows similar characteristics of sum average. Information measure of correlation - 2 is computationally expensive compare to other features.

Finally, in this chapter, we deeply analyze eleven co-occurrence based features. This type of in-depth analysis and categorization has not been done by others. Usually, researchers are using some of these features, mostly with random selection, for texture analysis. However, the selections of these features have not been studied deeply in the past. In the next chapter, we propose and study another type of statistical-based image representations for texture analysis.

# Texture Recognition Based on Diagonal-Crisscross Local Binary Pattern

## Chapter 6

---

### 6.1 Introduction

It is required to develop and formulate robust description of intensity values in the neighborhood pixels of an image. Texture can easily characterize surface of an object or image. Different methodologies have been proposed for analyzing textures. A well-known and widely-used method called local binary pattern (LBP) is proposed [198]. The proposed method or operator is a 2-tier version of the original method in Ref. [199]. The LBP operator is an operator that can convert or transform an image into an array or image of integer labels. The detailed computation of the LBP is presented below. Though the LBP operators and its variants are mostly developed and employed in the case of face analysis and recognition, it is also used for image texture analysis other than face-texture [17, 42-45].

In this chapter, we propose a new method called *diagonal-crisscross local binary pattern* (DCLBP) for texture representations in spatial domain, in a smarter way compare to the original local binary pattern. The proposed method is tested on two very difficult benchmark databases. We also propose two other variants of the LBP considering rotational feature and mean/median of the neighbor pixels. In this paper, we compare our proposed methods with other methods called median-LBP, Median-rotational-LBP, Interpolation-LBP, number-LBP, neighborhood-intensity-LBP. Details of these methods are available in literature and in the book on LBP [200].

## 6.2 Computation of Local Binary Pattern

In the local binary pattern (LBP) operator, a binary code is produced by thresholding a pixel's neighborhood. If we have an image of  $3 \times 3$ , then we take the center pixel and compare it with its neighbor (in this case, 8 neighbors). For an entire image, the basic operator starts can be a  $3 \times 3$  cell or block of that image.

Example	Threshold	Weights																											
<table border="1"> <tr> <td>6</td><td>5</td><td>2</td></tr> <tr> <td>7</td><td>6</td><td>1</td></tr> <tr> <td>9</td><td>8</td><td>7</td></tr> </table>	6	5	2	7	6	1	9	8	7	<table border="1"> <tr> <td>1</td><td>0</td><td>1</td></tr> <tr> <td>1</td><td></td><td>0</td></tr> <tr> <td>1</td><td>1</td><td>1</td></tr> </table>	1	0	1	1		0	1	1	1	<table border="1"> <tr> <td>1</td><td>2</td><td>4</td></tr> <tr> <td>128</td><td></td><td>8</td></tr> <tr> <td>64</td><td>32</td><td>16</td></tr> </table>	1	2	4	128		8	64	32	16
6	5	2																											
7	6	1																											
9	8	7																											
1	0	1																											
1		0																											
1	1	1																											
1	2	4																											
128		8																											
64	32	16																											

Figure 6.1: Calculating the basic local binary pattern for a  $3 \times 3$  block of an image.

Figure 6.1 shows an example of computing this local binary pattern. The basic steps of the LBP are as follows,

- Consider an image block of  $3 \times 3$ , as shown in Fig. 6.1(a). The neighborhood pixels ( $N_p$ ) in each block are thresholded by its center pixel value ( $N_c$ ). The basic concept is to consider whether a neighbor pixel is smaller than the central pixel value or not. If ( $N_p \geq N_c$ ), then the thresholded value will be 1 (if ( $N_p < N_c$ ), then thresholded value will be 0). This thresholded value will replace that neighbor pixel value. In this manner, a new threshold  $3 \times 3$  block will be produced where eight neighbors are changed, as shown in Fig. 6.1(b).
- These thresholded binary values (8 bits for 8 neighbors of the central pixel) are multiplied by the powers of two and then summed to obtain a label for the center pixel. This is simply a binary-to-decimal conversion.

- This new decimal value replaces the central pixel value. Based on the Fig. 6.1, we get a new value for the block in Fig. 6.1(a) according to the following computations,

<i>Binary Pattern:</i>	1 (MSB)	1	1	1	0	0	0	1 (LSB)
<i>Code/Weight(<math>2^p</math>):</i>	$1 \times 2^7$ =128	$1 \times 2^6$ =64	$1 \times 2^5$ =32	$1 \times 2^4$ =16	$0 \times 2^3$ =0	$0 \times 2^2$ =0	$0 \times 2^1$ =0	$1 \times 2^0$ =1
<i>LBP:</i>	$1 + 16 + 32 + 64 + 128 = 241$							

- So, the new central pixel will be 241. In this way, a new image or matrix is produced based on the local or neighbor binary pattern for block-by-block of that image.

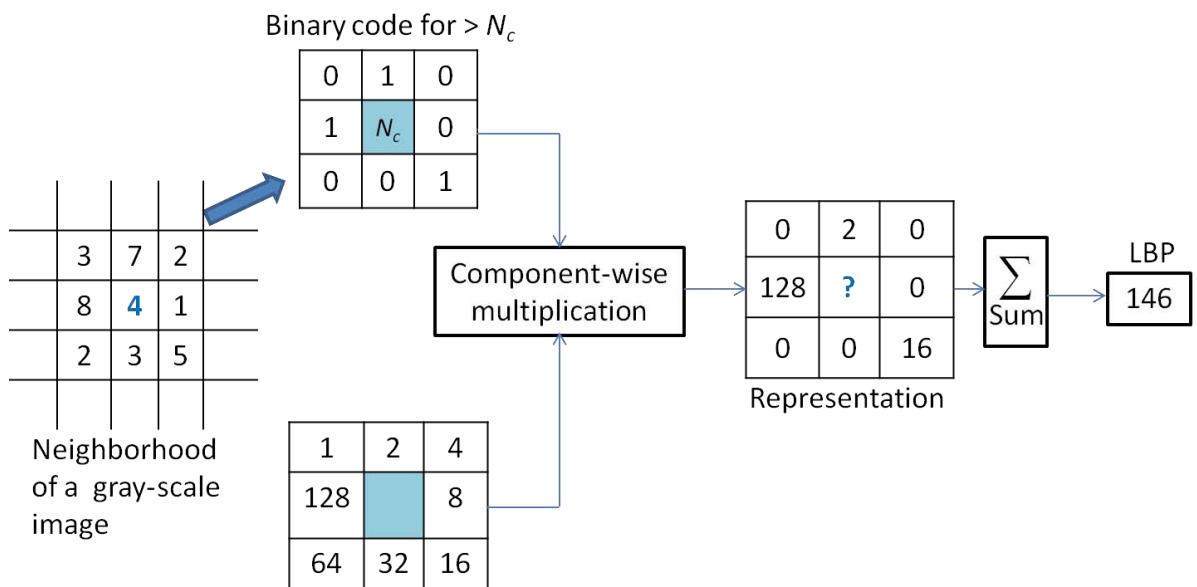


Figure 6.2: Another example of LBP computation.

If we consider 8 neighbors, i.e., if the neighborhood of a central pixel has 8 pixels or neighbors, then we can have  $2^8$  (=256) different labels. The above-mentioned process can be explained by the following mathematical expression:

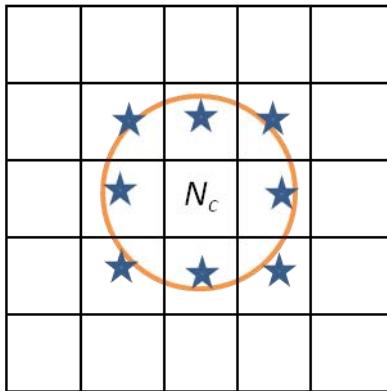
$$LBP_{p,r}(N_c) = \sum_{p=0}^{P-1} g(N_p - N_c)2^p$$

where, the binary threshold function  $g(x)$  is,

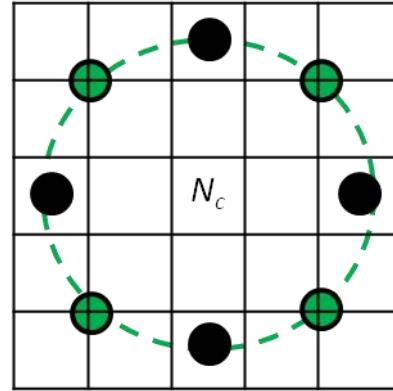
$$g(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases}$$

Figure 6.2 illustrates another example on how the LBP operator is computed.

So far, we discuss on basic local binary pattern, which is limited to 3x3 block or 8 neighbor pixels. On the other hand, the generic LBP operator has no limitations in terms of block size (it could be 3x3 or 5x5 or larger), or number of neighbors to consider. Figure 6.3 mentions two cases, one for 3x3 block where the selection of eight neighbor pixels are straight-forward.



(a)



(b)

Figure 6.3: LBP for 3x3 and 5x5 blocks for 8 neighborhood pixels.

In the case of 5x5 block (Fig. 6.3(b)), the crisscross values are easy to consider (pointed in black dots), but the diagonal values (as shown in green circle with black border) are difficult to consider directly. If a sampling is not in the center of a pixel location, it does not have any appropriate pixel values. In this case, interpolation or texture mapping is necessary.

Hence,

- We can take either the average of each four-pixel block in diagonal directions;
- Or, we can compute bilinear interpolation (also called as, bilinear texture mapping or bilinear filtering), if the sampling point is not in the center of a pixel [201].

### 6.3 Diagonal-Crisscross Local Binary Pattern (DCLBP)

In this thesis, we propose a new image or texture representative operator called *diagonal-crisscross local binary pattern* (DCLBP). It is based on the concept that an image feature should take diagonal pixel variations as well as horizontal and vertical (crisscross) pixel variations in the neighborhood, so that it can perform well even in the cases of rotations in images.

We also consider the center pixel value of each block for analysis, because in order to change a center pixel value with a new pixel value, the importance of that center pixel should be retained. Each block represents the neighborhood around a central pixel. In this case, we consider the differences of mutual *diagonal* pixels (both front-diagonal and back-diagonal) and the same for horizontal directions and vertical directions (called here as *crisscross*).

The computations of the neighborhood pixels within the square mask are thresholded differently in this case than the basic local binary pattern operator. According to the following strategy, we compute a new value. The diagonal-crisscross computation for a 3-by-3 mask or texture spectrum is shown in Fig. 6.4. Here for a 3-by-3 image patch or mask, we start from  $N_0$  pixel position from the left-topmost corner. We could start from another position too, which will be computed for all images.

Then we compute four difference values:

- Difference for front-diagonal values ( $N_0 - N_4$ ),
- Difference in vertical direction ( $N_1 - N_5$ ),
- Difference for back-diagonal direction ( $N_2 - N_6$ ), and
- Difference in horizontal direction ( $N_3 - N_7$ ).

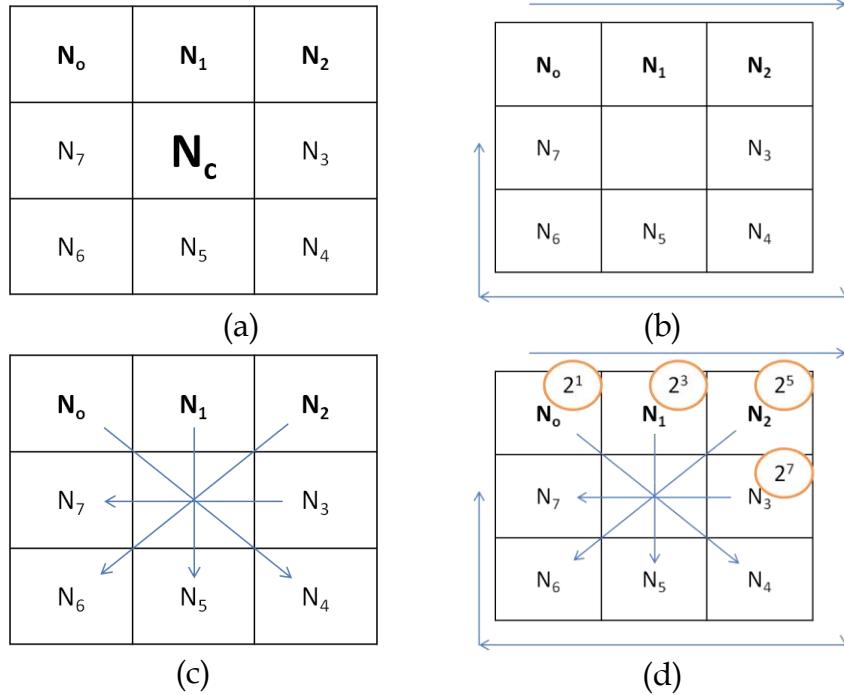


Figure 6.4: Computation of DCLBP:

- (a) 3-by-3 image patch;
- (b) Start from  $N_0$  pixel position;
- (c) Get differences for front-diagonal values ( $N_0 - N_4$ ), in vertical direction ( $N_1 - N_5$ ), for back-diagonal direction ( $N_2 - N_6$ ), and in horizontal direction ( $N_3 - N_7$ );
- (d) Multiply each difference with  $2^1$ ,  $2^3$ ,  $2^5$  and  $2^7$  sequentially to compute a new value.

Afterwards, we multiply each difference with  $2^1$ ,  $2^3$ ,  $2^5$  and  $2^7$  sequentially to compute a new value. We also computed  $2^0$ ,  $2^1$ ,  $2^2$  and  $2^3$  for one computation; as well as,  $2^4$ ,  $2^5$ ,  $2^6$  and  $2^7$ - to check the possibility of improvement. However, both cases provide poor results. It is due to the fact that, for the case of ' $2^0$ ,  $2^1$ ,  $2^2$  and  $2^3$ ' combinations, the maximum value becomes ' $2^0 + 2^1 + 2^2 + 2^3$ ', which is  $1+2+4+8 = 15$ . So the range from 0~255 for a gray-scale image turns into 0~15 values. Hence, the

results are poor for this combination. On the other hand, with the ‘ $2^4$ ,  $2^5$ ,  $2^6$  and  $2^7$ ’ combination, the maximum value becomes ‘ $2^4 + 2^5 + 2^6 + 2^7$ ’, which is  $16+32+64+128 = 240$ . Hence, the range of new values can be  $16\sim240$ , which is also relatively lower than the range of a gray-scale image. Another key point is the central pixel’s reflection in the new value.

So far, we have not computed or considered the central pixel value ( $N_c$ ). Based on our study, if we do not consider the central pixel value but replace it only through the neighborhood pixel values (as mentioned above), then the approaches do not demonstrate smarter representations. Therefore, in our proposed method, we consider ‘ $2^1$ ,  $2^3$ ,  $2^5$  and  $2^7$ ’ combinations (for the four differences as computed above), and the center pixel value ( $N_c$ ). We take the *mean* value of newly-computed value and the central pixel value. This *mean* value will replace the central pixel value ( $N_c$ ) and in this manner, we get the image based on the proposed *diagonal-crisscross local binary pattern* (DCLBP) method. The computation is done as per the equation below,

$$DCLBP_{p,r}(N_c) = \frac{[(\sum_{k=0}^{|P|-1} \vartheta(\delta_{k,|P|+k}) \times 2^{p_k \epsilon_p}) + N_c]}{2}$$

where,

The sampling-point set is,

$$P = \{1, 3, 5, 7\}$$

Cardinality of the set is,

$$|P| = 4$$

Difference parameter  $\delta$  is,

$$\delta_{k,|P|+k} = (N_k - N_{|P|+k})$$

The binary threshold function  $\vartheta(\delta)$  is,

$$\vartheta(\delta) = \begin{cases} 0, & \delta < 0 \\ 1, & \delta \geq 0 \end{cases}$$

The radius of the neighborhood is  $r$ , which can be 1 or more. For 3x3 image patch, the radius is 1. When sampling coordinates do not fall at integer positions (e.g., if the radius is 1, then sampling coordinates fall at integer positions), the intensity value is bilinearly interpolated. Increasing the value of radius enhances information redundancy and thereby, increases the computational cost. We can consider a multi-scaled computation where we can take different combinations of radius and then see its performance. But based on analysis, it does not provide stronger cue for better discriminating ability of different textures.

Here, the cardinality,  $|P| = 4$  that demonstrates that it has 4 different-possible values as ' $2^1$ ', ' $2^3$ ', ' $2^5$ ' and ' $2^7$ '. For  $k = 0$ , we get

$$\delta_{0,4+0} \times 2^1 = \delta_{0,4} \times 2^1$$

which, covers the front-diagonal difference of  $(N_0 - N_4)$ . Similarly,

when  $k = 1$ ,  $\delta_{1,5} \times 2^3$  covers the vertical difference of  $(N_1 - N_5)$ ;

when  $k = 2$ ,  $\delta_{2,6} \times 2^5$  covers the back-diagonal difference of  $(N_2 - N_6)$ ;

when  $k = 3$ ,  $\delta_{3,7} \times 2^7$  covers the horizontal difference of  $(N_3 - N_7)$ .

Through this manner, we get the new central pixel value for each patch.

## 6.4 Median-RILBP

The ‘Median-Rotational-Invariant-LBP’ method is proposed at the top of the concept of rotational-invariance. This approach returns the *median* value that neighbors may represent by rotating the computed decimal value (that is calculated based on the binary thresholding function). The rotation is done 8 times if  $P = 8$  and the median value is taken from these eight outcomes.

It is understood that when a rotation in an image is done, the corresponding LBP patterns can render into a different position, and can rotate about their center point or pixel. Therefore, if we circulate the LBP binary code for each case, and then consider one of these binary patterns as a central pixel value, then we could

minimize the rotational problem. Therefore, according to the following strategy, the circular rotation is done to get rotational-invariance on the computed patterns.

Under a rotational transform, the neighborhood pixel values of  $\{N_p | p = 0, 1, \dots, (P - 1)\}$  move along a circular path around the central pixel ( $N_c$ ) that produce different labels. This rotation-invariant concept can remove the rotational effects of images. We compute the *median* of each rotational pattern and then this *median* value becomes the new center pixel. It can be computed as follows,

$$\widetilde{LBP_{p,r}^{\alpha}(N_p)} = \text{median}\{\widetilde{\rho(LBP_{p,r}(N_p), p)}\}_{p=0,1,\dots,P-1}$$

where,

$\alpha$  symbolizes ‘rotational-invariant’ nature;

$p$  is pattern (e.g., 8 for a 3x3 patch);

$r$  is radius; and

$\rho(\ )$  is the rotational function where it circularly does bitwise right-shift operation. The bit-shift is done 8 times if  $P = 8$ . For example, if a usual computation provides a binary pattern – 00001010, then by the bit-wise right-shift operation, we get – 00000101, 10000010, 01000001, ..., 00010100. From these 8 binary patterns, we take the median of them. The concept here is to rotate the  $P$  neighbors and compute the *median* value that the neighbor chain may represent.

## 6.5 Mean-RILBP

This approach returns the *mean* value that neighbors may represent by rotating the computed decimal value (that is calculated based on the binary thresholding function). The rotation is done 8 times if  $P = 8$  and the *mean* value is taken from these eight outcomes. Based on the above-mentioned method ‘median-RILBP’, we propose another method where we consider the *mean* value instead of minimum or median values. This method performs well too. It can be computed as follows,

$$\overline{LBP_{p,r}^{\alpha}(N_p)} = \overline{\text{mean}\{\rho(LBP_{p,r}(N_p), i)\}|_{p=0,1,\dots,p-1}}$$

The basic notations are as above.

## 6.6 Experimental Setup

In this chapter, the proposed methods along with few other related methods are compared with two well-known benchmark datasets. These are: USC-SIPI (i.e., University of Southern California – Signal and Image Processing Institute) Rotated Textures dataset [112-114], and KTH-TIPS dataset [87,90,97,124,138,147,160].

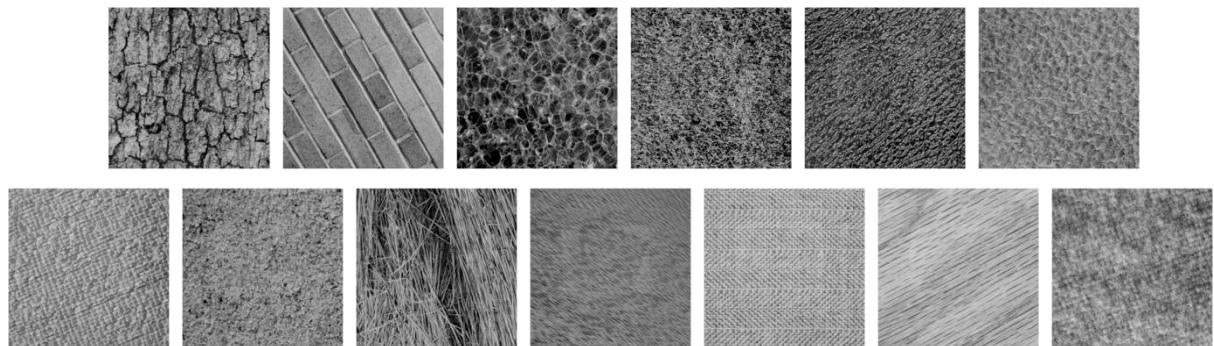


Figure 6.5: Sample images for each class of USC-SIPI Rotated Textures dataset.

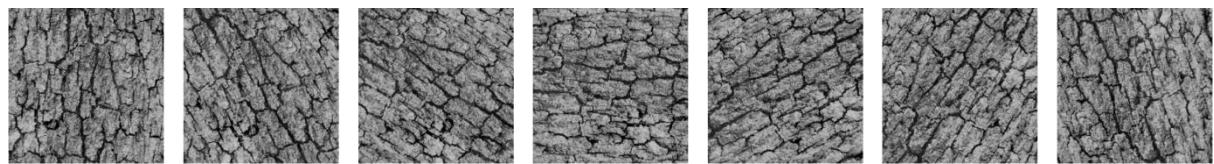


Figure 6.6: Seven rotated images for bark class (USC-SIPI Rotated Textures dataset).

We experiment on a dataset called USC-SIPI (i.e., University of Southern California – Signal and Image Processing Institute) Rotated Textures dataset [112-114] (Fig. 6.5), which has 7 different rotations (each image is digitized at seven different rotation angles: 0, 30, 60, 90, 120, 150, and 200 degree) for each of the 13 different images. Figure 6.6 shows 7 rotated images for bark class. These images are taken from the Brodatz database [69,71,80,83-97], which is the most widely-used dataset since long. In this dataset, there are 13 classes and each class has 7 images. These images have resolutions of 512x512 pixels (with 8 bits/pixel). These rotated images are scanned by employing a 512x512 pixel video digitizing camera. Hence,

the qualities of the scanned images are probably not as good as those in the main part images. It shows that our proposed three methods perform well.



Figure 6.7: KTH-TIPS dataset: (Top-row) (a) The variation with respect to scale in the KTH-TIPS database. Center image: It shows the central scale, and it was selected to correspond roughly to the scale used in the CUReT database. The left and right images are captured with the sample at half and twice that distance, respectively. (Bottom-row). (b) The variation of pose and illumination present in the KTH-TIPS database. It shows 3 out of 9 images per scale, showing the variation of pose and illumination. Prior to use, images were cropped so only foreground was present.

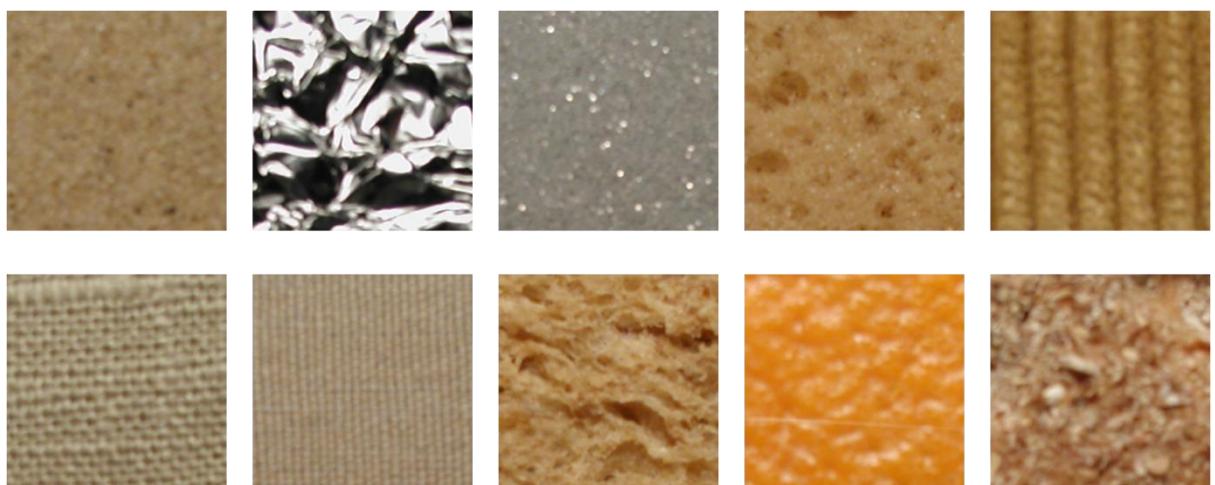


Figure 6.8: Ten different classes for KTH-TIPS dataset (ten images are shown: sandpaper, crumpled aluminum foil, styrofoam, sponge, corduroy, linen, cotton, brown bread, orange peel and cracker B).

We experiment on another very large dataset called KTH-TIPS (TIPS stands for ‘Textures under varying Illumination, Pose and Scale’) database

[87,90,97,124,138,147,160]. Figure 6.7 shows some sample images [160]. In this case, there are 10 different classes and each class has 81 images of different rotations and variations. These are sandpaper, crumpled aluminum foil, styrofoam, sponge, corduroy, linen, cotton, brown bread, orange peel and cracker B. These are imaged at nine distances from the camera to give equidistant log-scales over two octaves, as illustrated in Fig. 6.7(a) for the cracker [160]. At each distance images are captured using three different directions of illumination (front, side and top) and three different poses (central,  $22.5^\circ$  turned left,  $22.5^\circ$  turned right) giving a total of  $3 \times 3 = 9$  images per scale, and  $9 \times 9 = 81$  images per material. A subset of these is shown in Fig. 6.7(b). For each image, a  $200 \times 200$  pixel region is created by removing the background. Figure 6.8 shows ten sample images for this dataset.

## 6.7 Recognition Strategy

The classifications have been performed by two different classifiers. They are: *k*-Nearest Neighbors (KNN) classifier and Support Vector Machine (SVM).

### *k*-Nearest Neighbor:

It is one of the most basic but simple classification methods. It does not require any prior knowledge usually on the distribution of the data. It is a non-parametric method.

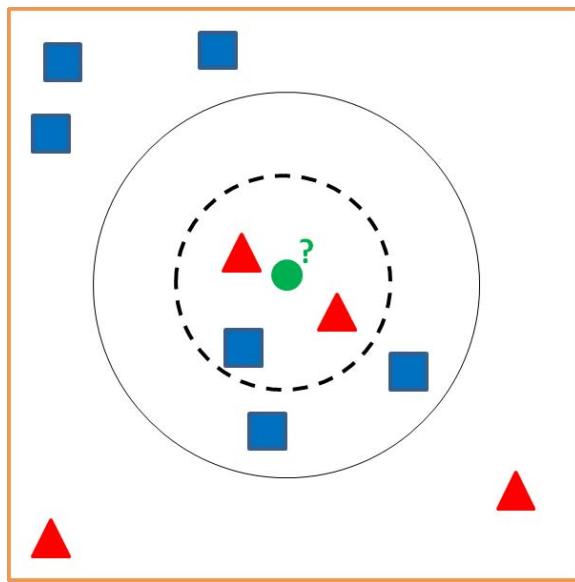


Figure 6.9: Example of *k*-NN classifier.

Figure 6.9 shows an example of  $k$ -NN classification. The **test sample** (**green circle**) should be classified. According to this example, the test sample is classified as one of the **blue squares** or to the second class of **red triangles**.

- Within dashed-line circle: if  $k=1$ , then the unknown test sample is classified as red triangle. Here, based on distance from the test sample 'green circle', the red triangle on left-top side has shortest distance.
- Within dashed-line circle: if  $k=3$ , then the unknown test sample is classified as red triangle. Here, based on distance from the test sample 'green circle'  
 $\rightarrow$ red triangle (left) < red triangle (right) < blue square.  
 Based on majority voting, red triangle is classified.
- Within solid circle: if  $k=1$ , then the unknown test sample is classified as red triangle. Here, based on distance from the test sample 'green circle', the red triangle on left-top side has shortest distance.
- Within solid circle: if  $k=3$ , then the unknown test sample is classified as red triangle. Here, based on distance from the test sample 'green circle'  
 $\rightarrow$ red triangle (left) < red triangle (right) < blue square (left) < blue square (right) < blue square (bottom)  
 Based on majority voting from top three cases as  $k=3$ , red triangle is classified (as top three has two red triangles and one blue square).
- Within solid circle: if  $k=5$ , then the unknown test sample is classified as blue square. Here, based on distance from the test sample 'green circle'  
 $\rightarrow$ red triangle (left) < red triangle (right) < blue square (left) < blue square (right) < blue square (bottom)  
 Based on majority voting from top five cases as  $k=5$ , blue square is classified (as top five has two red triangles and three blue squares).

### Support Vector Machine:

Support Vector Machines (SVMs) are supervised learning models and is one of the most widely used classifiers. In order to implement the SVM for classification,

we employ the LIBSVM [202]. It is a well-known library for SVM. Many papers in various conferences and journals use this library to implement SVM.

For the SVM, though there are different kinds kernels are proposed, found basic kernel functions are widely employed. These are: linear kernel function, polynomial kernel function, sigmoid function, and radial basis function (RBF). In this paper, we consider the radial basis function as kernel function, where the kernel parameter  $\gamma = 1$ . In this case, the following steps are considered for classifications:

- transform data to the format of an SVM package;
- conduct simple scaling on the data;
- select kernel function (we take the RBF kernel);
- use cross-validation to find the best parameter for the penalty parameter of the error term  $C$  and the gamma  $\gamma$ ;
- use the best parameters ( $C$  and  $\gamma$ ) to train the whole training set;
- finally, testing is done.

Scaling before applying SVM is done. Scaling can avoid attributes in greater numeric ranges dominating those in smaller numeric ranges. Scaling also can avoid numerical difficulties during the calculation. The range is considered within [0,1]. We use RBF kernel as it can nonlinearly map samples into a higher dimensional space.

As the above SVM is binary one - so it cannot be implemented directly for multi-class classification. Hence, it needs some modifications. LIBSVM implements the ‘one-against-one’ approach [203] for multiclass classification.

We partition samples into training and testing categories. The performance of most classifiers is typically evaluated through **cross-validation**, which involves the determination of classification accuracy for multiple partitions of the input samples used in training. For example, during 5-fold cross-validation training, a set of input samples is split up into 5 partitions ( $D_1, D_2, \dots, D_5$ ), having equal sample sizes to the extent possible. If possible, make each partition with equal number of sample sizes. To begin, for 5-fold cross-validation, samples in partitions ( $D_2, D_3, \dots, D_5$ ) are first

used for *training* while samples in partition ( $D_1$ ) are used for *testing*. Next, samples in groups ( $D_1, D_3, \dots, D_5$ ) are used for *training* and samples in partition ( $D_2$ ) used for *testing*. This is repeated until each partition has been used singly for testing. It is also customary to re-partition all of the input samples, e.g., 10 times in order to get a better estimate of accuracy. This strategy can be called *leave-one-group-out* or *n-fold* cross-validation scheme. If one does not consider partitions, but all classes, then using one class as *testing* and rest others as *training* (similar to the above concept), we can get *leave-one-out* cross-validation scheme.

For the SVM, we used the *n-fold* cross-validation scheme:

- partition a sample of data into complementary subsets,
- perform the analysis on one subset (*training* set), and
- validate the analysis on the other subset (*testing* set).

## 6.8 Result and Analysis

The recognition results show clearly that the proposed DCLBP method is the best considering the complexities of the datasets. These are classified by using two well-known and very widely used classifiers. These are the k-nearest neighbor classifier (KNN) and the support vector machine (SVM).

**Table 6.1:** Recognition results for USC-SIPI Rotated Textures dataset.

USC-SIPI Rotated Textures Dataset		
Feature	KNN	SVM
LBP	46	74.7
Interpolation_LBP	46.1	80.2
Median_LBP	30.8	80.2
MedianR_LBP	84.6	93.4
Number_LBP	92.3	87.9
NI_LBP	23.1	65.9
Mean-RILBP	100	95.6
Median-RILBP	92.3	96.7
DCLBP	100	90.1

**Table 6.2:** Recognition results for KTH-TIPS database.

KTH-TIPS DB		
Feature	KNN	SVM
LBP	80	53.7
Interpolation_LBP	60	53.1
Median_LBP	70	46.4
MedianR_LBP	70	28
Number_LBP	90	35.2
NI_LBP	70	55.7
Mean-RILBP	70	23.4
Median-RILBP	70	29.6
DCLBP	90	72.5

In this paper, we use LIBSVM library for multi-class classification with radial basis kernel function (default value for gamma in kernel function is 1). The following two tables show the comparative recognition results on the two datasets. Table 6.1 is for the famous USC-SIPI Rotated Textures dataset. Table 6.2 shows the recognition results for KTH-TIPS database.

## 6.9 Conclusions

The LBP is similar to the gray-scale based co-occurrence matrix or image: both have image on gray-scale variations and final features are computed directly from the LBP or co-occurrence matrices, instead of the original image. Therefore, in sequence of the earlier chapters, we introduce this chapter and propose method. In Chapter 5, we propose a category based on rotational-invariance/-variance issues of co-occurrence-based features. This chapter has concentrated on rotational-invariance properties to classify rotational images smartly based on LBP-based methods.

In this chapter, we propose a new method called *diagonal-crisscross local binary pattern* (DCLBP) for texture representations in spatial domain, in a smarter way compare to the original local binary pattern. The proposed method is tested on two very difficult benchmark databases. We also propose two other variants of the LBP considering rotational feature and mean/median of the neighbor pixels. In this

paper, we compare our proposed methods with other methods called median-LBP, Median-rotational-LBP, Interpolation-LBP, number-LBP, neighborhood-intensity-LBP. Details of these methods are available in literature and in the book on LBP [200].

Based on our experimental results in two benchmark datasets having rotational image, the proposed DCLBP method performs the best. The datasets are: USC-SIPI (i.e., University of Southern California - Signal and Image Processing Institute) Rotated Textures dataset, and KTH-TIPS (TIPS stands for ‘Textures under varying Illumination, Pose and Scale’) database. For rotational and complex texture analysis, compare with some other methods, we get satisfactory results. For recognition we use very well-known classification methods:  $k$ -nearest neighbor classifier and Support Vector Machine (SVM). Both of the classifiers show satisfactory results for our proposed methods.

# Conclusions and Future Work

## Chapter 7

---

### 7.1 Summary of the Thesis

In this chapter, we summarize our work as follows,

- We concentrate on several important areas on texture analysis. A survey is done based on these important features. We survey various important features that are suitable for texture analysis. There are various approaches for different types of texture images. So far, there is no comprehensive work that deals with key features.
- Apart from the issue of variety of features, different types of texture datasets are also discussed in-depth. However there are a number of texture datasets. There is no comprehensive works covering the important databases and analyzing these in various perspectives. We convincingly categorize texture databases, based on many references. In this survey, we put a nomenclature to split these texture datasets into few basic groups and later put related datasets. We discuss and analyze these in-depth and point some comparative issues.
- Next we exhaustively analyze eleven second-order statistical features or cues based on co-occurrence matrices to understand image texture surface. These features are exploited to analyze properties of image texture. The features are also categorized based on their angular orientations and it is noticeable that we can categorize these features into three different groups - orientation-invariant features, almost invariant to orientations, and

orientation-variant features. We categorize these features in terms of their applicability also. Based on two databases (standard database and real-world image database) we do our work. Standard database contain 21 images with different gray-level scratch. Our developed real-world database contains 17 classes.

- We propose a method called diagonal-crisscross local binary pattern (DCLBP) for texture analysis. We also propose another two methods based on median and average in rotational concept. Compare to the basic LBP and few other extensions, we find that our proposed method performs well in two very difficult benchmark datasets.

## 7.2 Outline on Future Work

In future, we want to analyze more on medical image processing-related texture datasets. As medical image processing is very sensitive and important area, so we need to find appropriate approaches for analysis to help medical physicians. However, there are only a few publicly-available datasets for medical texture analysis. We need to create some datasets and do study on the above approaches. We may also need to develop few more methods for these textures.

# Reference

---

1. F. Alam and R. Faruqui, "Optimized Calculations of Haralick Texture Features", European Journal of Scientific Research, Vol. 50, No. 4, pp. 543-553, 2011.
2. S. Zhu, Y. Wu, and D. Mumford, "Minimax entropy principle and its application to texture modeling", Neural Computation, Vol. 9, No. 8, pp. 1627-1660, 1997.
3. E. Adelson, "Image Statistics and surface perception", Proc. of SPIE-IS&T Electronic Imaging, SPIE, Vol. 6806, pp. 1-9, 2008.
4. Rosenfeld and A. Kak, Digital Picture Processing, Academic Press, 1982.
5. G. Castellano, L. Bonilha, L.M. Li, F. Cendes, "Texture analysis of medical images", Clinical Radiology, Vol. 59, pp. 1061-1069, 2004.
6. M. Unser, "Texture Classification and Segmentation using Wavelet Frames", IEEE Transactions on Image Processing, Vol. 4, No. 11, pp. 1549-1560, 1995.
7. S. A. Karkanis, G.D. Magoulas, M. Grigoriadou, and M. Schurr, "Detecting abnormalities in colonoscopic images by texture descriptors and neural networks", Proc. of the Workshop Machine Learning in Med. App., pp. 59-62.
8. S. Hossain and S. Serikawa, "Texture databases - a comprehensive survey", Pattern Recognition Letters, 2013.
9. U. Bagci, J. Yao, J. Caban, A. Suffrendini, T. Palmore, and D. Mollura, "Learning shape and texture characteristics of CT tree-in-bud opacities for CAD systems".
10. Qiaofeng Zhang and Jinfei Wang, "Texture analysis for urban spatial pattern study using SPOT imagery", IEEE, 2001.
11. D. Popescu, R. Dobrescu, and Maximilian Nicolae, "Texture Classification and Defect Detection by Statistical Features", International Journal of circuits, system and signal processing, 2007.
12. Y. Rubner and C. Tomasi, "Texture-Based Image Retrieval without Segmentation", 1999.
13. B. Aras, A. Ertuzun, and A. Ercil, "Higher order statistics based texture analysis method for defect inspection of textile products", IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing, pp.858-862, 1999.
14. K. Song, J. Kittler, and M. Petrou, "Defect detection in random colour textures", Image and Vision Computing, Vol. 14, pp. 667-683, 1996.
15. A. Conci and C. Proenca, "A fractal image analysis system for fabric inspection based on box-counting method", Computational Networks, 30, pp. 1887-1895, 1998.
16. A. Conci and C. Proenca, "A system for real-time fabric inspection and industrial decision", 14<sup>th</sup> International Conference on Software Engineering and Knowledge Engineering, pp. 707-714, 2002.
17. T. Maenpaa, "The local binary pattern approach to texture analysis – extensions and applications", Thesis, University of Oulu, 2003.
18. J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study", IJCV, Vol. 73, No. 2, pp. 213-238, 2007.
19. P.P. Ohanian and R.C. Dubes, "Performance Evaluation for Four Classes of Textural Features", Pattern Recognition, Vol. 25, pp. 819-833, 1992.
20. T. Randen and J.H. Husoy, "Filtering for Texture Classification: A Comparative Study", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 21, pp. 291-310, 1999.
21. T. Ojala, M. Pietikäinen and J. Nisula, "Determining Composition of Grain Mixtures by Texture Classification Based on Feature Distributions", Intl. J. of Pattern Recognition and Artificial Intelligence, Vol. 10, pp. 73-82, 1996.
22. T. Ojala and M. Pietikäinen, "Unsupervised Texture Segmentation using Feature Distributions", Pattern Recognition, Vol. 32, pp. 477-486, 1999.
23. K. Dana and J. Wang, "Device for Convenient Measurement of Spatially Varying Bidirectional Reflectance", J. Optical Soc. America, Vol. 21, No. 1, pp. 1-12, 2004.

24. A. Ngan and F. Durand, "Statistical Acquisition of Texture Appearance", Eurographics Symp. Rendering, pp. 31-40, 2006.
25. G. Beliakov, S. James, and L. Troiano, "Texture recognition by using GLCM and various aggregation functions", IEEE Int. Conf on Fuzzy Systems, 2008.
26. R. Dror, A. Willsky, and E. Adelson, "Statistical characterization of real-world illumination", Journal of Vision, Vol. 4, pp. 821-837, 2004.
27. S. Nishida and M. Shinya, "Use of image-based information in judgments of surface reflectance", Journal of the Optical Society of America, Vol. 15, pp. 2951-2965, 1998.
28. R. Fleming, R. Dror, and E. Adelson, "Real-world illumination and the perception of surface reflectance properties", Journal of Vision, Vol. 3, No. 5, pp. 347-368, 2003.
29. I. Motoyoshi, S. Nishida, and E. Adelson, "Adaptation to skewed image statistics alters perception of glossiness and lightness", European Conf. on Visual Perception, p. 24, 2005. <http://www.perceptionweb.com/ecvp05/0168.html>
30. I. Motoyoshi, S. Nishida, and E. Adelson, "Image statistics as a determinant of reflectance estimation", Vision Sciences Society Annual Meeting Abstracts, 2005. <http://journalofvision.org/5/8/569/>
31. S. Hossain and S. Serikawa, "Statistical Analysis and Psychological Evaluation of Surfaces under various Illumination", Int. J. of Applied Machines and Materials, Vol. 36, pp. 422-429, 2010.
32. L. Sharan, Y. Li, I. Motoyoshi, S. Nishida, and E. Adelson, "Image statistics for surface reflectance perception", Journal of Optical Society America, Vol. 25, No. 4, 2008.
33. I. Motoyoshi, S. Nishida, L. Sharan, and E. Adelson, "Image statistics and the perception of surface qualities", Nature, Vo. 447, pp. 206-209, 2007.
34. A. Baraldi and F. Parmiggiani, "An Investigation of the Textural Characteristics Associated with GLCM Matrix Statistical Parameters", IEEE Trans. on Geos. and Remote Sensing, Vol. 33, No. 2, pp. 293-304, 1995.
35. M. Mokji and S. Bakar, "Gray Level Co-Occurrence Matrix Computation Based on Haar Wavelet", Computer Graphics, Imaging and Visualisation, 2007.
36. M. Landy, "A gloss on surface properties", Nature, Vol. 447, pp. 158-159, 2007.
37. V. Kaji, B. Povazay, B. Hermann, Bernd Hofer, David Marshall, Paul L. Rosin, Wolfgang Drexler, "Robust segmentation of intraretinal layers in the normal human fovea using a novel statistical model based on texture and shape analysis", Optical Society of America, 2010.
38. S. Lazebnik, C. Schmid, and L. Ponce, "Affine-Invariant Local Descriptors and Neighborhood Statistics for Texture Recognition".
39. S. Bileschi and L. Wolf, "A Unified System for Object Detection, Texture Recognition, and Context Analysis Based on the Standard Model Feature Set", British Machine Vision Conference, 2005.
40. G. N. Srinivasan, and G. Shobha, "Statistical Texture Analysis", Proceedings of world academy of science, Vol. 36, 2008.
41. R. Haralick, K. Shanmugam, and I. Dinstein, "Textural Features for Image Classification", IEEE Trans. on Systems, Man and Cybernetics, Vol. 3, pp. 610-621, 1973.
42. L. Zhang, D. Tjondronegoro, and V. Chandran, "Evaluation of texture and geometry for dimensional facial expression recognition", International Conference on Digital Image Computing : Techniques and Applications (DICTA), 2011.
43. Y. Liu, M. Chen, H. Ishikawa, G. Wollstein, J. Schuman, and J. Rehg, "Automated macular pathology diagnosis in retinal OCT images using multi-scale spatial pyramid and local binary patterns in texture and shape encoding", Medical Image Analysis, Vol. 15, 2011.
44. B. Ghanem and N. Ahuja, "Maximum Margin Distance Learning for Dynamic Texture Recognition".
45. G. Zhao and M. Pietikainen, "Local Binary Pattern Descriptors for Dynamic Texture Recognition", Proc. of IEEE, 2006.
46. G. Corani, A. Giusti, D. Migliore, and J. Schmidhuber, "Robust Texture Recognition Using Credal Classifiers", BMVC, 2010.

47. L. Nanni, S. Brahma, and A. Lumini, "Random interest regions for object recognition based on texture descriptors and bag of features", *Expert Systems with Applications*, Vol. 39, 2012.
48. G. Zhao and M. Pietikainen, "Dynamic Texture Recognition Using Volume Local Binary Patterns".
49. M. Pietikainen, T. Ojala, and Z. Xu, "Rotation-invariant texture classification using feature distributions", *Pattern Recognition* 33 (2000) 43-52.
50. E. Wood, "Applying Fourier and associated transforms to pattern characterization in textiles", *Textile Research Journal*, Vol. 60, pp. 212-220, 1990.
51. M. Sharma, M. Markou, and S. Singh, "Evaluation of texture methods for image analysis", *Pattern Recognition Letters*. <http://www.eng.iastate.edu/ee528/Papers/texturecomparison.pdf>
52. P. Pachowicz and J. Bala, "Texture recognition through machine learning and concept optimization", P91-12, MLI 91-5.
53. A. Karahaliou et al., "A Texture Analysis Approach for Characterizing Microcalcifications on Mammograms", 2006.
54. F. Cohen, Z. Fan, and S. Attali, "Automated inspection of textile fabrics using textural models", *IEEE Trans. on PAMI*, Vol. 13, No. 8, pp. 803-809, 1991.
55. F. Pernkopf, "3D surface inspection using coupled HMMs", *International Conference on Pattern Recognition*, pp. 298-305, 2004.
56. M. Sayadi, S. Sakrani, F. Fnaiech, and M. Cheriet, "Gray-level Texture Characterization Based on a New Adaptive Nonlinear Auto-Regressive Filter", *Electronic Letters on Computer Vision and Image Analysis*, Vol. 7, No. 1, pp. 40-53, 2008.
57. M. Varma and R. Garg, "Locally Invariant Fractal Features for Statistical Texture Classification", *ICCV*, 2007.
58. Y. Xu, X. Yang, H. Ling, and H. Ji, "A New Texture Descriptor Using Multifractal Analysis in Multi-orientation Wavelet Pyramid", *IEEE CVPR*, 2010.
59. C. Palm, T. Lehmann, and K. Spitzer, "Color Texture Analysis of Moving Vocal Cords Using Approaches from Statistics and Signal Theory".
60. J. Bonet and P. Viola, "Texture Recognition Using a Non-parametric Multi-Scale Statistical Model".
61. N. Mittal, D. P. Mital, and K.L. Chan, "Features for texture segmentation using Gabor filters", *International Conference on Image Processing and its Applications*, pp. 353-357, 1999.
62. Y. Zhang, "MRI Texture Analysis in Multiple Sclerosis", *International Journal of Biomedical Imaging*, 2012.
63. A. Laine and J. Fan, "Texture classification by Wavelet packet signatures", *IEEE Trans. PAMI*, Vol. 15, No. 11, 1993.
64. T. Chang and C.-C. Kuo, "Texture analysis and classification with tree-structured wavelet transform", *IEEE Trans. Image Processing*, Vol. 2, pp. 429-441, 1993.
65. P. Southam and R. Harvey, "Towards texture classification in real scenes".
66. V. Manian and R. Vasquez, "Scaled and Rotated Texture Classification using a Class of Basis Functions", *Pattern Recognition*, Vol. 31, No. 12, pp. 1937-1948, 1998.
67. K. Rajpoot and N. Rajpoot, "Wavelets and Support Vector Machines for Texture Classification", *INMICS*, 2004.
68. M.A. Roula, J. Diamond, A. Bouridane, P. Miller, and A. Amira, "A Multispectral Computer Vision System for Automatic Grading of Prostatic Neoplasia", *IEEE International Symposium on Biomedical Imaging*, pp. 193-196, 2002.
69. J. Keller, J. M., S. Chen, and R. M. Crownover, "Texture Description and Segmentation through Fractal Geometry," *Computer Vision, Graphics, and Image Processing*, 45, pp. 150-166, 1989.
70. N. Harder, B. Neumann, M. Held, U. Liebel, H. Erfle, J. Ellenberg, R. Eils, and K. Rohr, "Automated recognition of mitotic patterns in fluorescence microscopy images of human cells", *IEEE Int. Symposium on Biomedical Imaging: From Nano to Macro*, pp. 1016-1019, 2006.

71. R. Sutton and E. Hall, "Texture Measures for Automatic Classification of Pulmonary Disease," *IEEE Transactions on Computers*, C-21, pp. 667-676, 1972.
72. V. Kovalev, F. Kruggel, H.-J Gertz, and D.Y. von Cramon, "Three-dimensional texture analysis of MRI brain datasets", *IEEE Trans. on Medical Imaging*, 20(5), pp. 424-433, 2001.
73. K. Otsuka, T. Horikoshi, S. Suzuki, and M. Fujii, "Feature extraction of temporal texture based on spatiotemporal motion trajectory", *ICPR*, 1998.
74. F. Kruggel and D. Y. von Cramon, "Alignment of magnetic-resonance brain datasets with the stereotactical coordinate system", *Med. Image Anal.*, 3, 2, pp. 175-185, 1999.
75. JelenaBozek, Mario Mustra, KresimirDelac, and MislavGrgic, A Survey of Image Processing Algorithms in Digital Mammography, *Rec. Advan. inMult. Sig. Process. and Commun.*, SCI 231, pp. 631-657, 2009.
76. A. Kassner and R. Thornhill, Texture Analysis: A Review of Neurologic MR Imaging Applications, *AJNR Am J Neuroradiol*, 31, pp. 809-816, 2010.
77. A. Abdalla, S. Dress, and N. Zaki, Detection of Masses in Digital Mammogram Using Second Order Statistics and Artificial Neural Network, *International Journal of Computer Science & Information Technology*, 3(3), 2011.
78. S. Bandyopadhyay, Pre-processing of Mammogram Images, *International Journal of Engineering Science and Technology*, 2(11), pp. 6753-6758, 2010.
79. S. Bandyopadhyay and I. Maitra, Digital Imaging in Mammography towards Detection and Analysis of Human Breast Cancer, *IJCA Special Issue on "Computer Aided Soft Computing Techniques for Imaging and Biomedical Applications"*, 2010.
80. R. Zwigelaar and C. Boggis, Classification of Linear Structures in Mammographic Images, *Conference on Medical Image Understanding and Analysis*, 2001.
81. M. Heath, K. Bowyer, D. Kopans, R. Moore, P. Kegelmeyer, "The digital database for screening mammography", *Int. Workshop on Digital Mammography*, pp. 212-218, 2000.
82. A. Karahaliou, I. Boniatis, S. Skiadopoulos, F. Sakellaropoulos, E. Likaki, G. Panayiotakis, and L. Costaridou, A Texture Analysis Approach for Characterizing Microcalcifications on Mammograms, 2006.
83. P. Brodatz, "Textures: A Photographic Album for Artists & Designers", Dover Publications, New York, 1966.
84. R. Peteri and M. Huskies, "DynTex: A comprehensive database of dynamic textures", [www.cwi.nl/projects/dyntex/](http://www.cwi.nl/projects/dyntex/), 2005.
85. S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using local affine regions", *IEEE Trans. PAMI*, 27(8), pp. 1265-1278, 2005.
86. [http://vivid.cse.psu.edu/texturedb/gallery/albums.php?set\\_albumListPage=2](http://vivid.cse.psu.edu/texturedb/gallery/albums.php?set_albumListPage=2)
87. J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study", *Computer Vision and Pattern Recognition workshop*, 2006.
88. T. Ojala, K. Valkealahti, E. Oja and M. Pietikäinen, "Texture Discrimination with Multidimensional Distributions of Signed Gray Level Differences", *Pattern Recognition*, 2001, vol. 34, pp. 727-739.
89. Y. Xu, J. Hui, and C. Fermuller, "A projective invariant for texture", *CVPR*, pp. 1932-1939, 2006.
90. A.T. Targhi, J.-M. Geusebroek, and A. Zisserman, "Texture Classification with Minimal Training Images", *ICPR*, 2008.
91. B. Caputo, E. Hayman, and P. Mallikarjuna, "Class-Specific Material Categorisation", *ICCV*, pp. 1597-1604, 2005
92. O. Cula, K. Dana, F. Murphy, and B. Rao, "Bidirectional imaging and modeling of skin texture", *IEEE Trans. Biomedical Engn.*, 2004.
93. J. Han and K. Perlin, "Measuring Bidirectional Texture Reflectance with a Kaleidoscope", *ACM Trans. Graphics*, 22(3), pp. 741-748, 2003.
94. J. Zhang and T. Tan, "Brief review of invariant texture analysis methods", *Pattern Recognition*, Vol. 35, pp. 735-747, 2002.
95. M. Varma and A. Zisserman, "Unifying Statistical Texture Classification Frameworks", Elsevier Science, 2004.

96. R. Picard, T. Kabir, and F. Liu, "Real-time recognition with the entire Brodatz texture database", IEEE CVPR, 1993.
97. H.-G. Nguyen, R. Fablet, and J.-M. Boucher, "Visual textures as realizations of multivariate log-Gaussian Cox processes", IEEE CVPR, pp. 2945-2952, 2011.
98. On Brodatz Texture, accessed on: Dec. 2012.  
<http://graphics.stanford.edu/projects/texture/faq/brodatz.html>
99. A. Carkacioglu, "Texture descriptors for content-based image retrieval", Thesis, The Middle East Technical University, Aug. 2003.
100. T. Hermes, A. Miene, and O. Moehrke, "Automatic Texture Classification by Visual Properties", Annual Conference of the Gesellschaft für Klassifikation, 1999.
101. H. Ong and H. Khoo, "Improved Image Texture Classification Using Grey Level Co-occurrence Probabilities with Support Vector Machines Post-Processing", European Journal of Scientific Research, Vol. 36, No. 1, 2009.
102. A. Materka and M. Strzelecki, "Texture Analysis Methods - A Review", Technical University of Lodz, Institute of Electronics, COST B11 report, Brussels, 1998.
103. T. Ojala and M. Pietikäinen, "Nonparametric multichannel texture description with simple spatial operators", Intl. Conf. on Pattern Recognition, Vol. 2, pp. 1052, 1998.
104. B. Georgescu, I. Shimshoni, and P. Meer, "Mean shift based clustering in high dimensions: A texture classification example", ICCV, pp. 456-463, 2003.
105. F. Liu and R. W. Picard, "Periodicity, directionality, and randomness: Wold features for image modeling and retrieval", IEEE Trans. PAMI, Vol. 18, No. 7, pp. 722-733, 1996.
106. K. Xu, B. Georgescu, D. Comaniciu, and P. Meer, "Performance analysis in content-based retrieval with textures", Intl. Conf. on Pattern Recognition, Vol. 4, pp. 275-278, 2000.
107. Y. Liu, Y. Tsin, and W. Lin, "The promise and perils of near-regular texture", IJCV, 2004.
108. <http://vismod.media.mit.edu/vismod/imagery/VisionTexture/> (Massachusetts Institute of Technology, Media Laboratory, Cambridge, Massachusetts)
109. 'How to download VisTex?' Available:  
<http://vismod.media.mit.edu/vismod/imagery/VisionTexture/distribution.html>
110. S. Singh and M. Sharma, "Texture analysis experiments with Meastex and Vistex benchmarks", Intl. Conf. on Advances in Pattern Recognition, pp. 417-424, 2011.
111. [http://www.texturesynthesis.com/meastex/www/for\\_images.html](http://www.texturesynthesis.com/meastex/www/for_images.html)
112. A.G. Weber, "The USC texture mosaic images", 2004. Accessed on: Dec. 2012. Available in <http://sipi.usc.edu/database/USCTextureMosaics.pdf>
113. USC-SIPI database, accessed on Dec. 2012.  
<http://sipi.usc.edu/database/database.php?volume=textures>
114. K. I. Laws, "Textured Image Segmentation", PhD thesis, University of Southern California, USCIP Report 940, 1980.
115. MeasTex mage texture database and test suit,  
<http://www.texturesynthesis.com/meastex/meastex.html>  
<http://textures.forrest.cz/>
117. T. Ojala, T. Mäenpää, M. Pietikäinen, J. Viertola, J. Kyllönen and S. Huovinen, "Outex - New Framework for Empirical Evaluation of Texture Analysis Algorithms", ICPR, pp. 701-706, 2002.
118. PhoTex Database. <http://www.cee.hw.ac.uk/texturelab/database/photex> and  
<http://www.macs.hw.ac.uk/texturelab/resources/databases/photex/>
119. O. Drbohlav and M. Chantler, "Illuminant-invariant texture classification using single training images" Texture05, 2005.
120. PhoTex in 3D. <http://www.taurusstudio.net/research/photex/index.htm>
121. <http://www.taurusstudio.net/research/pmtedb/download.htm>
122. [http://www.science.uva.nl/\\_mark/ALOT](http://www.science.uva.nl/_mark/ALOT)
123. M. Chantler, M. Petrou, A. Penirsche, M. Schmidt, and G. McGunnigle, "Classifying surface texture while simultaneously estimating illumination direction", IJCV, 2005.
124. G. Burghouts and J. Geusebroek, "Material-specific adaptation of color invariant features", Pattern Recognition Letters, Vol. 30, pp. 306-313, 2009.
125. P. Vácha, M. Haindl, and T. Suk, "Colour and rotation invariant textural features based on Markov random fields", Pattern Recognition Letters, Vol. 32, pp. 771-779, 2011.

126. Y. Xu, J. Hui, and C. Fermuller, "Viewpoint invariant texture description using fractal analysis", [http://www.cfar.umd.edu/~fer/website-texture/fractal\\_0816.pdf](http://www.cfar.umd.edu/~fer/website-texture/fractal_0816.pdf)
127. M. Varma and D. Ray, "Learning the Discriminative Power-Invariance Trade-off", ICCV, 2007.
128. [http://www.cfar.umd.edu/~fer/High-resolution-data-base/hr\\_database.htm](http://www.cfar.umd.edu/~fer/High-resolution-data-base/hr_database.htm)
129. Y. Xu, S. Huang, and H. Ji, "Integrating local feature and global statistics for texture analysis", ICIP, 2009.
130. Y. Xu, S.-B. Huang, H. Ji, AND C. Fermuller, "Combining powerful local and global statistics for texture description", CVPR, pp. 573-580, 2009.
131. UIUC Database. [http://www-cvr.ai.uiuc.edu/ponce\\_grp/data/index.html](http://www-cvr.ai.uiuc.edu/ponce_grp/data/index.html)
132. Outex Texture Database: [http://www.outex.oulu.fi/index.php?page=image\\_database](http://www.outex.oulu.fi/index.php?page=image_database)
133. <http://www.outex.oulu.fi/>
134. X. Shi and R. Manduchi, "On Rotational Invariance for Texture Recognition", 4th International Workshop on Texture Analysis and Synthesis, 2005.
135. Photometric Texture Database. <http://www.taurusstudio.net/research/pmtexdb/compare.htm>
136. K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink, "Reflectance and texture of real world surfaces", ACM Transactions on Graphics, 18(1), pp. 1-34, 1999.
137. K. Dana, and S. Nayar, "Correlation model for 3D texture", ICCV, pp. 1061-1067, 1999.
138. M. Varma and A. Zisserman, "Texture classification: Are filter banks necessary?", CVPR, pp. 691-698, 2003.
139. M. Varma and A. Zisserman, "A Statistical Approach to Texture Classification from Single Images", IJCV, pp. 61-81, 2005.
140. M. Varma and A. Zisserman, "Classifying images of materials: Achieving viewpoint and illumination independence", ECCV, pp. 255-271, 2002.
141. R. Broadhurst, "Statistical estimation of histogram variation for texture classification", Int. Workshop on Texture Analysis and Synthesis, pp. 25-30, 2005.
142. T. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons", IJCV, Vol. 43, No. 1, pp. 29-44, 2001.
143. O. Cula and K. Dana, "3D Texture recognition using bidirectional feature histograms", IJCV, Vol. 59, No. 1, 2004.
144. J. Geusebroek and A. Smeulders, "A Six-Stimulus Theory for Stochastic Texture", IJCV, pp. 7-16, 2005.
145. M. Pietikainen, T. Nurmiela, T. Maenpaa, M. Turtinen, "View-based recognition of real-world textures", Pattern Recognition, Vol. 37, No. 2, pp. 313-323, 2004.
146. O. Cula and K. Dana, "Compact representation of bidirectional texture functions", CVR2001, 2001.
147. M. Crosier and L. Griffin, "Texture Classification with a Dictionary of Basic Image Features", IEEE CVPR, 2008.
148. X. Xie and M. Mirmehdi, "Texture Exemplars for Defect Detection on Random Textures", ICAPR 2005, LNCS 3687, pp. 404-413, 2005.
149. G. Burghouts and J. Geusebroek, "Color Textons for Texture Recognition", BMVC, 2006.
150. M. Oren and S.K. Nayar, "Generalization of Lambert's Reflectance Model," ACM 21st Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH), pp. 239-246, 1994.
151. M. Oren and S.K. Nayar, "Seeing Beyond Lambert's Law", European Conference on Computer Vision, pp. 269-280, 1994.
152. J.J. Koenderink, A.J. van Doorn, and M. Stavridi, "Bidirectional reflection distribution function expressed in terms of surface scattering modes", European Conference on Computer Vision, pp. 28-39.
153. M. Sattler, R. Sarlette, and R. Klein, "Efficient and Realistic Visualization of Cloth", Eurographics Symp. Rendering, pp. 167-178, 2003.
154. J. Filip and M. Haindl, "Bidirectional Texture Function modeling: a state of the art survey", IEEE Trans. PAMI, Vol. 31, No. 11, pp. 1921-1940, 2009.

155. M. Koudelka, S. Magda, P. Belhumeur, and D. Kriegman, "Acquisition, Compression, and Synthesis of Bidirectional Texture Functions", Intl. Workshop Texture Analysis and Synthesis, pp. 47-52, 2003.
156. R. Furukawa, H. Kawasaki, K. Ikeuchi, and M. Sakauchi, "Appearance Based Object Modeling Using Texture Database: Acquisition, Compression and Rendering", Eurographics Workshop Rendering, pp. 257-266, 2002.
157. G. Muller, G. Bendels, and R. Klein, "Rapid Synchronous Acquisition of Geometry and BTF for Cultural Heritage Artefacts", Proc. Sixth Intl. Symp. Virtual Reality, Archaeology and Cultural Heritage, pp. 13-20, 2005.
158. A. Neubeck, A. Zalesny, and L. Gool, "3D Texture Reconstruction from Extensive BTF Data", Proc. Fourth Intl. Workshop Texture Analysis and Synthesis, pp. 13-18, 2005.
159. S. Ozdemir, A. Baykut, R. Meylani, A. Ercil, and A. Ertuzun, "Comparative Evaluation of Texture Analysis Algorithms for Defect Inspection of Textile Products", Intl. Conf. on Pattern Recognition, 1998.
160. E. Hayman, B. Caputo, M. Fritz, and J.-O. Eklundh, "On the Significance of Real-World Conditions for Material Classification", ECCV, pp. 253-266, 2004.
161. Y. Liu, W. Lin, and J. Hays, "Near-Regular Texture Analysis and Manipulation", ACM Transactions on Graphics (TOG), Vol. 23, No. 3, 2004.
162. P. Mallikarjuna, A.T. Targhi, Mario Fritz, Eric Hayman, B. Caputo, and Jan-OlofEklundh, "The KTH-TIPS2 database", <http://www.nada.kth.se/cvap/databases/kth-tips/kth-tips2.pdf>
163. <http://www.photomichaelwolf.com/hongkongarchitecture/>
164. B. Kjell, "Determining composition of grain mixtures using texture energy operators", SPIE Vol. 1825 Intelligent Robots and Computer Vision XI, pp. 395-400, 1992.
165. T. Ojala, M. Pietikäinen, and J. Nisula, "Determining Composition of Grain Mixtures by Texture Classification Based on Feature Distributions", Intl. J. of Pattern Recognition and Artificial Intelligence, Vol. 10, 1996.
166. G. Kylberg, "The Kylberg Texture Dataset v. 1.0, Centre for Image Analysis", Swedish University of Agricultural Sciences and Uppsala University, External report (Blue series) No. 35.
167. <http://www.cb.uu.se/~gustaf/texture/>
168. G. Doretto and S. Soatto, "Editable dynamic textures", IEEE CVPR, pp. 137-142, 2003.
169. L. Yuan, F. Wen, C. Liu, and H.-Y. Shum, "Synthesizing Dynamic Texture with Closed-Loop Linear Dynamic System", ECCV, pp. 603-616, 2004.
170. G. Doretto, A. Chiuso, Y.N. Wu, and S. Soatto, "Dynamic textures", International Journal of Computer Vision, Vol. 51, No. 2, pp. 91-109, 2003.
171. D. Culibrk, M. Mancas, and V. Crnojevic, "Dynamic Texture Recognition Based on Compression Artifacts", Towards Advanced Data Analysis by Combining Soft Computing and Statistics, Studies in Fuzziness and Soft Computing, vol. 285, pp. 253-266, 2012.
172. R. Peteri and D. Chetverikov, "Qualitative characterization of dynamic textures for video retrieval", ICCVG, 2004.
173. C.H. Peh and L.-F. Cheong, "Synergizing spatial and temporal texture", IEEE Trans. on Image Processing, Vol. 11, No. 10, pp. 1179-1191, 2002.
174. P. Saisan, G. Doretto, Y.Wu, and S. Soatto, "Dynamic texture recognition", CVPR, pp. 58-63, 2001.
175. A.B. Chan and N. Vasconcelos, "Supplemental material for classifying video with kernel dynamic textures", Statistical Visual Computing Lab, Tech. Rep. SVCL-TR-2007-03, 2007, <http://www.svcl.ucsd.edu>
176. M. Szummer, "Temporal Texture Modeling", Technical Report 346, MIT, 1995. <http://xenia.media.mit.edu/~szummer/icip-96/>
177. R. Peteri and D. Chetverikov, "Dynamic texture recognition using normal flow and texture regularity", Iberian Conf. on Pattern Recognition and Image Analysis, LNCS, pp. 223-230, 2005.
178. F. Woolfe and A. Fitzgibbon, "Shift-invariant dynamic texture recognition", ECCV, pp. 549-562, 2006.

179. M. Szummer and R. Picard, "Temporal Texture Modeling", ICIP, pp. 823-826, 1996.
180. A. B. Chan and N. Vasconcelos, "Layered dynamic textures", Neural Information Processing Systems, Vol. 18, pp. 203-210, 2006.
181. S. Fazekas and D. Chetverikov, "Normal Versus Complete Flow in Dynamic Texture Recognition: A Comparative Study", International Workshop on Texture Analysis and Synthesis, pp. 37-42, 2005.
182. A. Ravichandran, R. Chaudhry, and R. Vidal, "View-Invariant Dynamic Texture Recognition using a Bag of Dynamical Systems".
183. B. Ghanem and N. Ahuja, "Sparse Coding of Linear Dynamical Systems with an Application to Dynamic Texture Recognition".
184. G. Doretto and S. Soatto, "Editable dynamic textures", Technical Report 020001, UCLA Computer Science Department, 2002.
185. G. Doretto, D. Cremers, P. Favaro, and S. Soatto, "Dynamic texture segmentation", IEEE Conference on Computer Vision, pp. 44-49, 2003.
186. A. Chan and N. Vasconcelos, "Mixtures of dynamic textures", IEEE International Conference on Computer Vision, pp. 641-647, 2005.
187. L. Cooper, J. Liu, K. Huang, "Spatial segmentation of temporal texture using mixture linear models", Dynamical Vision Workshop in the International Conference of Computer Vision, 2005.
188. R. Vidal and A. Ravichandran, "Optical flow estimation and segmentation of multiple moving dynamic textures", IEEE Computer Vision and Pattern Recognition, pp. 516-521, 2005.
189. A. Schodl, R. Szeliski, D. H. Salesin, and I. Essa, "Video Textures", Proc. of Siggraph, pp. 489-498, 2000.
190. R. Peteri, S. Fazekas, and M.J. Huskies, "DynTex: A comprehensive database of dynamic textures", Pattern Recognition Letters, Vol. 31, No. 12, pp. 1627-1632, 2010.
191. DynTex database. <http://projects.cwi.nl/dyntex/database.html>
192. K.G. Derpanis and R.P. Wildes, "Dynamic Texture Recognition Based on Distributions of Spacetime Oriented Structure".
193. B. Woods, B. Clymer, T. Kurc, et al., "Malignant-lesion segmentation using 4D cooccurrence texture analysis applied to dynamic contrast-enhanced magnetic resonance breast image data", J. MagnReson Imaging, 25, pp. 495-501, 2007.
194. H. Greenspan, S. Belongie, R. Goodman and P. Perona, "Rotation Invariant Texture Recognition Using a Steerable Pyramid", Int. Conf. on Pattern Recognition, 1994.
195. J. Montoya-Zegarra, J. Papa, N. Leite, and R. Torres, and A. Falcao, "Rotation-invariant Texture Recognition", 3rd Int. Conf. on Advances in Visual Computing, 2007.
196. S. Hossain and S. Serikawa: Feature for texture analysis, SICE Annual Conf., Japan, pp. 1739-1744, 2012.
197. R. Connors, M. Trivedi, and C. Harlow, "Segmentation of a High-Resolution Urban Scene using Texture Operators", Computer Vision, Graphics, and Image Processing, Vol. 25, pp. 273-310, 1984.
198. Ojala, T., Pietikainen, M., and Harwood, D., "Performance evaluation of texture measures with classification based on Kullback discrimination of distributions", IAPR, Vol. 1, pp. 582-585 1994.
199. Wang, L. and He, D. C., "Texture classification using texture spectrum," Pattern Recognition, 23(8), 905-910, 1990.
200. M. Pietikäinen, A. Hadid, G. Zhao, and T. Ahonen, "Computer Vision using Local Binary Patterns", Springer, 2011.
201. Bilinear interpolation, [http://en.wikipedia.org/wiki/Bilinear\\_interpolation](http://en.wikipedia.org/wiki/Bilinear_interpolation)
202. Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, "A practical guide to support vector classification", Available in  
<http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>
203. S. Knerr, L. Personnaz, and G. Dreyfus, "Single-layer learning revisited: a stepwise procedure for building and training a neural network", Intl. J. Fogelman, editor, Neurocomputing: Algorithms, Architectures and Applications. Springer-Verlag, 1990.

204. U. H.-G. Kressel, "Pairwise classification and support vector machines", In B. Scholkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 255-268, Cambridge, MA, 1998. MIT Press.
205. T. Leung and J. Malik, "Recognizing surfaces using three-dimensional textons", International Conference on Computer Vision, 1999.
206. T.N. Tan, "Geometric transform invariant texture analysis", SPIE 2488, pp. 475-485, 1995.
207. T.N. Tan, "Texture segmentation approaches: a brief review", CIE and IEEE International Conference on Neural Networks and Signal, 1993.
208. T. R. Reed and Hans Du Buf, "A Review of Recent Texture Segmentation and Feature Extraction Technique", Computer Vision and Image Understanding, Vol. 57, No. 3, pp. 359-372, 1993.
209. J. Portilla and E. P. Simoncelli, "A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients", International Journal of Computer Vision, Vol. 40, No. 1, pp. 49-71, 2000.
210. Larry S. Davis, "Polarogram: a new tool for image texture analysis", Pattern Recognition 13 (3) (1981) 219-223.
211. Kashyap RL, Khotanzad A. A Model-Based Method for Rotation Invariant Texture Classification. IEEE Trans Pattern Anal Machine Intell. 1986;8:472-481.
212. Mao J, Jain AK. Texture classification and segmentation using multiresolution simultaneous autoregressive models. Pattern Recognition. 25(2):173-188. 1992.
213. Kourosh Jafari-Khouzani and Hamid Soltanian-Zadeh, "Rotation Invariant Multiresolution Texture Analysis Using Radon and Wavelet Transforms", IEEE Trans. Image Processing, Vol. 14, No. 6, pp. 783-795, 2005.
214. Cohen FS, Fan Z, Patel MA. Classification of rotated and scaled textured images using Gaussian Markov random field models. IEEE Trans Pattern Anal Machine Intell. 1991 Feb;13(2):192-202.
215. Chen J-L, Kundu A. Rotation and gray scale transform invariant texture identification using wavelet decomposition and hidden Markov model. IEEE Trans Pattern Anal Machine Intell. 1994 Feb;16(2):208-214.
216. Pun CM, Lee MC. Log-polar wavelet energy signatures for rotation and scale invariant texture classification. IEEE Trans Pattern Anal Machine Intell. 2003 May;25(5):590-603
217. Manthalkar R, Biswas PK, Chatterji BN. Rotation and scale invariant texture features using discrete wavelet packet transform. Pattern Recognition Letters. 24:2455-2462. 2003.
218. Charalampidis D, Kasparis T, "Wavelet-based rotational invariant roughness features for texture classification and segmentation", IEEE Trans Image Process. 2002; 11(8):825-37.
219. Haley GM, Manjunath BS, "Rotation-invariant texture classification using a complete space-frequency model", IEEE Trans Image Process. 1999; 8(2):255-69.
220. Wu WR, Wei SC, "Rotation and gray-scale transform-invariant texture classification using spiral resampling, subband decomposition, and hidden Markov model", IEEE Trans Image Process. 1996;5(10):1423-34.
221. Do MN, Vetterli M., "Rotation invariant characterization and retrieval using steerable wavelet-domain hidden Markov models", IEEE Trans Multimedia. 2002 Dec;4(4):517-526.
222. Magli E, Lo Presti L, Olmo G. "A pattern detection and compression algorithm based on the joint wavelet and Radon transform", Proc. IEEE 13th Int. Conf. Digital Signal Processing; 1997. pp. 559-562
223. Warrick AL, Delaney PA. "Detection of linear features using a localized Radon transform with a wavelet filter", Proc. ICASSP; 1997. pp. 2769-2772.
224. Leavers VF, "Use of the two-dimensional Radon transform to generate a taxonomy of shape for the characterization of abrasive powder particles", IEEE Trans Pattern Anal Machine Intell. 2000 Dec;22(12):1411-1423
225. Do MN, Vetterli M, "The finite ridgelet transform for image representation", IEEE Trans Image Process. 2003; 12(1):16-28.
226. Starck JL, Candès EJ, Donoho DL, "The curvelet transform for image denoising", IEEE Trans Image Process. 2002; 11(6):670-84.
227. S. R. Deans, *The Radon Transform and Some of Its Applications*. New York: Wiley, 1983.

228. M. Mayorga and L. Ludman, "Shift and rotation invariant texture recognition with neural nets", Proceedings of IEEE International Conference On Neural Networks, 1994, pp. 4078– 4083.
229. Timo Ojala, Matti Pietikäinen and Topi Mäenpää, "Multiresolution Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns", IEEE Trans. PAMI.
230. D. Casasent, D. Psaltis, Position, rotation and scale invariant optical correlation, Appl. Opt. 15 (7) (1976) 1795–1799.
231. Y.L. Sheng, H. Arsenault, Experiment on pattern recognition using invariant Fourier-Mellin descriptors, Opt. Soc. Am. A 3 (6) (1986) 771–776.
232. Yunlong Sheng, Jacques Duvernay, Circular-Fourier-radical-Mellin transform descriptors for pattern recognition, JOSA Communications, 3 (6) (1986) 885 –888.
233. H.H. Arsenault, Yunlong Sheng, Properties of the circular harmonic expansion for rotation-invariant pattern recognition, Appl. Opt. 25 (18) (1986) 3225 –3229.
234. J. Davernoy, Optical digital processing of directional terrain textures invariant under translation, rotation, and change of scale, Applied Optics, 23 (6) (1984) 828–837.
235. M.K. Tsatsanis, G.B. Giannakis, Object and texture classification using high order statistics, IEEE Trans. PAMI 14 (7) (1992) 733–750.
236. S. Chang, L.S. Davis, S.M. Dunn, J.-D. Eklundh, A. Rosenfeld, Texture discrimination by projective invariants, Pattern Recognition Lett. 5 (1987) 337–342.
237. B. Julesz, T. Caelli, "On the limits of Fourier decompositions in visual texture perception", Perception, 8, pp. 69-73, 1979.
238. B. Julesz, "A theory of preattentive texture discrimination based on first-order statistics of textons", Biological Cybernetics, Vol. 41, No. 2, pp. 131-138, 1981.
239. Julesz, B.: Spatial nonlinearities in the instantaneous perception of textures with identical power spectra. Philos. Trans. R. Soc. London B290, 83–94 (1980).
240. Marr, D.: Early processing of visual information. Philos. Trans. R. Soc. London B275, 483–524 (1976)
241. Caelli, T.M., Julesz, B.: On perceptual analyzers underlying visual texture discrimination: Part I. Biol. Cybernetics 28, 167–175 (1978)
242. Terry Caelli, Bela Julesz, Edgar Gilbert, "On perceptual analyzers underlying visual texture discrimination: Part II", Biological Cybernetics, Vol. 29, No. 4, pp. 201-214, 1978.
243. B. Julesz, "The Role of Terminators in Preattentive Perception of Line Textures", Recognition of Pattern and Form, Lecture Notes in Biomathematics, Vol. 44, pp. 33-58, 1982.
244. M. Hu, Visual pattern recognition by moment invariants, IRE Trans. Inform. Theory 8 (1962) 179–187.
245. A. Khotanzad, Y.H. Hong, Invariant image recognition by Zernike moments, IEEE Trans. PAMI 12 (1990), 489–497.
246. L. Wang, G. Healey, Using Zernike moments for the illumination and geometry invariant classification of multispectral texture, IEEE Trans. Image Process. 7 (2) (1998) 196–203.
247. Robert R. Bailey, Mandyam Srinath, Orthogonal moment features for use with parametric and non-parametric classifiers, IEEE Trans. PAMI 18 (4) (1996) 389–399.
248. C.-H. The, R.T. Chin, On image analysis by the methods of moments, IEEE Trans. PAMI 10 (1988) 496–513.
249. J. Flusser, T. Suk, and B. Zitová, "Moments and Moment Invariants in Pattern Recognition", Wiley & Sons Ltd., 2009.
250. J. Flusser, "Moment Invariants in Image Analysis", World Academy of Science, Engineering and Technology, Vol. 11, pp. 196-201, 2006.
251. MAR Ahad, J. Tan, H. Kim, and S. Ishikawa, "Lower-Dimensional Feature Sets for Template-based Motion Recognition Approaches", J. of Computer Science, Vol. 6, No. 8, pp. 920-927, 2010.
252. Y. Li, "Reforming the theory of invariant moments for pattern recognition," Pattern Recognition, Vol. 25, pp. 723–730, 1992.
253. A. Wallin and O. Kubler, "Complete sets of complex Zernike moment invariants and the role of the pseudo-invariants," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 17, pp. 1106–1110, 1995.

254. M. R. Teague, "Image analysis via the general theory of moments," *J. Optical Soc. of America*, vol. 70, pp. 920–930, 1980.
255. L. Wang and G. Healey, "Using Zernike moments for the illumination and geometry invariant classification of multispectral texture," *IEEE Trans. Image Processing*, vol. 7, pp. 196–203, 1998.
256. Y. S. Abu-Mostafa and D. Psaltis, "Recognitive aspects of moment invariants," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, pp. 698–706, 1984.
257. J. Flusser and T. Suk, "Pattern recognition by affine moment invariants," *Pattern Recognition*, vol. 26, pp. 167–174, 1993.
258. T. H. Reiss, "The revised fundamental theorem of moment invariants," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, pp. 830–834, 1991.
259. J. Flusser and T. Suk, "Pattern Recognition by Means of Affine Moment Invariants," Tech. Rep. 1726, UTIA AV CR, Praha, 1991.
260. Suk Tomáš, Flusser Jan: Graph method for generating affine moment invariants, Proceedings of the 17th International Conference on Pattern Recognition. ICPR 2004, p. 192-195, 2004.
261. Suk Tomáš, Flusser Jan: Affine Moment Invariants Generated by Automated Solution of the Equations, Proceedings of the 19th International Conference on Pattern Recognition, ICPR 2008.
262. Suk Tomáš, Flusser Jan: The Independence of the Affine Moment Invariants, Information Optics, p. 387-396, Eds: Cristobal G., International Workshop on Information Optics.
263. Suk Tomáš, Flusser Jan: Tables of Affine Moment Invariants Generated by the Graph Method, UTIA AV CR, (Praha 2005), Research Report 2156.
264. Suk Tomáš, Flusser Jan: Affine normalization of symmetric objects, Proceedings of the 7th International Conference on Advanced Concepts for Intelligent Vision Systems vol. 3708, pp. 100-107, 2005.
265. G.M. Haley and B.S. Manjunath, Rotation-invariant texture classification using modified Gabor filters, in Proc. IEEE Conference on Image Processing, Austin, TX, 655-659 (1994).
266. M.M. Leung and A.M. Peterson, Scale and rotation invariant texture classification, in Proc. 26th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA (1992).
267. M. Porat and Y. Zeevi, Localized texture processing in vision: analysis and synthesis in the Gaborian space, *IEEE Transactions on Biomedical Engineering*, Vol. 36, pp. 115-129, 1989.
268. S.V.R. Madiraju and C.C. Liu, Rotation invariant texture classification using covariance, in Proc. IEEE Conference on Image Processing, Vol. 1, Washington, D.C., 262-265 (1995).
269. J. You and H. Cohen, Classification and segmentation of rotated and scaled textured images using texture "tuned" masks, *Pattern Recognition* 26, 245-258 (1993).
270. M. Pietikäinen, T. Ojala and O. Silven, "Approaches to texture-based classification, segmentation and surface inspection", in *Handbook of Pattern Recognition & Computer Vision* (Second Edition), C.H. Chen, L.F. Pau, and P.S.P. Wang, eds., World Scientific, Singapore (1998).
271. T. Ojala, Multichannel approach to texture description with feature distributions, Technical Report CAR-TR-846, Center for Automation Research, University of Maryland (1998).
272. D. Harwood, T. Ojala, M. Pietikäinen, S. Kelman, and L.S. Davis, Texture classification by center-symmetric auto-correlation, using Kullback discrimination of distributions, *Pattern Recognition Letters* 16, 1-10 (1995).
273. K. Sivakumar, Morphologically constrained GRFs: applications to texture synthesis and analysis, *IEEE Trans. PAMI* 21 (2) (1999) 148–153.
274. Fang Liu, R.W. Picard, Periodicity, directionality, and randomness: wold features for perceptual pattern recognition, Proceedings of the International Conference on Pattern Recognition, Vol. 11, Jerusalem, October 1994, pp. 184–185.
275. Fang Liu, R.W. Picard, Periodicity, directionality, and randomness: wold features of image and modeling and retrieval, *IEEE Trans. Pattern Anal. Mach. Intell.* 18 (7) (1996) 722–733.

276. R. Chellappa, R.L. Kashyap, B.S. Manjunath, "Model based texture segmentation and classification", in: C.H. Chen, L.F. Pau, P.S.P. Wang (Eds.), *Handbook of Pattern Recognition and Computer Vision*, 1993, World Scientific Publishing, pp. 277-310.
277. M. Tuceryan and 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*, 2<sup>nd</sup> edition, World Scientific Publishing Co., pp. 207-248, 1998.
278. A.R. Rao, *A Taxonomy for Texture Description and Identification*, Springer, Berlin, 1990.
279. F. Tomita, S. Tsuji, *Computer Analysis of Visual Textures*, Kluwer Academic, Hingham, MA, 1990.
280. R.M. Haralick, Statistical and structural approaches to Texture, Proc. IEEE 67 (1979) 786-804.
281. L. Van Gool, P. Dewaele, A. Oosterlinck, Survey: texture analysis anno 1983, CVGIP 29 (1985) 336-357.
282. Tuceryan, M. and A. K. Jain, "Texture Segmentation Using Voronoi Polygons," IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-12, pp. 211-216, 1990.
283. Ahuja, N., "Dot Pattern Processing Using Voronoi Neighborhoods," IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-4, pp. 336-343, 1982.
284. Voorhees, H. and T. Poggio, "Detecting textons and texture boundaries in natural images," In Proceedings of the First International Conference on Computer Vision, pp. 250-258, London, 1987.
285. Blostein, D. and N. Ahuja, "Shape from Texture: Integrating Texture-Element Extraction and Surface Estimation," IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-11, pp. 1233-1251, 1989.
286. Zucker, S. W., "Toward a model of Texture," Computer Graphics and Image Processing, 5, pp. 190-202, 1976.
287. Chellappa, R. and S. Chatterjee, "Classification of Textures Using Gaussian Markov Random Fields," IEEE Transactions on Acoustic, Speech, and Signal Processing, ASSP-33, pp. 959-963, 1985.
288. Cross, G.C. and A.K. Jain, "Markov Random Field Texture Models," IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5, pp. 25-39, 1983.
289. Khotanzad, A. and R. Kashyap, "Feature Selection for Texture Recognition Based on Image Synthesis," IEEE Transactions on Systems, Man, and Cybernetics, 17, pp. 1087-1095, 1987.
290. Cohen, F.S. and D.B. Cooper, "Simple Parallel Hierarchical and Relaxation Algorithms for Segmenting Noncausal Markovian Random Fields," IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-9, pp. 195-219, 1987.
291. Therrien, C.W., "An Estimation-Theoretic Approach to Terrain Image Segmentation," Computer Vision, Graphics, and Image Processing, 22, pp. 313-326, 1983.
292. Geman, S. and D. Geman, "Stochastic Relaxation, Gibbs distributions, and the Bayesian Restoration of Images," IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6, pp. 721-741, 1984.
293. Besag, J., "Spatial Interaction and the Statistical Analysis of Lattice Systems," Journal of Royal Statistical Society, B-36, pp. 344-348, 1974.
294. Derin, H. and H. Elliott, "Modeling and segmentation of noisy and textured images using Gibbs random fields," IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI, Vol. 9, No. 1, pp. 39-55, 1987.
295. Qi Jia, Xuliang Lv, Xianhui Rong, "Application of Markov random field model in the design of digital pattern painting", International Conference on Artificial Intelligence, Management Science and Electronic Commerce, 2011.
296. M. Hassner and J. Sklansky. The use of Markov Random Fields as Models of Textures. Computer Graphics and Image Processing, 12:357-370, 1980.
297. R. Chellappa. Two Dimensional Discrete Gaussian Markov Random Field Models for Image Processing. In L. N. Kanal and A. Rosenfeld, editors, *Progress in Pattern Recognition*, volume 2, pages 79-122. North Holland Pub. Co., 1986.
298. J. E. Besag. On the Statistical Analysis of Dirty Pictures. *Journal of the Royal Statistical Society, B*48:259-302, 1986.

299. G. L. Gimel'farb. Texture modeling with multiple pairwise pixel interactions. In IEEE Trans. Pattern Anal. Mach. Intell., volume 18, pages 1110-1114, 1996.
300. S. C. Zhu, Y. N. Wu, and D. Mumford. Filters, random fields and maximum entropy (frame): To a unified theory for texture modeling. International Journal of Computer Vision, 27(2):107-126, 1998.
301. Rupert Paget and Dennis Longsta, "Texture Synthesis via a Non-parametric Markov Random Field", Digital Image Computing: Techniques and Applications (DICTA), 1995.
302. H. Derin, H. Elliot, R. Cristi, and D. Geman. Bayes smoothing algorithm for segmentation of images modelled by markov random field. IEEE Trans. Pattern Anal. Mach. Intell., 6(6):707-720, 1984.
303. B. S. Manjunath and R. Chellappa. Unsupervised texture segmentation using markov random field models. IEEE Trans. Pattern Anal. Mach. Intell., 13(5):478-482, 1991.
304. A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In ICCV (2), pages 1033-1038, 1999.
305. L. Wei and M. Levoy. Fast texture synthesis using tree-structured vector quantization. In K. Akeley, editor, Siggraph 2000, Computer Graphics Proceedings, pages 479-488. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.
306. V. Kwatra, A. Schödl, I. A. Essa, G. Turk, and A. F. Bobick. Graphcut textures: Image and video synthesis using graph cuts. In ACM Transactions on Graphics, SIGGRAPH 2003, volume 22(3), pages 277-286, July 2003.
307. L. Liang, C. Liu, and H. Y. Shum. Real-time texture synthesis by patch-based sampling. Technical Report MSR-TR-2001-40, Microsoft Research, 2001.
308. Mandelbrot, B. B., The Fractal Geometry of Nature, Freeman, San Francisco, 1983.
309. Pentland, A., "Fractal-based description of natural scenes," IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-9, pp. 661-674, 1984.
310. Voss, R., "Random fractals: Characterization and measurement," In Scaling Phenomena in Disordered Systems, R. Pynn and A. Skjeltorp, (Editors), Plenum, New York, 1986.
311. Super, B. J. and A. C. Bovik, "Localized Measurement of Image Fractal Dimension Using Gabor Filters," Journal of Visual Communication and Image Representation, 2, pp. 114-128, 1991.
312. Chen, C. C., J. S. Daponte, and M. D. Fox, "Fractal Feature Analysis and Classification in Medical Imaging," IEEE Transactions on Medical Imaging, 8, pp. 133-142, 1989.
313. Lundervold, A., Ultrasonic Tissue Characterization - A Pattern Recognition Approach, Technical Report, Norwegian Computing Center, Oslo, Norway, 1992.
314. Schistad, A. H. and A. K. Jain, "Texture Analysis in the Presence of Speckle Noise," In Proceedings of IEEE Geoscience and Remote Sensing Symposium, Houston, TX, May 1992.
315. Triloki Pant, Dharmendra Singh, Tanuja Srivastava, "Advanced fractal approach for unsupervised classification of SAR images", Advances in Space Research, Vol. 45, No. 11, pp. 1338-1349, 2010.
316. Fabrizio Berizzi, Gabriele Bertini, Marco Martorella, Massimo Bertacca, "Two-Dimensional Variation Algorithm for Fractal Analysis of Sea SAR Images", IEEE Transactions on Geoscience and Remote Sensing, Vol. 44, No. 9, pp. 2361-2373, 2006.
317. K.C. Clarke, "Computation of the fractal dimension of topographic surfaces using the triangular prism surface area method", Computers & Geosciences, Vol. 12, No. 5, pp. 713-722, 1986.
318. H. Zhao and S. Wang, "Application of Fractal Mathematics in Mode Recognition and Image Processing", Intl. Conf. on Measuring Technology and Mechatronics Automation, 2011.
319. Benoit B. Mandelbrot, Dann. E. Passoja, Alvin J. Paullay, "Fractal character of fracture surfaces of metals", Nature, Vol. 308, No. 5961, pp. 721-722, 1984.
320. J. J. Gagnepain, C. Roques-carmes, "Fractal approach to two-dimensional and three-dimensional surface roughness", Wear , vol. 109, no. 1-4, pp. 119-126, 1986.
321. P.-W. Huang and C.-H. Lee, "Automatic Classification for Pathological Prostate Images Based on Fractal Analysis", IEEE Transactions on Medical Imaging, vol. 28, no. 7, pp. 1037-1050, 2009.

322. Edward D. Weinberger and Peter F. Stadler, "Why *Some* Fitness Landscapes are Fractal", 1993.
323. Malik, J. and P. Perona, "Preattentive Texture Discrimination with Early Vision Mechanisms," Journal of the Optical Society of America, Series A, 7, pp. 923-932, 1990.
324. Unser, M. and M. Eden, "Nonlinear Operators for Improving Texture Segmentation Based on Features Extracted by Spatial Filtering," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 20, No. 4, pp. 804-815, 1990.